

Proceedings of the  
Federated Conference on  
Computer Science and  
Information Systems



September 18–21, 2011.  
Szczecin, Poland

ISBN 978-83-60810-22-4

**Proceedings of the Federated Conference on Computer Science  
and Information Systems**

**2011**

M. Ganzha, L. Maciaszek, M. Paprzycki (editors)

ISBN 978-83-60810-22-4

IEEE Computer Society Press  
10662 Los Vaqueros Circle  
Los Alamitos, CA 90720  
USA

TEXnical editor: Aleksander Denisiuk



# Proceedings of the Federated Conference on Computer Science and Information Systems

September 18–21, 2011. Szczecin, Poland



2011



**D**EAR Reader, it is our pleasure to present to you Proceedings of the 2011 Federated Conference on Computer Science and Information Systems (FedCSIS), which took place in Szczecin, Poland, on September 18–21, 2011.

FedCSIS was organized by the Polish Information Processing Society (Mazowsze and West Pomerania Chapters) in technical cooperation with: IEEE Region 8, Computer Society Chapter Poland, Gdańsk Computer Society Chapter, Poland, Gesellschaft für Informatik, Asociación de Técnicos de Informática, Naukowe Towarzystwo Informatyki Ekonomicznej. Furthermore, the 6th International Symposium Advances in Artificial Intelligence and Applications (AAIA'11) was organized in technical cooperation with: World Federation of Soft Computing, European Neural Networks Society, Poland Chapter of the IEEE Computational Intelligence Society (CIS), Polish Neural Networks Society; while the Workshop on Computer Aspects of Numerical Algorithms (CANA'11) was organized in-cooperation with the Society for Industrial Applied Mathematics.

FedCSIS consisted of the following events (and Proceedings are organized into sections that correspond to each of them):

- 6th International Symposium Advances in Artificial Intelligence and Applications (AAIA'11),
  - International Workshop on Artificial Intelligence in Medical Applications (AIMA'2011)
  - 1st International Workshop on Advances in Semantic Information Retrieval (ASIR'11),
  - Workshop on Computational Optimization (WCO'2011),
- International Workshop on Advances in Business ICT (ABICT'11),
- Advances in Healthcare Information Technologies (HIT 2011)
  - 1st International Workshop on Interoperable Healthcare Systems (IHS'2011)—Challenges, Technologies, and Trends,
  - International Workshop on Ubiquitous Home Healthcare (UHH),
- Computer Aspects of Numerical Algorithms (CANA'11),

- 3rd International Symposium on Services Science,
- Joint Agent-oriented Workshops in Synergy (JAWS 2011)
  - Workshop on Agent Based Computing: from Model to Implementation—VIII (ABC:MI),
  - 5th International Workshop on Multi-Agent Systems and Simulation (MAS&S),
- International Symposium on Multimedia Applications and Processing (MMAAP 2011),
- Risks Awareness and Management through Smart Solutions (RAMSS 2011),
- 3rd Workshop on Advances in Programming Languages (WAPL'11),
- 3rd Workshop on Software Services: Semantic-based Software Services (WoSS'11),

Each of these events had its own Organizing and Program Committee (listed in these Proceedings). We would like to express our warmest gratitude to members of all of them for their hard work in attracting and later refereeing 250 submissions.

FedCSIS was organized under the auspices of Prof. Barbara Kudrycka, Minister of Science and Higher Education and Prof. Michał Kleiber, President of Polish Academy of Sciences. It was sponsored by Ministry of Science and Higher Education and Intel.

***Maria Ganzha**, Conference Co-Chair, Systems Research Institute Polish Academy of Sciences, Warsaw, Poland, and Gdańsk University, Gdańsk, Poland.*

***Leszek Maciaszek**, Conference Co-Chair, Wrocław University of Economics, Wrocław, Poland and Macquarie University, Sydney, Australia.*

***Marcin Paprzycki**, Systems Research Institute Polish Academy of Sciences, Warsaw and Management Academy, Warsaw, Poland.*



# Proceedings of the Federated Conference on Computer Science and Information Systems

September 18–21, 2011. Szczecin, Poland

---

## TABLE OF CONTENTS

---

### 6<sup>TH</sup> INTERNATIONAL SYMPOSIUM "ADVANCES IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS":

---

<b>Call For Papers</b>	<b>1</b>
<b>A Web Statistics based Conflation Approach to Improve Arabic Text Retrieval</b>	<b>3</b>
<i>Farag Ahmed, Andreas Nürnberger</i>	
<b>Improving the predictiveness of ICU medical scales by the method of pairwise comparisons</b>	<b>11</b>
<i>Mohammed Alqarni, Yassen Arabi, Tamar Kakiashvili, Mohammed Khedr, Waldemar W. Koczkodaj, J. Leszek, A. Przelaskowski, K. Rutkowski</i>	
<b>A rough k-means fragile watermarking approach for image authentication</b>	<b>19</b>
<i>Lamiaa M. El Bakrawy, Neveen I. Ghali, Aboul ella Hassanien, Tai-hoon Kim</i>	
<b>Sparse PCA for gearbox diagnostics</b>	<b>25</b>
<i>Anna Bartkowiak, Radosław Zimroz</i>	
<b>CWJess: Implementation of an Expert System Shell for Computing with Words</b>	<b>33</b>
<i>Elham S. Khorasani, Shahram Rahimi, Patel Purvag, Daniel Houle</i>	
<b>Visual Exploration of Cash Flow Chains</b>	<b>41</b>
<i>Jerzy Korczak, Walter Łuszczuk</i>	
<b>Logical Inference in Query Answering Systems Based on Domain Ontologies</b>	<b>47</b>
<i>Juliusz Kulikowski</i>	
<b>Competitive and self-contained gene set analysis methods applied for class prediction</b>	<b>55</b>
<i>Henryk Maciejewski</i>	
<b>Knowledge patterns for conversion of sentences in natural language into RDF graph language</b>	<b>63</b>
<i>Rostislav Miarka, Martin Žáček</i>	
<b>Graph Mining Approach to Suspicious Transaction Detection</b>	<b>69</b>
<i>Krzysztof Michalak, Jerzy Korczak</i>	
<b>Growing Hierarchical Self-Organizing Map for searching documents using visual content</b>	<b>77</b>
<i>Paweł Myszkowski, Bartłomiej Buczek</i>	
<b>Automatic Image Annotation by Image Fragment Matching</b>	<b>83</b>
<i>Mariusz Paradowski, Andrzej Śluzek</i>	
<b>How to predict future in a world of antibody-antigen chromosomes</b>	<b>91</b>
<i>Zbigniew Pliszka, Olgierd Unold</i>	
<b>The Fuzzy Genetic Strategy for Multiobjective Optimization</b>	<b>97</b>
<i>Krzysztof Pytel</i>	

<b>New property for rule interestingness measures</b>	<b>103</b>
<i>Izabela Szczęch, Salvatore Greco, Roman Słowinski</i>	
<b>The Add-Value of Cases on WUM Plans Recommendation</b>	<b>109</b>
<i>Cristina Wanzeller, Orlando Belo</i>	
<b>Problem of website structure discovery and quality valuation</b>	<b>117</b>
<i>Dmitrij Žatuchin</i>	

---

**INTERNATIONAL WORKSHOP ON ARTIFICIAL INTELLIGENCE IN MEDICAL APPLICATIONS:**

---

<b>Call For Papers</b>	<b>123</b>
<b>Identification of Patient Deterioration in Vital-Sign Data using One-Class Support Vector Machines</b>	<b>125</b>
<i>Lei Clifton, David Clifton, Peter Watkinson, Lionel Tarassenko</i>	
<b>Data Mining Research Trends in Computerized Patient Records</b>	<b>133</b>
<i>Payam Homayounfar, Mieczysław Owoc</i>	
<b>A Bezier Curve Approximation of the Speech Signal in the Classification Process of Laryngopathies</b>	<b>141</b>
<i>Krzysztof Pancierz, Jarosław Szkola, Jan Warchol</i>	
<b>Validation of Data Categorization Using Extensions of Information Systems: Experiments on Melanocytic Skin Lesion Data</b>	<b>147</b>
<i>Krzysztof Pancierz, Grzegorz Owsiany, Łukasz Piątek</i>	
<b>Interval based attribute evaluation algorithm</b>	<b>153</b>
<i>Mostafa Salama, Nashwa El-Bendary, Aboul Ella Hassanien, Kenneth Revett, Aly A. Fahmy</i>	
<b>Medical Image Segmentation Using Information Extracted from Deformation</b>	<b>157</b>
<i>Kai Xiao, Neven Ghali, Aboul Ella Hassanien</i>	
<b>Discovering similarities for the treatments of liver specific parasites</b>	<b>165</b>
<i>Pinar Yildirim, Kagan Ceken, Osman Saka</i>	
<b>Reliability Analysis of Healthcare System</b>	<b>169</b>
<i>Elena Zaitseva, Vitaly Levashenko, Miroslav Rusin</i>	

---

**1<sup>ST</sup> INTERNATIONAL WORKSHOP ON ADVANCES IN SEMANTIC INFORMATION RETRIEVAL:**

---

<b>Call For Papers</b>	<b>177</b>
<b>Fuzzy Cognitive Maps Theory for the Political Domain</b>	<b>179</b>
<i>Sameera Alshayji, Nahla Elzant Elkadhi, Zidong Wang</i>	
<b>Building a Model of Disease Symptoms Using Text Processing and Learning from Examples</b>	<b>187</b>
<i>Marek Jaszuk, Grazyna Szostek, Andrzej Walczak, Leszek Puzio</i>	
<b>Query Expansion: Term Selection using the Semantic Relatedness Measure</b>	<b>195</b>
<i>Vitaly Klyuev, Yannis Haralambous</i>	
<b>LTIMEX: Representing the Local Semantics of Temporal Expressions</b>	<b>201</b>
<i>Paweł Mazur, Robert Dale</i>	
<b>Dependency-Based Rules for Grammar Checking with LanguageTool</b>	<b>209</b>
<i>Maxim Mozgovoy</i>	
<b>Preserving pieces of information in a given order in HRR and GA<sub>c</sub></b>	<b>213</b>
<i>Agnieszka Patyk-Łońska</i>	
<b>Some tests on geometric analogues of Holographic Reduced Representations and Binary Spatter Codes</b>	<b>221</b>
<i>Agnieszka Patyk-Łońska, Marek Czachor, Diederik Aerts</i>	

---

**WORKSHOP ON COMPUTATIONAL OPTIMIZATION:**

---

<b>Call For Papers</b>	<b>229</b>
<b>Task Scheduling with Restricted Preemptions</b> <i>Tomasz Baranski</i>	<b>231</b>
<b>A Branch-and-Cut-and-Price Algorithm for a Fingerprint-Template Compression Application</b> <i>Andreas Chwatal, Corinna Thöni, Karin Oberlechner, Günther Raidl</i>	<b>239</b>
<b>Improved asymptotic analysis for SUMT methods</b> <i>Jean-Pierre Dussault</i>	<b>247</b>
<b>Numerical Assessment of Finite Difference Time Domain (FDTD) and Complex-Envelope Alternating-Direction-Implicit Finite-Difference-Time-Domain (CE-ADI-FDTD) Methods</b> <i>Gebriel Gannat</i>	<b>255</b>
<b>Performing Conjoint Analysis within a Logic-based Framework</b> <i>Adrian Giurca, Ingo Schmitt, Daniel Baier</i>	<b>261</b>
<b>Extending the definition of beta-consistent biclustering for feature selection</b> <i>Antonio Mucherino</i>	<b>269</b>

---

**INTERNATIONAL WORKSHOP ON ADVANCES IN BUSINESS ICT:**

---

<b>Call For Papers</b>	<b>275</b>
<b>Formal Verification of Business Processes Represented as Role Activity Diagrams</b> <i>Amelia Badica, Costin Badica</i>	<b>277</b>
<b>Virtualization as an approach in the development of IT system implementation process</b> <i>Iwona Chomiak-Orsa, Wiesława Gryncewicz, Maja Leszczyńska</i>	<b>281</b>
<b>An architecture of a Web recommender system using social network user profiles for e-commerce</b> <i>Damian Fijałkowski, Radostaw Zatoka</i>	<b>287</b>
<b>Geospatial presentation of purchase transactions data</b> <i>Maciej Grzenda, Krzysztof Kaczmarek, Mateusz Kobos, Marcin Luckner</i>	<b>291</b>
<b>Explaining MCDM acceptance: a conceptual model of influencing factors</b> <i>Martina Maida, Konradin Maier, Nikolaus Obwegeser, Volker Stix</i>	<b>297</b>
<b>A Context-Aware Mobile Accessible Electric Vehicle Management System</b> <i>Nils Masuch, Marco Luetzenberger, Sebastian Ahrndt, Axel Hessler</i>	<b>305</b>
<b>NotX Service Oriented Multi-platform Notification System</b> <i>Filip Nguyen, Jaroslav Škrabálek</i>	<b>313</b>
<b>Commonality in Various Design Science Methodologies</b> <i>Lukasz Ostrowski, Markus Helfert</i>	<b>317</b>
<b>A Hybrid Algorithm for Detecting Changes in Diagnostic Signals Received From Technical Devices</b> <i>Tomasz Petech-Pilichowski</i>	<b>321</b>
<b>Adapting Scrum for Third Party Services and Network Organizations</b> <i>Lukasz D. Sienkiewicz, Leszek A. Maciaszek</i>	<b>329</b>
<b>Extending the Descartes Specification Language Towards Process Modeling</b> <i>Joseph Urban, Vinitha Hannah Subburaj, Lavanya Ramamoorthy</i>	<b>337</b>
<b>Influence of search engines on customer decision process</b> <i>Marek Zgódko</i>	<b>341</b>

---

**ADVANCES IN HEALTHCARE INFORMATION TECHNOLOGIES:**

---

<b>1<sup>ST</sup> INTERNATIONAL WORKSHOP ON INTEROPERABLE HEALTHCARE SYSTEMS—CHALLENGES, TECHNOLOGIES, AND TRENDS:</b>	
<b>Call For Papers</b>	<b>345</b>
<b>The Intersection of Clinical Decision Support and Electronic Health Record: A Literature Review</b>	<b>347</b>
<i>Hajar Kashfi</i>	
<b>EMeH: Extensible Mobile Platform for Healthcare</b>	<b>355</b>
<i>Jacek Kobusiński, Maciej Matecki, Krzysztof Stefaniak, Tomasz Biały</i>	
<b>Semantic Interoperability for Infectious Disease Reporting System</b>	<b>363</b>
<i>Murugavell Pandiyan, Osama Elhassan, Zakaria Maamar, Pallikonda Rajasekar</i>	
<b>INTERNATIONAL WORKSHOP ON UBIQUITOUS HOME HEALTHCARE:</b>	
<b>Call For Papers</b>	<b>369</b>
<b>The role of a mobile device in a home monitoring healthcare system</b>	<b>371</b>
<i>Marcin Bajorek, Jędrzej Nowak</i>	
<b>MuSA: a multisensor wearable device for AAL</b>	<b>375</b>
<i>Valentina Bianchi, Ferdinando Grossi, Ilaria De Munari, Paolo Ciampolini</i>	
<b>Intelligent bathroom</b>	<b>381</b>
<i>Adam Bujnowski, Adam Wtorek, Arkadiusz Paliński</i>	
<b>Low-coherence method of hematocrit measurement</b>	<b>387</b>
<i>Małgorzata Jędrzejewska-Szczerska, Marcin Gnyba, Michał Kruczkowski</i>	
<b>Multimodal platform for continuous monitoring of elderly and disabled at home</b>	<b>393</b>
<i>Mariusz Kaczmarek, Jacek Ruminski, Adam Bujnowski</i>	
<b>Design of a wearable sensor network for home monitoring of human behavior</b>	<b>401</b>
<i>Eliasz Kańtoch, Joanna Jaworek, Piotr Augustyniak</i>	
<b>Measuring Pulse Rate with a Webcam—a Non-contact Method for Evaluating Cardiac Activity</b>	<b>405</b>
<i>Magdalena Lewandowska, Jacek Ruminski, Tomasz Kocejko, Jędrzej Nowak</i>	
<b>Pulse pressure velocity measurement—a wearable sensor</b>	<b>411</b>
<i>Mateusz Moderhak, Mariusz Moderhak, Jerzy Wtorek, Bart Truyen</i>	
<b>Analysis of Correlation between Heart Rate and Blood Pressure</b>	<b>417</b>
<i>Artur Polinski, Jacek Kot, Anna Meresta</i>	
<hr/> <b>COMPUTER ASPECTS OF NUMERICAL ALGORITHMS:</b> <hr/>	
<b>Call For Papers</b>	<b>421</b>
<b>The incomplete factorization preconditioners applied to the GMRES(m) method for solving Markov chains</b>	<b>423</b>
<i>Beata Bylina, Jarosław Bylina</i>	
<b>Cache-Aware Matrix Multiplication on Multicore Systems for IPM-based LP Solvers</b>	<b>431</b>
<i>Mujahed Eleyat, Lasse Natvig, Jorn Amundsen</i>	
<b>Object Oriented Model of Generalized Matrix Multiplication</b>	<b>439</b>
<i>Maria Ganzha, Stanislav Sedukhin, Marcin Paprzycki</i>	
<b>Parallel alternating directions algorithm for 3D Stokes equation</b>	<b>443</b>
<i>Ivan Lirkov, Marcin Paprzycki, Maria Ganzha, Paweł Gepner</i>	
<b>GPGPU calculations of gas thermodynamic quantities</b>	<b>451</b>
<i>Igor Mračka, Peter Somora, Tibor Žáčik</i>	



<b>The influence of a matrix condition number on iterative methods' convergence</b>	<b>459</b>
<i>Anna Pызara, Beata Bylina, Jarosław Bylina</i>	
<b>Solving Linear Recurrences on Hybrid GPU Accelerated Manycore Systems</b>	<b>465</b>
<i>Przemysław Stpiczynski</i>	
<b>A multipoint shooting feasible-SQP method for optimal control of state-constrained parabolic DAE systems</b>	<b>471</b>
<i>Krystyn Styczeń, Wojciech Rafajłowicz</i>	
<b>A modified multipoint shooting feasible-SQP method for optimal control of DAE systems</b>	<b>477</b>
<i>Krystyn Styczeń, Paweł Drąg</i>	
<b>On the implementation of stream ciphers based on a new family of algebraic graphs</b>	<b>485</b>
<i>Vasyl Ustimenko, Stanisław Kotorowicz, Urszula Romanczuk</i>	
<b>Implementation of Movie-based Matrix Algorithms on OpenMP Platform</b>	<b>491</b>
<i>Dmitry Vazhenin, Alexander Vazhenin</i>	

---

### 3<sup>RD</sup> INTERNATIONAL SYMPOSIUM ON SERVICES SCIENCE:

---

<b>Call For Papers</b>	<b>495</b>
<b>Learning to Innovate in Distributed Mobile Application Development: Learning Episodes from Tehran and London</b>	<b>497</b>
<i>Neek Alyani, Sara Shirzad</i>	
<b>Configuring services regarding service environment and productivity indicators</b>	<b>505</b>
<i>Michael Becker, Stephan Klingner, Martin Bottcher</i>	
<b>Service quality description—a business perspective</b>	<b>513</b>
<i>Marija Bjekovic, Sylvain Kubicki</i>	
<b>Towards an Interdisciplinary View on Service Science—The Case of the Financial Services Industry</b>	<b>521</b>
<i>Michael Fischbach, Thomas Puschmann, Rainer Alt</i>	
<b>Services Composition Model for Home-Automation peer-to-peer Pervasive Computing</b>	<b>529</b>
<i>Juan A. Holgado-Terriza, Sandra Rodríguez-Valenzuela</i>	
<b>Violation of Service Availability Targets in Service Level Agreements</b>	<b>537</b>
<i>Maurizio Naldi, Loretta Mastroeni</i>	
<b>Orchestration of Service Design and Service Transition</b>	<b>541</b>
<i>Bernd Pfitzinger, Thomas Jestadt</i>	
<b>Service Innovation Capability: Proposing a New Framework</b>	<b>545</b>
<i>Jens Pöppelbuß, Ralf Plattfaut, Kevin Ortbach, Andrea Malsbender, Matthias Voigt, Björn Niehaves, Jörg Becker</i>	
<b>A Framework for Comparing Cloud-Environments</b>	<b>553</b>
<i>Rainer Schmidt</i>	

---

### JOINT AGENT-ORIENTED WORKSHOPS IN SYNERGY:

---

<b>Call For Papers</b>	<b>557</b>
<b>WORKSHOP ON AGENT BASED COMPUTING: FROM MODEL TO IMPLEMENTATION—VIII:</b>	
<b>Call For Papers</b>	<b>559</b>
<b>A methodology for developing component-based agent systems focusing on component quality</b>	<b>561</b>
<i>George Eleftherakis, Petros Kefalas, Evangelos Kehris</i>	

<b>Monitoring Building Indoors through Clustered Embedded Agents</b>	<b>569</b>
<i>Giancarlo Fortino, Antonio Guerrieri</i>	
<b>Multiagent Distributed Grid Scheduler</b>	<b>577</b>
<i>Victor Korneev, Dmitry Semenov, Andrey Kiselev, Boris Shabanov, Pavel Telegin</i>	
<b>Tuning Computer Gaming Agents using Q-Learning</b>	<b>581</b>
<i>Purvag Patel, Norman Carver, Shahram Rahimi</i>	
<b>Developing intelligent bots for the Diplomacy game</b>	<b>589</b>
<i>Sylwia Polberg, Marcin Paprzycki, Maria Ganzha</i>	
<b>Computing Equilibria for Constraint-based Negotiation Games with Interdependent Issues</b>	<b>597</b>
<i>Mihnea Scafes, Costin Badica</i>	
<b>Agent-Oriented Knowledge Elicitation for Modeling the Winning of "Hearts and Minds"</b>	<b>605</b>
<i>Inna Shvartsman, Kuldar Taveter</i>	
<b>5<sup>TH</sup> INTERNATIONAL WORKSHOP ON MULTI-AGENT SYSTEMS AND SIMULATION:</b>	
<b>Call For Papers</b>	<b>609</b>
<b>Multi Agent Simulation for Decision Making in Warehouse Management</b>	<b>611</b>
<i>Massimo Cossentino, Carmelo Lodato, Lopes Salvatore, Patrizia Ribino</i>	
<b>A Multi-Agent Architecture for Simulating and Managing Microgrids</b>	<b>619</b>
<i>Massimo Cossentino, Carmelo Lodato, Salvatore Lopes, Marcello Pucci, Gianpaolo Vitale, Maurizio Cirrincione</i>	
<b>Agent.GUI: A Multi-agent Based Simulation Framework</b>	<b>623</b>
<i>Christian Derksen, Cherif Branki, Rainer Unland</i>	
<b>Minority Game: the Battle of Adaptation, Intelligence, Cooperation and Power</b>	<b>631</b>
<i>Akihiro Eguchi, Hung Nguyen</i>	
<b>Towards a Generic Testing Framework for Agent-Based Simulation Models</b>	<b>635</b>
<i>Onder Gurcan, Oguz Dikenelli, Carole Bernon</i>	
<b>Modeling Agent Behavior Through Online Evolutionary and Reinforcement Learning</b>	<b>643</b>
<i>Robert Junges, Franziska Klügl</i>	
<b>Visualizing Agent-Based Simulation Dynamics in a CAVE—Issues and Architectures</b>	<b>651</b>
<i>Athanasia Louloudi, Franziska Klügl</i>	
<b>SimConnector: An Approach to Testing Disaster-Alerting Systems Using Agent Based Simulation Models</b>	<b>659</b>
<i>Muaz Niazi, Qasim Siddique, Amir Hussain</i>	
<b>A Chemical Inspired Simulation Framework for Pervasive Services Ecosystems</b>	<b>667</b>
<i>Danilo Pianini, Sara Montagna, Mirko Viroli</i>	
<b>BioMASS: a Biological Multi-Agent Simulation System</b>	<b>675</b>
<i>Candelaria Sansores, Flavio Reyes, Hector Gomez, Juan Pavon, Luis Calderon</i>	
<hr/>	
<b>INTERNATIONAL SYMPOSIUM ON MULTIMEDIA APPLICATIONS AND PROCESSING:</b>	
<hr/>	
<b>Call For Papers</b>	<b>683</b>
<b>Robust Digital Watermarking System for Still Images</b>	<b>685</b>
<i>Sergey Anfinogenov, Valery Korzhik, Guillermo Morales-Luna</i>	
<b>Estimating Topographic Heights with the StickGrip Haptic Device</b>	<b>691</b>
<i>Tatiana V. Evreinova, Grigori Evreinov, Roope Raisamo</i>	

<b>Image Indexing by Spatial Relationships between Salient Objects</b>	<b>699</b>
<i>Eugen Ganea, Marius Brezovan</i>	
<b>From icons perception to mobile interaction</b>	<b>705</b>
<i>Chrysoula Gatsou, Anastasios Politis, Dimitrios Zevgolis</i>	
<b>Automatic Speech Recognition for Polish in a Computer Game Interface</b>	<b>711</b>
<i>Artur Janicki, Dariusz Wawer</i>	
<b>Classification of Learners Using Linear Regression</b>	<b>717</b>
<i>Cristian Mihaescu</i>	
<b>Data Centered Collaboration in a Mobile Environment</b>	<b>723</b>
<i>Maciej Panka, Piotr Bala</i>	
<b>Computerized Three-Dimensional Craniofacial Reconstruction from Skulls Based on Landmarks</b>	<b>729</b>
<i>Leticia Carnero Pascual, Carmen Lastres Redondo, Belen Rios Sanchez, David Garrido Garrido, Asuncion Santamaria Galdon</i>	
<b>DCFMS: A Chunk-Based Distributed File System for Supporting Multimedia Communication</b>	<b>737</b>
<i>Cosmin Marian Poteras, Constantin Petrisor, Mihai Mocanu, Cristian Marian Mihaescu</i>	
<b>Automatic classification of gestures: a context-dependent approach</b>	<b>743</b>
<i>Mario Refice, Michelina Savino, Michele Caccia, Michele Adduci</i>	
<b>Concurrency control for a Multimedia Database System</b>	<b>751</b>
<i>Cosmin Stoica Spahiu</i>	
<b>Automated annotation system for natural images</b>	<b>755</b>
<i>Liana Stanescu</i>	
<b>Fuzzy UML and Petri Nets Modeling Investigations on the Pollution Impact on the Air Quality in the Vicinity of the Black Sea Constanta Romanian Resort</b>	<b>763</b>
<i>Elena-Roxana Tudoroiu, Adina Astilean, Tiberiu Letia, Gabriela Neacsu, Zoltan Maroszy, Nicolae Tudoroiu</i>	
<b>Pass-Image Authentication Method Tolerant to Video-Recording Attacks</b>	<b>767</b>
<i>Hirakawa Yutaka, Hiroyuki Take, Kazuo Ohzeki</i>	
<hr/>	
<b>RISKS AWARENESS AND MANAGEMENT THROUGH SMART SOLUTIONS:</b>	
<hr/>	
<b>Call For Papers</b>	<b>775</b>
<b>Enhancing DNS Security using Dynamic Firewalling with Network Agents</b>	<b>777</b>
<i>Joao Afonso, Pedro Veiga</i>	
<b>Enhanced CakES representing Safety Analysis results of Embedded Systems</b>	<b>783</b>
<i>Yasmin I. Al-Zokari, Daniel Schneider, Dirk Zeckzer, Liliana Guzman, Yarden Livnat, Hans Hagen</i>	
<b>Integrated management of risk information</b>	<b>791</b>
<i>José Barateiro, José Borbinha</i>	
<hr/>	
<b>3<sup>RD</sup> WORKSHOP ON ADVANCES IN PROGRAMMING LANGUAGES:</b>	
<hr/>	
<b>Call For Papers</b>	<b>799</b>
<b>Implementation of a Domain-Specific Language EasyTime using LISA Compiler Generator</b>	<b>801</b>
<i>Iztok Jr. Fister, Marjan Mernik, Iztok Fister, Dejan Hrnčič</i>	
<b>Using Aspect-Oriented State Machines for Resolving Feature Interactions</b>	<b>809</b>
<i>Tom Dinkelaker, Mohammed Erradi</i>	
<b>Domain-Specific Modeling in Document Engineering</b>	<b>817</b>
<i>Verislav Djukic, Ivan Luković, Aleksandar Popovic</i>	

<b>A MOF based Meta-Model of IIS* Case PIM Concepts</b>	<b>825</b>
<i>Milan Čeliković, Ivan Luković, Slavica Aleksić, Vladimir Ivančević</i>	
<b>Memory Safety and Race Freedom in Concurrent Programming Languages with Linear Capabilities</b>	<b>833</b>
<i>Niki Vazou, Michalis Papakyriakou, Nikolaos Papaspyrou</i>	
<b>Decomposition of SBQL Queries for Optimal Result Caching</b>	<b>841</b>
<i>Piotr Cybula, Kazimierz Subieta</i>	
<b>Automated Conversion of ST Control Programs to Why for Verification Purposes</b>	<b>849</b>
<i>Jan Sadolewski</i>	
<b>Implementing Attribute Grammars Using Conventional Compiler Construction Tools</b>	<b>855</b>
<i>Daniel Rodriguez Cerezo, Antonio Sarasa Cabezuelo, Jose Luis Sierra Rodriguez</i>	
<b>The embedded left LR parser</b>	<b>863</b>
<i>Bostjan Slivnik</i>	
<b>Nonlinear Tree Pattern Pushdown Automata</b>	<b>871</b>
<i>Jan Travnicek, Jan Janoušek, Borivoj Melichar</i>	
<b>A Type and Effect System for Implementing Functional Arrays with Destructive Updates</b>	<b>879</b>
<i>Georgios Korfiatis, Michalis Papakyriakou, Nikolaos Papaspyrou</i>	
<b>Checking the Conformance of Grammar Refinements with Respect to Initial Context-Free Grammars</b>	<b>887</b>
<i>Bryan Temprado Battad, Antonio Sarasa Cabezuelo, Jose Luis Sierra Rodriguez</i>	
<b>Identification of Patterns through Haskell Programs Analysis</b>	<b>891</b>
<i>Jan Kollar, Sergej Chodarev, Emilia Pietrikova, Lubomir Wassermann</i>	
<b>Computer Language Notation Specification through Program Examples</b>	<b>895</b>
<i>Miroslav Sabo, Jaroslav Porubán, Dominik Lakatoš, Michaela Kreutzová</i>	
<b>Tree Indexing by Pushdown Automata and Repeats of Subtrees</b>	<b>899</b>
<i>Tomas Flouri, Jan Janoušek, Borivoj Melichar, Costas Iliopoulos, Solon Pissis</i>	
<b>Subtree Oracle Pushdown Automata for Ranked and Unranked Ordered Trees</b>	<b>903</b>
<i>Martin Plicka, Jan Janoušek, Borivoj Melichar</i>	
<b>Semi-Automatic Component Upgrade with RefactoringNG</b>	<b>907</b>
<i>Zdeněk Troníček</i>	
<b>Extension of Iterator Traits in the C++ Standard Template Library</b>	<b>911</b>
<i>Norbert Pataki, Zoltán Porkoláb</i>	
<hr/>	
<b>3<sup>RD</sup> WORKSHOP ON SOFTWARE SERVICES: SEMANTIC-BASED SOFTWARE SERVICES:</b>	
<hr/>	
<b>Call For Papers</b>	<b>915</b>
<b>Search-Based Testing, the Underlying Engine of Future Internet Testing</b>	<b>917</b>
<i>Arthur Baars, Kiran Lakhota, Tanja E. J. Vos, Joachim Wegener</i>	
<b>Testing and Remote Maintenance of Real Future Internet Scenarios, Towards FITTEST and FastFix Advanced Software Engineering</b>	<b>925</b>
<i>Alessandra Bagnato, Anna Esparcia Alcazar, Tanja E. J. Vos, Beatriz Marin, José Oliver Murillo, Salvador I. Folgado, Auxiliadora Carlos Alberola</i>	
<b>A Neural Model for Ontology Matching</b>	<b>933</b>
<i>Emil Stefan Chifu, Ioan Alfred Letia</i>	
<b>An Adaptive Virtual Machine Replication Algorithm for Highly-Available Services</b>	<b>941</b>
<i>Adrian Coleșa, Mihai Bica</i>	

<b>Service Modelling for Internet of Things</b>	<b>949</b>
<i>Suparna De, Payam Barnaghi, Martin Bauer, Stefan Meissner</i>	
<b>Self-Healing Approach in the FastFix Project</b>	<b>957</b>
<i>Benoit Gaudin, Mike Hinchey</i>	
<b>Autonomic Execution of Computational Workflows</b>	<b>965</b>
<i>Tomasz Haupt, Nitin Sukhija, Igor Zhuk</i>	
<b>An Analysis of mOSAIC ontology for Cloud Resources annotation</b>	<b>973</b>
<i>Francesco Moscato, Rocco Aversa, Beniamino Martino, Teodor-Florin Fortis, Victor Munteanu</i>	
<b>Multi-Agent Architecture for Solving Nonlinear Equations Systems in Semantic Services Environment</b>	<b>981</b>
<i>Victor Ion Munteanu, Cristina Mindruta, Viorel Negru, Calin Sandru</i>	
<b>Cloud-based Assistive Technology Services</b>	<b>985</b>
<i>Ane Murua, Igor González, Elena Gómez-Martínez</i>	
<b>Semantic P2P Search engine</b>	<b>991</b>
<i>Ilya Rudomilov, Ivan Jelinek</i>	
<b>Hybrid Immune-inspired Method for Selecting the Optimal or a Near-Optimal Service Composition</b>	<b>997</b>
<i>Ioan Salomie, Monica Vlad, Viorica Rozina Chifu, Cristina Bianca Pop</i>	
<b>Dynamic Consolidation Methodology for Optimizing the Energy Consumption in Large Virtualized Service Centers</b>	<b>1005</b>
<i>Cioara Tudor, Ionut Anghel, Ioan Salomie, Daniel Moldovan, Georgiana Copil, Pierluigi Plebani</i>	



# 6<sup>th</sup> International Symposium Advances in Artificial Intelligence and Applications

CELEBRATING 65TH BIRTHDAY OF PROFESSOR BOHDAN MACUKOW

THE AAIA'11 will bring researchers, developers, practitioners, and users to present their latest research, results, and ideas in all areas of artificial intelligence. We hope that theory and successful applications presented at the AAIA'10 will be of interest to researchers and practitioners who want to know about both theoretical advances and latest applied developments in Artificial Intelligence. As such AAIA'11 will provide a forum for the exchange of ideas between theoreticians and practitioners to address the important issues.

Papers related to theories, methodologies, and applications in science and technology in this theme are especially solicited. Topics covering industrial issues/applications and academic research are included, but not limited to:

- Knowledge management
- Decision Support System
- Approximate Reasoning
- Fuzzy modeling and control
- Data Mining
- Web Mining
- Machine learning
- Combining multiple knowledge sources in an integrated intelligent system
- Neural Networks
- Evolutionary Computation
- Nature Inspired Methods
- Natural Language processing
- Image processing and understanding (interpretation)
- Applications in Bioinformatics
- Hybrid Intelligent Systems
- Granular Computing
- Architectures of intelligent systems
- Robotics
- Real-world applications of Intelligent Systems

## PROGRAM COMMITTEE

**Janos Abonyi**, University of Pannonia, Hungary

**Ajith Abraham**, Machine Intelligence Research Labs, USA

**Hans Jorgen Andersen**, Aalborg University, Denmark

**Jaroslav Arabas**, Warsaw University of Technology, Poland

**Mohammad Azzeh**, Applied Science Private University, Jordan

**Valentina Emilia Balas**, Aurel Vlaicu University of Arad, Romania

**Anna Bartkowiak**, Wrocław University, Poland

**Roberto Basili**, University of Roma, Italy

**Bernhard Beckert**, University of Koblenz-Landau, Germany

**Francesco Bergadano**, University of Torino, Italy

**Jerzy Blaszczynski**, Poznan University of Technology, Poland

**Sheryl Brahnam**, Missouri State University, USA

**Minhua Eunice Ma**, The Glasgow School of Art, UK

**Min Cai**, Oklahoma State University

**Giovanna Castellano**, University of Bari, Italy

**Ryszard Choras**, Institute of Telecommunications, Poland

**Alfredo Cuzzocrea**, University of Calabria, Italy

**Jeremiah Da Deng**, University of Otago, New Zealand

**Krzysztof Dembczyński**, Poznan University of Technology, Poland

**Włodzisław Duch**, Nicolaus Copernicus University, Poland

**Krzysztof Goczyla**, Gdańsk University of Technology, Poland

**Zdzisław Hippe**, University of Information Technology and Management in Rzeszów, Poland

**Elżbieta Hudyma**, Wrocław University of Technology, Poland

**Lim-Joo Hwee**, Nanyang Technological University, Singapore

**Giancarlo Iannizzotto**, University of Messina, Italy

**Jerzy W. Jaromczyk**, University of Kentucky, USA

**Piotr Jędrzejowicz**, Gdynia Maritime University, Poland

**Jerzy Józefczyk**, Wrocław University of Technology, Poland

**Janusz Kacprzyk**, Polish Academy of Sciences, Poland

**Radosław Katarzyniak**, Wrocław University of Technology, Poland

**Przemysław Kazienko**, Wrocław University of Technology, Poland

**Etienne Kerre**, University of Gent, Belgium

**Jacek Kluska**, Rzeszów University of Technology, Poland

**Waldemar Koczkodaj**, Laurentian University, Canada

**Mieczysław M. Kokar**, Northeastern University, USA

**Yiannis Kompatsiaris**, Informatics and Telematics Institute, Greece

**Józef Korbicz**, University of Zielona Góra, Poland

**Adam Krzyzak**, Concordia University, Canada

**Juliusz Lech Kulikowski**, Polish Academy of Sciences, Poland

**Rory Lewis**, University of Colorado at Colorado Springs, USA

**Penousal Machado**, University of Coimbra, Portugal

**Jacek Mandziuk**, Warsaw University of Technology, Poland

**Victor Marek**, University of Kentucky, USA

**Radek Matousek**, Brno University of Technology, Czech Republic

**Zbigniew Michalewicz**, University of Adelaide, Australia

**Łukasz Mirosław**, Wrocław University of Technology, Poland

**Pawel Myszkowski**, Wrocław University of Technology, Poland

**Ngoc-Thanh Nguyen**, Wrocław University of Technology, Poland

**Marek Ogiela**, AGH University of Science and Technology, Poland

**Mariusz Paradowski**, Wrocław University of Technology, Poland

**Witold Pedrycz**, University of Alberta, Canada

**Marco Porta**, University of Pavia, Italy

**Tapani Raiko**, Aalto University, Finland

**Sheela Ramanna**, University of Winnipeg, Canada

**Zbigniew Ras**, University of North Carolina, USA

**Jan Rauch**, University of Economics, Prague, Czech Republic

**Izabela Rejer**, West Pomeranian University of Technology, Poland

**Paolo Rosso**, Universidad Politécnica Valencia, Spain

**Khalid Saeed**, AGH University of Science and Technology, Poland

**Abdel-Badeeh Salem**, Ain Shams University, Egypt

**Jerzy Sas**, Wrocław University of Technology, Poland

**Mika Sato-Ilic**, University of Tsukuba, Japan

**Christelle Scharff**, Pace University, USA

**Lothar Schmitt**, University of Aizu, Japan

**Franciszek Seredynski**, Polish Academy of Sciences, Poland

**Zhongzhi Shi**, Chinese Academy of Sciences, China

**Andrzej Sluzek**, Nanyang Technological University, Singapore

**Dipti Srinivasan**, National University of Singapore, Singapore

**Jerzy Stefanowski**, Poznan University of Technology, Poland

**Siergey Subbotin**, Zaporozhye National Technical University, Ukraine

**Piotr Szczepaniak**, Technical University of Lodz, Poland

**Stan Szpakowicz**, University of Ottawa, Canada

**Jerzy Świątek**, Wrocław University of Technology, Poland

**Ryszard Tadeusiewicz**, AGH University of Science and Technology, Poland

**Paul Trundle**, University of Bradford, UK

**Li-Shiang Tsay**, North Carolina A&T State University, USA

**Izabela Szczęch**, Poznan University of Technology, Poland

**Josef Tvrdik**, University of Ostrava, Czech Republic

**Angelina Tzacheva**, University of South Carolina, USA

**Anita Wasilewska**, Stony Brook University, NY, USA

**Daniela Zaharie**, West University of Timisoara, Romania

**Wojciech Ziarko**, University of Regina, Canada

**Djamel Zighed**, Université Lumière Lyon 2, France

**Jacek Zurada**, University of Louisville, USA

#### ORGANIZING COMMITTEE

**Halina Kwasnicka**, **Urszula Markowska-Kaczmar**, Wrocław University of Technology, Poland



# A Web Statistics based Conflation Approach to Improve Arabic Text Retrieval

Farag Ahmed

Data & Knowledge Engineering Group  
Faculty of Computer Science  
Otto-von-Guericke-University of Magdeburg  
Email: farag.ahmed@ovgu.de

Andreas Nürnberger

Data & Knowledge Engineering Group  
Faculty of Computer Science  
Otto-von-Guericke-University of Magdeburg  
Email: andreas.nuernberger@ovgu.de

**Abstract**—We present a language independent approach for conflation that does not depend on predefined rules or prior knowledge of the target language. The proposed unsupervised method is based on an enhancement of the pure  $n$ -gram model that is used to group related words based on a revised string-similarity measure. In order to detect and eliminate terms that are created by this process, but that are most likely not relevant for the query (“noisy terms”), an approach based on mutual information scores computed based on web statistical co-occurrences data is proposed. Furthermore, an evaluation of this approach is presented.

## I. INTRODUCTION

ARABIC is a Semitic language that is based on the Arabic alphabet containing 28 letters. Its basic feature is that most of its words are built up from, and can be analyzed down to common roots. The exceptions to this rule are common nouns and particles. Arabic is a highly inflectional language with 85% of words derived from trilateral roots. Nouns and verbs are derived from a closed set of around 10,000 roots [1]. Arabic has three genders, feminine, masculine, and neuter; and three numbers, singular, dual, and plural.

The specific characteristics of Arabic morphology make the Arabic language particularly difficult for developing natural language processing methods for Information Retrieval (IR). Arabic is different from English and other Indo-European languages with respect to a number of important aspects: words are written from right to left; it is mainly a consonantal language in its written forms, i.e., it excludes vowels; its two main parts of speech are the verb and the noun in that word order, and these consist, for the main part, of trilateral roots (three consonants forming the basis of noun forms that are derived from them); it is a morphologically complex language, in that it provides flexibility in word formation: as briefly mentioned above, complex rules govern the creation of morphological variations, making it possible to form hundreds of words from one root [2]. Furthermore, the letter shapes are changeable in form, depending on the location of the letter at the beginning, middle or at the end of the word.

One of the main problems we face when indexing and retrieving unstructured text is the variations in word forms. These variations result from the morphological variants, for example in English, verb forms like walk, walked, walking. In the Arabic language, the variations are even more abundant

and a word can sometimes be represented by more than 100 different forms. These variation in word forms results from the fact that Arabic nouns and verbs are heavily prefixed. The definite article *al* “the” is always attached to nouns, and many conjunctions and prepositions are also attached as prefixes to nouns and verbs, hindering the retrieval of morphological variants of words. In Table I some word form variations for the word “student” is presented in order to clarify this issue. The absence of these word form variations in the user query causes a loss of vast amounts of retrieved information. One way to tackle such problems are conflation methods. Conflation is a general term for all processes of merging together nonidentical words that refer to the same principal concept, i.e., merging words that belong to the same meaning class. The primary goal of conflation is to allow matching of different variants of the same word. In natural language processing, conflation is the process of merging or lumping together nonidentical words that refer to the same principal concept. In the context of information retrieval (IR), conflation has a more restricted meaning and usually refers to the process of grouping together morphological variants of the same or related words [3]. Since the variants have similar semantics, it is possible to consider them as equivalent for the purpose of the retrieval tasks. Applying conflation approaches in morphologically complex languages, such as Arabic, improves the retrieval effectiveness and frees the users from taking into account all variants of the same word. Therefore, conflation approaches can be quite beneficial in many fields such as information retrieval and word-processing systems. In order to solve or at least alleviate some of the problems raised by a high inflectional morphology, the stem of the word need to be detected. There are two, widely used, stemming approaches: First, approaches that are language dependent and designed to handle morphological variants. In stemming, morphological variants are reduced to common basic form called root, and second, string-similarity approaches i.e ( $n$ -gram), which are (usually) language independent and designed to handle all types of word variants. In this paper, the proposed approach is based on the enhancement of the  $n$ -gram pure approach therefore we will focus in describing the  $n$ -gram approach in more detail with respect to the Arabic language.

TABLE I  
WORD FORM VARIATIONS FOR طالب *ālb* (STUDENT).

Feminine	Masculine	English
طالبة <i>ālbh</i>	طالب <i>ālb</i>	student
الطالبة <i>ālālbh</i>	الطالب <i>ālālb</i>	the student
طالبتان <i>ālbīān</i>	طالبان <i>ālbān</i>	(two) students(dual)
بِطالبة <i>biālbh</i>	بِطالب <i>biālb</i>	by student
بِطالبة <i>bālbh</i>	بِطالب <i>bālb</i>	by the student
وطالبة <i>wālbh</i>	وطالب <i>wālb</i>	and student
والطالبة <i>wālālbh</i>	والطالب <i>wālālb</i>	and the student
إِطالبة <i>īālbh</i>	إِطالب <i>īālb</i>	to the, for a student
إِطالبة <i>īālbhā</i>	إِطالبة <i>īālbh</i>	his student
إِطالبة <i>īālbhā'</i>	إِطالبة <i>īālbhā</i>	her student
إِطالبة <i>īālbāh</i>	إِطالبة <i>īālbh</i>	his students
إِطالبة <i>īālbāhā'</i>	إِطالبة <i>īālbhā</i>	her students

## II. CONFLATION TECHNIQUES

Conflation algorithms can be categorized into four main classes: affix removal, table lookup, successor variety, and  $n$ -gram [4]. Affix-removal algorithms reduce a word to its morphological root or a stable stem by stripping off suffixes and prefixes in order to determine the stem. They are the most popular group of conflation algorithms, mainly due to the work of [5] and [6]. In the table-lookup approach, all desired stems for a particular surface-form word are stored in a table. Therefore, this approach can be implemented in a computationally efficient manner, since no word transformation is needed. However, one of the main drawbacks is that due to its manual creation usually not all words and word forms can be covered and thus table-lookup approaches are in most cases domain-dependent. The successor-variety approach was first introduced by Hafer and Weiss (1974) [7]. In successor variety, a lexical text is segmented into stems and affixes. The method uses statistical properties-successor and predecessor variety counts-of a corpus, in order to identify the root [8]. The idea is to count the number of different letters encountered after (or before, respectively), a part of a word and to compare it to the counts before and after that position. Morpheme boundaries are then likely to occur at sudden peaks or increases of that value [9].

In the following, we describe two main approaches (Stemmer and  $n$ -gram) which are used to solve or at least alleviate some of the problems raised by a high inflectional morphology.

### A. Stemmer Approaches

In information retrieval systems stemming is used to reduce variant word forms to common roots and thereby improve the ability of the system to match query and document vocabulary [10]. Although stemming has been studied mainly for English, stemming approaches have also been developed for several other languages .e.g., Latin [11], Indonesian [12], Swedish [13], Dutch [14], German [15] and Arabic [16]. There are three main types of approaches for stemming, dictionary-based, rule-based, and statistical-based (mainly  $n$ -gram based) approaches [17].

*Dictionary-based approaches* provide very good results at the cost of high development efforts for the dictionary. The dictionary contains all known words with their inflection forms. The main weakness for this approach is the missing words in the dictionary which would not be recognized by the system for stemming. Another weakness is the inability of this method to stem inert names and foreign words. Also the need to process a large dictionary during runtime can result in high requirements for storage space and processing time. The closest Arabic equivalent for this kind of stemmer is the *root-based stemmer* for Arabic [18] which is based on extracting the root of a given Arabic surface word by stripping off all attached prefix and/or suffix then attempt to extract the root of it. Several morphological analyzers were developed based on this concept [19], [18]. The weaknesses of this stemmer is that the construction of the corresponding dictionaries or rules is a tedious and labor-intensive task due to the result of the morphology complexity of Arabic language. Another problem is that only some small linguistic resources are available for Arabic language. The second type are the *rule-based approaches*. They are based on set of predefined conditions rules. The most well known stemmer of this type is Porter stemmer [5]. The main weakness for this stemmer is that building the rules for the arbitrary language is time consuming. Furthermore, there is a need for experts with linguistic knowledge in that particular language. The Arabic equivalent for this is the *Light stemmer* [16]. Unlike English, both prefixes and suffixes need to be removed for effective stemming. it is based on stripping off prefix and suffix from the word, it uses predefined list of prefix and suffix, it is simply stripping off prefix and/or suffix without any further processing in the rest of the stemmed word [20], [16]. The weakness of this stemmer is that the stripping off prefixes or suffix in Arabic is a not an easy task. Removing them can lead to unexpected results, as many words start with one letter or more which can mistakenly assumed to be prefix or suffix.

## III. $n$ -GRAM

The main idea of  $n$ -gram-based approaches, which groups together words that contain identical character substrings of length  $n$ , called  $n$ -grams, is that the character structure of the word can be used to find semantically similar words and word variants.  $n$ -gram, as a conflation technique, differs from stemmers in that they do not require language knowledge, predefined rules, or a vocabulary database. Furthermore,  $n$ -gram approaches take into account misspelled and transliterated words <sup>1</sup>. For example, Table II shows 15 different spellings for the name Condoleezza; four of them were found in the same news web site ( 'CNN-Arabic') <sup>2</sup>.

### A. $n$ -gram and Arabic Text

Over the last years, several studies, which explore the use of  $n$ -grams for processing Arabic text, have been performed.

<sup>1</sup>Transliteration is the process of converting one orthography from one language into another.

<sup>2</sup><http://arabic.cnn.com/> Retrieved on 01/10/2010, www.Google.com

TABLE II  
MULTIPLE SPELLINGS FOR THE NAME "CONDOLEEZZA".

S/N	Transliteration	Web Occ.	Comments
1	كونداليزا <i>kwndālyzā</i>	3.000.000	CNN
2	كوندوليزا <i>kwndwlyzā</i>	197.000	CNN
3	كوندليزا <i>kwndlyzā</i>	51.100	CNN
4	كونداليسا <i>kwndālyśā</i>	26.300	
5	كوندوليسا <i>kwndwlyśā</i>	26.200	CNN
6	كاندوليزا <i>kāndwlyzā</i>	12.700	
7	كانداليزا <i>kāndālyzā</i>	2.310	
8	كانداليزا <i>kāndālyzā</i>	1.530	
9	كونداليزة <i>kwndālyzh</i>	491	
10	كندليسا <i>kndlyśā</i>	344	
11	كونداليزه <i>kwndālyzh</i>	195	
12	كانداليسا <i>kāndālyśā</i>	144	
13	كانداليسا <i>kāndālyśā</i>	9	
14	كونداليسة <i>kwndālysh</i>	9	
15	كوندليسي <i>kwndlyśy</i>	4	

Mayfield et al. (2001) have found that  $n$ -grams work well in many languages. Furthermore, they investigated the use of character  $n$ -grams for Arabic retrieval in TREC-2001 and found that  $n$ -grams of length 4 were most effective [21]. Darwish and Oard examined multiple tokenization strategies for retrieval of scanned Arabic documents. They found that  $n$ -grams of size  $n = 3$  or  $n = 4$  are well suited to Arabic document retrieval [22]. Mustafa (2004) assessed the overall performance of two  $n$ -gram techniques that he called conventional and hybrid. In his results, Mustafa pointed out that the hybrid approach outperforms the conventional approach [23]. However, all of the previous studies rely on studying the use of  $n$ -gram on the Arabic text based on the following aspects: The effectiveness of  $n$ -gram size and assessing the performance of existing  $n$ -gram approaches. None of the prior studies attempt to modify the pure  $n$ -gram model, so that it also considers language characteristics, while computing the similarity score, in order to improve its performance. Ghaoui et al. (2005) investigated a new morphological class based language model. They used the Morphological rules to derive the different words in a class from their stem. Furthermore, a linear interpolation between the  $n$ -gram model and the morphological model has been evaluated. In their experiments they pointed out that morphological class-based model yields poor performance compared to a classical trigram [24]. The performance of the  $n$ -gram in Arabic text has been studied by many researchers. For example, Abu-Salem (2004) found that all of the proposed  $n$ -gram methods outperform the Word, Stem, and Root index methods [25]. We would like to emphasize again, that none of the prior studies attempted to modify the pure  $n$ -gram model, such that it also considers language characteristics while computing the similarity score, in order to improve its performance. All of the previous studies considered only to investigate the performance on Arabic text based on the effectiveness of  $n$ -gram size using existing  $n$ -

gram approaches. Due to the mentioned insufficiencies of the existing approaches, we propose a "revised"  $n$ -gram algorithm that makes it possible to handle one-character infixes, prefixes, and suffixes, which are frequent in Arabic. The proposed method obtained superior results on a large newspaper corpus.

### B. Computing Similarity Scores Based on $n$ -grams

The  $n$ -gram model can be used to compute the similarity between two strings by counting the number of similar  $n$ -grams they share. The more similar  $n$ -grams between two strings exist, the more similar they are. Based on this idea the similarity coefficient can be derived. The similarity coefficient  $\delta$  is defined by the following equation:

$$\delta = \frac{|\alpha \cap \beta|}{|\alpha \cup \beta|} \quad (1)$$

where  $\alpha$  and  $\beta$  are the  $n$ -gram sets for two words  $a$  and  $b$  to be compared.  $\alpha \cap \beta$  denotes the number of similar  $n$ -grams in  $\alpha$  and  $\beta$ , and  $\alpha \cup \beta$  denotes the number of unique  $n$ -grams in the union of  $\alpha$  and  $\beta$ .

## IV. THE PROPOSED APPROACH

We successfully used our revised  $n$ -gram approach for the conflation task in [26]. However, the revised  $n$ -gram approach in some cases expanded the user query with terms not relevant for the query ("noisy terms"). Here in this paper, we propose an approach, based on computed mutual information scores, based on web statistical co-occurrences data, in order to detect and eliminate such noisy terms.

In the following, we describe first in Section IV-A our algorithm based on the enhancement of the  $n$ -gram model, in order to expand the user query with relevant terms; then in Section IV-B, the approach to eliminate any potentially generated noisy terms based on mutual information scores computed on corpora or web based co-occurrence statistics is presented. The  $n$ -gram based approach assumes two strings are alike based only on a string similarity comparison: the more  $n$ -grams existing between two strings, the more similar they are. However, there are many words that are have a very similar text pattern but a quite different meaning. Therefore, we improved our  $n$ -gram approach by eliminating such noisy terms that could have been generated. This is done by computing the cohesion score between all revised  $n$ -gram generated expanded terms using the mutual information ( $MI$ ) measure. The term/terms that have a lower  $MI$  score than the  $MI$  score mean for all expanded terms can be considered as noisy term/terms and thus will be eliminated.

### A. Revised $n$ -gram Approach

Arabic nouns and verbs are heavily prefixed and suffixed as described in the first section. As a result, it is possible to have words with different lengths that share the same principal concept. Furthermore, the pure  $n$ -gram based approach to compute the similarity coefficient as described above (see Eq. (1)) does not consider the order of the  $n$ -grams in the target word [27]. This increases the probability that the matching

score between two strings will be higher even though they do not share the same concept. Therefore, we revised the computation of the similarity between words to take these two aspects into account. For simplicity, we describe our algorithm for  $n = 2$  (bigrams). However, the approach can be applied for trigrams and  $n$ -grams with  $n > 3$  as well. We define bigrams of words by their respective position in the word  $w_{i,i+(n-1)}$  where  $i$  defines the position of the first letter and  $i + (n - 1)$  the position of the last letter of the considered  $n$ -gram. Thus, the last possible position of an  $n$ -gram, in a word, is defined by  $j = |w| - n + 1$  where  $|w|$  defines the length of the word. In order to deal with the first and second aspect mentioned above, we define a window of  $n$ -grams of the target candidate words that should be compared, i.e., while in Eq. (1) all  $n$ -grams are compared with each other, we only compare  $n$ -grams that are in close proximity to the position of the  $n$ -gram in the word to be compared when computing the similarity score.

Overall, the computation of the similarity score  $S$  for a given  $n$ -gram size  $n$  and a given odd-numbered window size  $m$  can be defined as follows. Assuming that  $u$  is the longer word (if  $v$  is longer than  $u$  then  $u$  and  $v$  can be simply exchanged):

$$S_{n,m}(u, v) = \frac{\sum_{i=2}^{|u|-n+1} \sum_{j=\frac{m-1}{2}}^{\frac{m-1}{2}} g(u_{i,i+(n-1)}, v_{i+j,i+j+(n-1)})}{N} \quad (2)$$

$$\text{where } g(a, b) = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{otherwise.} \end{cases} \quad \text{and}$$

$$u_{i,j} = \begin{cases} \text{substring}(u, i, j) & \text{if } i \leq j \\ \text{""} & \text{otherwise.} \end{cases}$$

Here,  $u$  and  $v$  are the words to be compared, the nested sum counts the number of  $n$ -grams in  $v$  that are similar to  $n$ -grams in a window the size of  $m$  around the same position in word  $v$ .  $N$  is computed similarly as in Eq. (1).

### B. Mutual Information (MI)

Given a query, the set of possible expanded terms using the revised  $n$ -gram will be generated; the coherence between the expanded terms is computed based on mutual information (MI). Giving a source of data, Mutual Information (MI) is a measure to calculate the correlation between terms in specific space (corpus or web). MI based approaches have been used often in word sense disambiguation task e.g., [28], [29]. Here in this paper mutual information approach is used to detect the noise term/terms based on its correlation with other terms in web.

Given a query term  $q_i = \{t_1, t_2, \dots, t_n\}$  and a set of its revised  $n$ -gram model generated expanded terms  $\{ext_{i,1}, ext_{i,2}, \dots, ext_{i,m_i}\}$ , where  $m_i$  defines the number of extended terms for  $t_i$  and  $1 \leq i \leq n$ . Given the set of  $\frac{n(n-1)}{S}$  combinations, where  $S$  is the size of each combinations set, then the set of combinations between all expanded terms is defined as  $Com_i = \{\{ext_{i,j}, ext_{i,k}\} | 1 \leq j < n, j < k \leq n\}$ . The mutual information of each combination set can be

computed based on the following equation:

$$MI(q_{t_1}, q_{t_2}) = \log_2 \frac{p(q_{t_1}, q_{t_2})}{p(q_{t_1})p(q_{t_2})} \quad (3)$$

where  $p(q_{t_i}, q_{t_j})$  being the joint probability of both expanded terms in the combination sets to occur in web. The probability is estimated by the relative frequency of the expanded terms in a given corpus, here the web, i.e., it is estimated by how many times  $q_{t_i}, q_{t_j}$  occur together in a (web) document.

### C. A Walk Through Example

To illustrate the improvement of the revised  $n$ -gram algorithm using the statistical co-occurrences data obtained from web, let us consider the following example.

The user submit the query صحيفة *shyfh* (Newspaper), the system using the revised  $n$ -gram model with similarity threshold of 60% expanded the user query with the following terms: (بصحيفة *bshyfh* "by Newspaper", وصحيفة *wshyfh* "and Newspaper", الصحيفة *alshyfh* "for the Newspaper", نحيفة *nhyfh* ("slim" Feminine) and الصحيفة *alshyfh* "for a Newspaper". The algorithm starts by generating all possible combinations between the expanded terms where  $Com_i = \{\{ext_{i,j}, ext_{i,k}\} | 1 \leq j < 5, j < k \leq 5\}$ . After generating all possible combinations between the expansion terms, the mutual information score for each expansion term combination will be calculated based on Eq. (3). Table III illustrates possible expanded term combinations and their mutual information score. As shown in Table III, one of the expanded term combinations included the expanded term نحيفة *nhyfh* "slim". It has the lowest mutual scores (23.793, 23.790, 23.165 and 21.314).

As shown in Table IV, the same expanded term has the lowest MI average score (23.015), which is below the MI score mean (25.456), and thus will be classified by the proposed approach as a noisy term and will be eliminated. In contrast, all other expanded terms have an average mutual score, which is above the MI score mean and thus should be correct expanded terms for the user's query.

## V. EVALUATION

In our evaluation, we compared our Revised  $n$ -gram and (revised  $n$ -gram + MI) approaches with the pure  $n$ -gram and edit distance approaches. We used  $n = 2$  (bigrams) to enable retrieval of short words, as well as other word lengths. In order, to gain a certain degree of accuracy, we obtained the

TABLE III  
EXPANDED TERM COMBINATIONS AND THEIR *MI* SCORES.

Expanded Terms Combinations	<i>MI</i> Score
(صحيفة و صحيفه , الصحيفة الصحيفه ) "and Newspaper, for the Newspaper"	28.651
(الصحيفة الصحيفه , الصحيفة الصحيفه ) "for the Newspaper, for a Newspaper"	28.075
(بصحيفة الصحيفه , الصحيفة الصحيفه ) "by Newspaper, for a Newspaper"	27.054
(صحيفة و صحيفه , الصحيفة الصحيفه ) "and Newspaper, for a Newspaper"	27.047
(بصحيفة و صحيفه , صحيفة و صحيفه ) "by Newspaper, and Newspaper"	26.486
(بصحيفة الصحيفه , الصحيفة الصحيفه ) "by Newspaper, for the Newspaper"	25.186
(الصحيفة نحيفة , صحيفة نحيفة ) "for the Newspaper, slim"	23.793
(بصحيفة نحيفة , صحيفة نحيفة ) "by Newspaper, slim"	23.790
(صحيفة و صحيفه , صحيفة نحيفة , صحيفة نحيفة ) "and Newspaper, slim"	23.165
(صحيفة نحيفة , الصحيفة الصحيفه ) "slim, for a Newspaper"	21.314
The <i>MI</i> score mean	25.456

TABLE IV  
EXPANDED TERMS AND THEIR AVERAGE *MI* SCORES.

Expanded Terms	<i>MI</i> average Score
(الصحيفة الصحيفه ) "for the Newspaper"	26.421
(صحيفة و صحيفه ) "and Newspaper"	26.337
(الصحيفة الصحيفه ) "for a Newspaper"	25.872
(بصحيفة الصحيفه ) "by Newspaper"	25.629
(صحيفة نحيفة ) "slim"	23.015

statistical co-occurrence data needed for the *MI* algorithm, from the web, using the yahoo search engine <sup>3</sup>.

#### A. Data Selection

In order to create the lists of expanded terms for each test query, we crawled the Web for articles published on one popular Arabic news Web site (CNN-Arabic)<sup>4</sup> in the period from January 2002 to March 2007. We obtained 5,792 Arabic documents, all of which are abstracts of articles on politics, sports, art, economy, and information science (size 60 MB). The approaches were evaluated against 500 queries that were formulated randomly, ensuring that the length of the query terms vary and short as well as long query terms are included. In order to construct the random queries, the algorithm requires the availability of a lexicon of terms that were extracted from the test data.

#### B. Comparison of Conflation Approaches and Web Experiment

To evaluate the proposed approaches, we used Precision and Recall. Precision and Recall are measures used to evaluate the performance of information retrieval (IR) approaches. Precision is the number of relevant documents retrieved divided by the total number of documents retrieved. Recall is the number of relevant documents retrieved divided by the total number of existing relevant documents in the data collection that should have been retrieved.

As table VI shows, in the first experiment, we calculated the average precision (based on the randomly selected 500 queries) for the pure trigram, edit distance, revised bigram and

(revised bigram + *MI*) for the similarity thresholds of 60, 65, 70, 75, 80, 85, 90, and 95% (Table VI shows the precision average). The trigram approaches (pure and revised) achieved higher precision than the revised bigram approach but in the same time it achieved lower recall than the revised bigram as it will be shown next in this section. The revised bigram precision was improved by 3.3% using mutual information approach based on statistical data obtained from web.

For the second experiment, we estimated the average recall and F-measure for a sample of 30 queries out of 500. The query terms were selected in the same way as described in Section V-A. For all queries, the number of relevant documents were obtained manually, by selecting all possible word variations (due to this huge manually work we could only provide this data for 30 queries). We obtained the precision, recall and F-Measure using five conflation approaches pure-trigram, pure-bigram, edit distance, revised-bigram and (revised-bigram + *MI*). As shown in Table V the revised bigram approach gained a higher F-measure of up to 84% compared to the pure trigram, pure bigram, and edit-distance approaches. These results show that the revised *n*-gram has gained an overall higher degree of retrieval performance than the pure *n*-gram and edit-distance approaches. As Table V shows, the revised bigram and pure trigram approaches have very similar precision, but the pure trigram approach missed many relevant documents and therefore has a lower average recall than the revised bigram approach.

The pure bigram approach has similar recall compared to the revised bigram approach. The pure bigram approach has lower precision since it does not take into account the order of the *n*-grams and therefore it is possible that many irrelevant

<sup>3</sup>yahoo.com, search performed on 19/02/2011

<sup>4</sup>http://arabic.cnn.com/

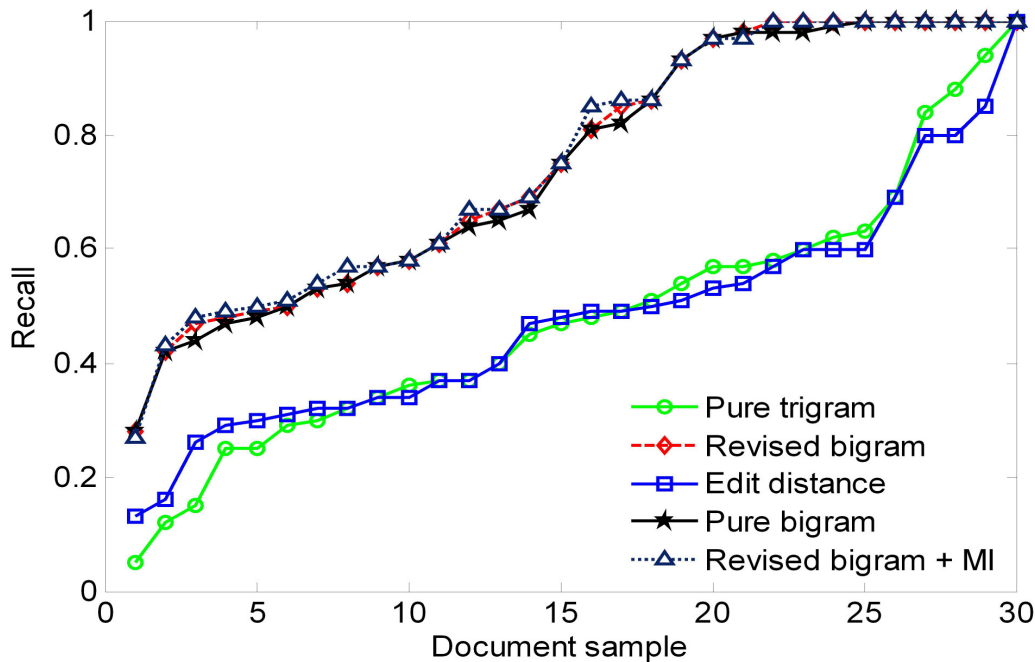


Fig. 1. Average recall for pure trigram, edit distance, pure bigram, revised bigram and (revised bigram +  $MI$ ) approaches (sorted by recall value).

documents will be retrieved. The approach using edit distance has lower precision. This is due to the fact that this method does not take into account the relationship between characters in the compared words as the  $n$ -gram approaches do. Figure 1 illustrates that the revised bigram approach gained a higher average recall than the pure trigram, edit distance and pure bigram approaches, since it took into account different word lengths and similarity enhancement.

For the third experiment, we performed the web experiments using the mutual information approach to improve the precision of revised bigram approach. This was done by eliminating the bigram generated noisy expanded terms as discussed in Section IV-B, Table V and Figure 1 shows that the mutual information approach using statistical co-occurrence data obtained from the web succeeded in eliminating 25 irrelevant expanded terms generated by the revised bigram approach. The failed cases were counted when the algorithm failed to eliminate the noisy terms or when the algorithm eliminate a corrected expanded term/terms along with the noisy one.

For example, we consider the query  $افريقيا$   $afryqyā$  "Africa", the algorithm succeeded in eliminating the noisy term  $فريقي$   $fryqy$  "my team" or "two teams" but at the same time, it eliminated a relevant term  $بافريقيا$   $bāfryqyā$  "by Africa". One interpretation for this lack, is that the word  $فريقي$   $fryqy$  "my team" or "two teams" with average  $MI$  score's (27.999) frequently appeared in the context of African sport and thus it increases the  $MI$  score mean (28.437) in that the average

$MI$  scores for the relevant word  $بافريقيا$   $bāfryqyā$  "by Africa" (27.708) is below the  $MI$  score mean.

TABLE VI  
AVERAGE PRECISION FOR ALL APPROACHES.

Techniques	Precision %
Revised bigram	91.3
Revised-bigram + $MI$	94.6
Pure bigram	79.4
Revised trigram	98.7
Pure trigram	95.7
Edit distance	87.3

## VI. CONCLUSION

We presented a language-independent conflation approach, i.e., an approach that does not depend on any predefined rules or previous knowledge of linguistic information about the target language. Furthermore, we evaluated our approach successfully on the Arabic language, which is one of most inflected languages in the world. In order to deal with  $n$ -gram noisy expanded terms, a mutual information approach applied to statistical co-occurrences data obtained from web was developed, in that the terms that have less cohesion score with other will be assumed as noisy terms and thus will be eliminated. The eliminations of the  $n$ -gram noisy generated terms improved the precision of the revised  $n$ -gram with 4%. The failed cases by the algorithm can be interrelated by the lack of the training data or by the very generic term usage where terms can appear in different contexts.

TABLE V  
AVERAGE RECALL, PRECISION, AND F-MEASURE FOR THE FIVE APPROACHES FOR A SAMPLE OF 30 QUERIES OUT OF 500.

	Pure-trigram	Pure-bigram	Edit distance	Revised-bigram	Revised-bigram + MI
Retrieved	366	629	400	596	571
Relevant	360	539	358	554	554
Irrelevant	6	90	42	42	17
Miss Relevant	6	195	376	180	180
Precision	0.98	0.86	0.89	0.93	0.97
Recall	0.49	0.73	0.49	0.76	0.76
F-Measure	0.65	0.80	0.64	0.84	0.86

## REFERENCES

- [1] S. Al-Fedaghi and F. Al-Anzi, "A new algorithm to generate arabic root-pattern forms," in *Proceedings of the 11th National Computer Conference, King Fahd University of Petroleum and Minerals*, Dhahran, Saudi Arabia, 1989, pp. 04–07.
- [2] H. Moukdad and A. Large, "Information retrieval from full-text arabic databases: Can search engines designed for english do the job?" *International Journal of Libraries and Information Services*, pp. 63–74, 2001.
- [3] S. Kosinov, "Evaluation of n-grams conflation approach in text-based information retrieval," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL*, 2001, pp. 657–664.
- [4] W. B. Frakes, "Stemming algorithms," *Information retrieval: data structures and algorithms*, pp. 131–160, 1992.
- [5] M. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.
- [6] J. Lovins, "Development of a stemming algorithm," *Mechanical Translation and Computational Linguistics*, vol. 11, pp. 22–31, 1968.
- [7] H. M. and W. S., "Word segmentation by letter successor varieties," *Information Processing and Management*, vol. 10, pp. 371–386, 1974.
- [8] M. Dang and S. Choudri, "Simple unsupervised morphology analysis algorithm," in *Unsupervised Segmentation of Words into Morphemes: Challenge 2005, Laboratory of Computer and Information Science*, 2005.
- [9] S. Bordag, "Unsupervised knowledge-free morpheme boundary detection," in the *International Conference on Recent Advances in Natural Language Processing (RANLP 05)*, 2005. [Online]. Available: <http://wortschatz.unileipzig.de/?sbordag/papers/BordagMorphy05.pdf>
- [10] J. Xu and W. B. Croft, "Corpus-based stemming using co-occurrence of word variants," *ACM Transactions on Information Systems*, vol. 16, no. 1, pp. 61–81, 1998.
- [11] M. Greengrass, A. M. Robertson, S. Robyn, and Willett, "Processing morphological variants in searches of latin text," *Information research news*, vol. 6, no. 4, pp. 2–5, 1996.
- [12] V. Berlian, S. N. Vega, and S. Bressan, "Indexing the indonesian web: Language identification and miscellaneous issues," in *Proceedings of Tenth International World Wide Web Conference*, Hong Kong, 2001.
- [13] J. Carlberger, H. Dalianis, M. Hassel, and O. Knutsson, "Improving precision in information retrieval for swedish us-ing stemming," in *Proceedings of NODALIDA '01 - 13th Nordic conference on computational linguistics*, Uppsala, Sweden, 2001.
- [14] W. Kraaij and R. Pohlmann, "Viewing stemming as recall enhancement," in *Proceedings of ACM SIGIR96*, 1996, pp. 40–48.
- [15] C. Monz and M. de Rijke, "Shallow morphological analysis in monolingual information retrieval for dutch, german and italian," in *Proc. of Evaluation of Cross-Language Information Retrieval Systems CLEF 2001*, ser. Lecture Notes in Computer Science, vol. 2406. Springer-Verlag, 2002, pp. 262–277.
- [16] L. Larkey, L. Ballesteros, and M. Connell, "Light stemming for arabic information retrieval," in *Arabic computational morphology*, A. Soudi, A. V. den Bosch, and G. Neumann, Eds. Netherlands: Springer-Verlag, 2007, vol. 38, pp. 221–243.
- [17] A. Gelbukh, M. Alexandrov, and S. Han, *Detecting Inflection Patterns in NL by Minimization of Morphological Model*, ser. LNCS 3287. Springer, 2004, pp. 432–438.
- [18] S. Khoja and R. Garside, "Stemming arabic," Website, 1999, available online at <http://zeus.cs.pacificu.edu/shereen/research.htm>; visited on January 15th 2009.
- [19] T. Buckwalter, "Arabic morphological analyzer version 1.0." Website, 2002, available online at <http://www ldc.upenn.edu/>; visited on January 8th 2010.
- [20] A. N. D. Roeck and W. Al-Fares, "A morphologically sensitive clustering algorithm for identifying arabic roots," in *Proceedings of ACL 2000*, Hong Kong, 2000, pp. 199–206.
- [21] J. Mayfield, P. McNamee, C. Costello, C. Piatko, and A. Banerjee, "Experiments in filtering and in arabic, video, and web retrieval," in *Proc. of the Eighth International Symposium on String Processing and Information Retrieval (SPIRE 2001)*, 2001, pp. 136–142.
- [22] K. Darwish and D. W. Oard, "Term selection for searching printed arabic," in *Proc. of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland, 2002, pp. 261–268.
- [23] S. H. Mustafa, "Character contiguity in n-gram-based word matching: the case for arabic text searching," *Processing and Management*, vol. 41, no. 4, pp. 819–827, 2004.
- [24] A. Ghaoui, F. Yvon, C. Mokbel, and G. Chollet, "On the use of morphological constraints in n-gram statistical language model," in *Proc. of Interspeech-2005*, 2005, pp. 1281–1284.
- [25] H. Abu-Salem, "Comparison of stemming and n-gram matching for term-conflation in arabic text," *International Journal of Computer Processing of Oriental languages*, vol. 17, no. 2, pp. 61–81, 2004.
- [26] F. Ahmed and A. Nürnberger, "Evaluation of n-gram conflation approaches for arabic text retrieval," *Journal of the American Society for Information Science and Technology (JASIST)*, vol. 60, no. 7, pp. 1448–1465, 2009.
- [27] B.-O. Khaltar, A. Fujii, and T. Ishikawa, "Extracting loanwords from mongolian corpora and producing a japanese-mongolian bilingual dictionary," in *Proc. of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL*, 2006, pp. 657–664.
- [28] F. Ahmed and A. Nürnberger, "multi searcher: can we support people to get information from text they can't read or understand?" in *SIGIR '10: Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 2010, pp. 837–838.
- [29] F. Ahmed, A. Nürnberger, and M. Nitsche, "Supporting arabic cross-lingual retrieval using contextual information," in *Multidisciplinary Information Retrieval*, A. Rauber and A. de Vries (Eds.), Eds. Berlin-Heidelberg: Springer-Verlag, 2011, vol. 6653, pp. 30–45.





# Improving the predictability of ICU illness severity scales

M. Alqarni,<sup>\*</sup> Y. Arabi,<sup>†</sup> T. Kakiashvili,<sup>‡</sup> M. Khedr,<sup>†</sup> W. W. Koczkodaj,<sup>\*</sup>  
 J. Leszek,<sup>§</sup> A. Przelaskowski,<sup>¶</sup> K. Rutkowski<sup>||</sup>

<sup>\*</sup> Laurentian University, Sudbury, Ontario, Canada

<sup>†</sup> King Saud Bin Abdulaziz University for Health Sciences, Saudi Arabia

<sup>‡</sup> Sudbury Therapy, Sudbury, Ontario, Canada

<sup>§</sup> Medical University, Wrocław, Poland

<sup>¶</sup> Warsaw University of Technology, Warsaw, Poland

<sup>||</sup> Jagellonian University, Cracow, Poland

**Abstract**—This study demonstrates how to improve the predictability of one of the commonly used ICUs severity of illness scales, namely APACHE II, by using the consistency-driven pairwise comparisons (CDPC) method. From a conceptual view, there is little doubt that all items have exactly equal importance or contribution to predicting mortality risk of patients admitted to ICUs. Computing new weights for all individual items is a considerable step forward since it is based on reasonable to assume that not all individual items have equal contribution in predicting mortality risk. The received predictability improvement is 1.6% (from 70.9% to 72.5%) and the standard error decreased from 0.046 to 0.045. This must be taken as an indication of the right way to go.

**Index Terms**—medical scales, illness severity, expert system, consistency-driven pairwise comparisons, inconsistency analysis.

## I. INTRODUCTION

**E**VER growing medical care costs motivate us to conduct more research toward the improvement of the severity of illness scales. The challenge is to leave the severity of illness scales unchanged as they are based on well established medical knowledge. Under this assumption, we are left with only one solution: weight for individual scale items must be computed instead of being arbitrarily set.

Severity of illness scales have wide application in medicine but psychiatry and intensive care settings are two top specializations where the use of severity of illness scales seems to be of great use, although for different reasons. In psychiatry, the use of tests, such as blood or X-rays, is limited and psychiatrists often rely on asking questions or observations. On the other hand, patients in the intensive care medicine must be rapidly evaluated based on many factors upon arrival and it can be at least in part done by nurses. Systems for predicting hospital mortality, such as the Acute Physiology and Chronic Health Evaluation (APACHE) II, are attractive options for this purpose because they rely on data collected within 24 hours after admission to the intensive care unit (ICU). It is mainly used to predict hospital mortality and reflect the severity of illness.

The corresponding author: wkoczkodaj@cs.laurentian.ca  
 Alphabetical order implies the equal contribution.

Acute Physiology and Chronic Health Evaluation, APACHE II, was introduced in [7] by Knaus in 1985 for predicting the hospital mortality in ICU patients. It has been designed based on data collected in 5,815 intensive care admissions from 13 hospitals. APACHE II measures severity of illness by a numeric score which can be converted into predicted mortality by using a logistic regression formula developed and validated on populations of ICU patients (for details, see [2], [3]).

## II. ICU SCALES

Medical scales (sometimes called medical measures) are scales used to describe or assess medical conditions. Amongst them, ICU scales are of considerable importance. An intensive care unit (ICU) also called intensive therapy unit, critical care unit (CCU), or intensive treatment unit (ITU) is a specialized department in a hospital for providing intensive-care medicine. Some hospitals also have designated intensive care areas for certain specialties of medicine, depending on the needs and resources of the hospital. For example, stroke is usually treated this way. Glasgow Coma Scale (GCS) was first introduced in 1974 by Teasdale G, Jennett B. in [17]. It aims to provide an understandable and clear way of observing change in the level of consciousness of patients having head injuries. In essence, the GCS was developed to standardize the reporting of neurologic findings and to provide an objective measure of the level of function of comatose patients [6]. Currently, GCS is one of the most used scales to assist the conditions of Trauma patients. It has only three items (elements):

- 1) Best eye response (E)
- 2) Best verbal response (V)
- 3) Best motor response (M)

There are four grades eye responses (E) starting with the most severe: 1 = "No eye opening" to 4 = Eyes opening spontaneously. For verbal response (V), the grads range from 1 = "Makes no sounds" to 5 = "Oriented, converses normally". Motor response (M) starts with 1 = "Makes no movements" and ends with 6 = "Obeys commands". Generally, brain injury is classified as:

- Severe, with  $GCS \leq 8$

TABLE I  
REVISED TRAUMA SCORE

Glasgow Coma Scale (GCS)	Systolic Blood Pressure (SBP)	Respiratory Rate (RR)	Coded Value
13-15	>89	10-29	4
9-12	76-89	>29	3
6-8	50-75	6-9	2
4-5	1-49	1-5	1
3	0	0	0

- Moderate, GCS 9 - 12
- Minor, GCS  $\geq$  13.

GCS is a part of several ICU scales, including APACHE II.

The Revised Trauma Score is a concatenation of: Glasgow Coma Scale, systolic blood pressure, and respiratory rate. Based on [4], TABLE II demonstrate the Revised Trauma Score. The RTS ranges from 0 to 12. A patient with an RTS = 12 is categorized as DELAYED (e.g., walking wounded), 11 is URGENT (intervention is required but the patient can wait a short time), and 10-3 is IMMEDIATE (immediate intervention is necessary). The last possible category is MORGUE, which is given to mortally injured people having RTS score from 0 to 3.

Needless to say that with the method presented in this study, the predictability can be improved for all these scales. However, the deep throat for our research is data gathering. It is not only costly and time consuming but it is easy to envision that the data collection may often interfere with the rescuing efforts so the emergency physician intuition may surpass it.

### III. DATA GATHERING AND ANALYSIS

Raw data, received from the Intensive Care Unit (ICU) of King Abdulaziz Medical City in Riyadh, Saudi Arabia in paper form, were entered into MS Excel to ease processing by other systems (e.g., SPSS). Excel provides a good tool for building forms for such a task using Visual Basic for Application (VBA) environment. Several forms were designed and then used to enter raw data. During the data entry process, 22 records from the received 165 records had to be removed. That was because three patients were re-admitted to the ICU, three patients had length of stay less than 24 hours, one patient had incomplete data, and all others had one or more missing value either it was not measured or was not available. According to [7], the first 3 group (re-admitted patients, patients with less than 24 hours stay, patients with incomplete data) are excluded from APACHE scoring. Moreover, in statistical analysis, all patients with one or more missing values were removed.

While considerable efforts have been taken to ensure high standards throughout all stages of collection and processing, the resulting data may not be sterile as they are clinical. Despite this effort, it should be clear that accidental errors are inevitable. Data was looked at from the medical point of view for mistakes and errors. We stress the use of the clinical data in our analysis as opposed to trial data.

Trial data tend to be more sterile than clinical data but as such, less valuable. They have the tendency of generating

TABLE II  
APACHE II SIMPLIFIED SPECIFICATION WITH GIVEN WEIGHTS

No	Item	Description	Range	weight
1	Temp	Temperature	0-4	1.67%
2	MAP	Mean Arterial Pressure	0-4	1.67%
3	HR	Heart Rate	0-4	1.67%
4	RR	Respiratory Rate	0-4	1.67%
5	OXY	Oxygenation	0-4	1.67%
6	$\rho$ H	$\rho$ H (Arterial)	0-4	1.67%
7	SS	Sodium (Serum)	0-4	2%
8	PS	Potassium (Serum)	0-4	2%
9	CS	Creatinine (Serum)	0-4	2%
10	He	Hematocrit	0-4	2%
11	WBC	White Blood Count	0-4	2%
12	GCS	15-GCS	0-15	20%
13	Age	Age	0-6	20%
14	CHP	Chronic Health Points	0, 2, 5	40%

better results since experiments are designed to prove a certain hypothesis.

### IV. PAIRWISE COMPARISONS LOGICAL MODEL

Based on existing data, we hypothesize that assuming identical weights for all scale items is not realistic for computing the predictability. Therefore, we grouped scale items according to their cohesiveness trying to have as loosely coupled groups as possible. It is done by the domain (medical) experts as following: First, a conceptual model must be designed by grouping criteria together. A rule of thumb proposed by Saaty in [14] is that no group should have more than seven criteria. To solve the problem of one group having a large number of criteria, split the group into subgroups with an acceptable number of criteria. Evidently, GCS, age and chronic conditions of patients have been kept in separation in the spirit of what was previously mentioned in the *Data collection and analysis* section. The remaining items have been grouped as shown in Fig. 1.

In a simplified model for APACHE II, we compared only the upper level observing that 11 items in the physiological group account for maximum 44 points while Chronic Health Point (CHP) is 2 or 5. It has created an immediate problem for relating groups at this level since our system allows us to define the ratio from 1 to 5 (and the inverse values). It is not a deficiency since the introduction of more groups is a solution. However, it requires more time so we decided to “cut the corners” and entered the compromised relative pairwise comparisons in Fig. 2.

The challenge posed to the pairwise comparisons method comes from the lack of consistency in assessments which arise in the real world [11]. With the development of the software, The Concluser, the consistency analysis has become relatively easy despite its complicated look. During the analysis process, the most inconsistent combinations of criteria are highlighted in the pairwise comparisons matrix in Fig. 3.

On Fig. 3, the maximum inconsistency is shown 0.67 and evidently bigger than the assumed heuristic  $\frac{1}{3}$  (as explained in the Appendix) hence the relative importance had to be

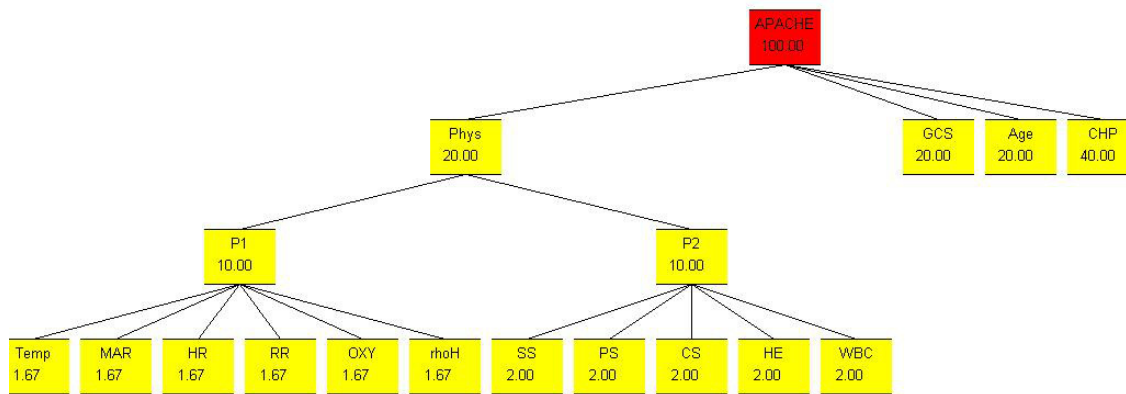


Fig. 1. The conceptual model of Apache II

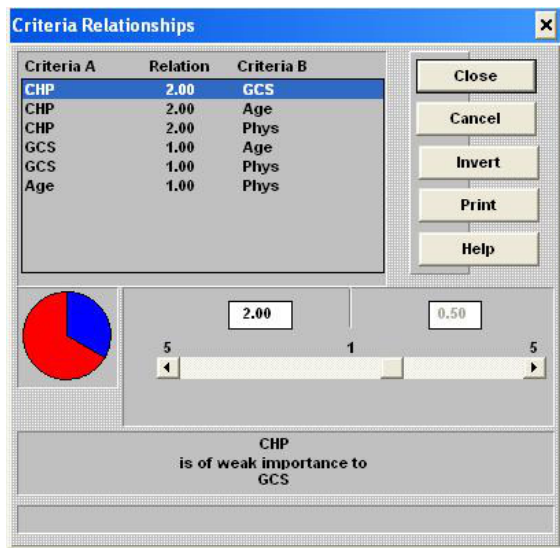


Fig. 2. The relative importance of the APACHE II scale items for level 1

reconsidered by experts based on their professional judgement and medical knowledge.

When an acceptable consistency level is reached (in our case, it has happened to be 0.00) as shown in fig 4, the weights [40%, 20%, 20%, 20%] are computed as normalized geometric means of rows and illustrated by Fig. 5. It needs to be stressed that for inconsistent pairwise matrix only approximated solution, in terms of weights, exists but it is sufficient since the reconstructed matrix from computed weights does not vary dramatically from the inconsistent pairwise comparisons matrix.

By the method of pairwise comparisons, the weight of 2.5 for CHP to Physio, GCS, and Age was optimal for the collected data. By dividing the original ratios of CHP to Physio, CHP to GCS, and CHP to Age over 2.5, we get the values in the upper row in Fig. 4. By the inconsistency analysis, we reduced values in the second row of matrix in Fig. 4 from 3 to 1.

The comorbidity component of APACHE II is represented

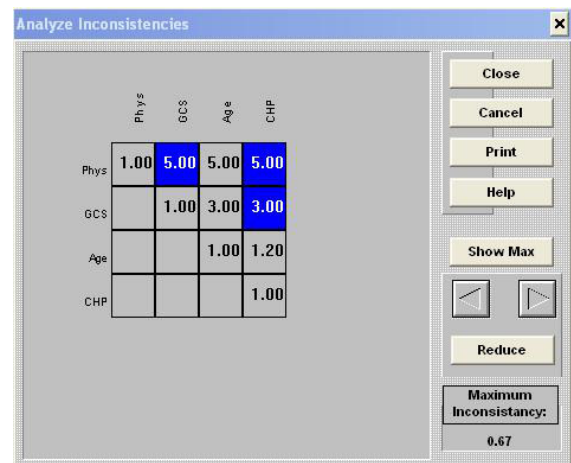


Fig. 3. The initial inconsistency analysis

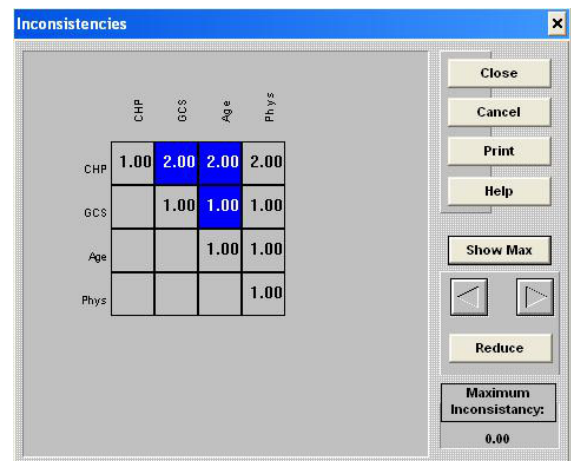


Fig. 4. The inconsistency analysis after the improvement

by Chronic Health Points (CHP). CHP are added for patients with a history of severe organ system deficiency or for patients who have immuno-compromised as follows:

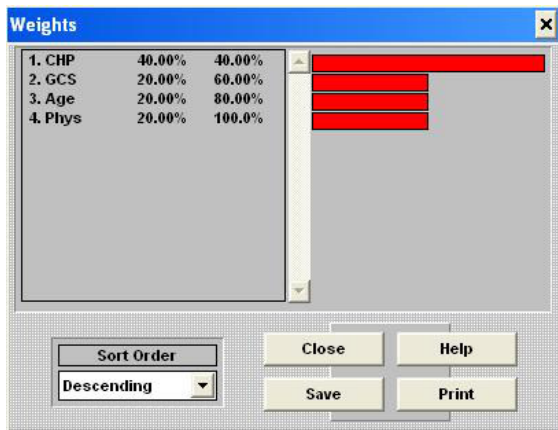


Fig. 5. The final weights computed for the APACHE II scale

- for nonoperative or emergency postoperative patients (5 points),
- for selective postoperative patients (2 points).

Immuno-compromised state must have been evident prior to the hospital admission and conform to the set medical criteria of liver, respiratory or renal system which are beyond the scope of this presentation. It was pointed out in [13] that: "The Chronic Health Points component of APACHE II had no significant discriminating ability (ROC area = 0.57, SE = 0.05)." However, we have provided evidence that the importance of CHP is greater than it has been originally assumed in [7] and should be changed from 2 and 5 to 5 and 12.5 respectively. To verify that these values are giving a better prediction of mortality, we applied ROC analysis by using SPSS. For CHP set to be 2.5, and 5, AUC (Area Under the Curve) was computed for the original data as 70.9% with the standard error of 0.046. While for the weight of 5 and 7.5 for CHP, AUC has increased to 72.5% with a better standard error of 0.045. The received predictability improvement is 1.6% and the standard error by 0.1%. Two ROCs are illustrated by Fig. 6.

In our humble opinion, the higher contribution of chronic conditions (CHP) can be explained by the simple observation that patients with chronic conditions are already receiving more medical attention than the rest of the population. When they are brought to ICU, usually it is for a serious enough reason that increases chance for their mortality.

## V. CONCLUSIONS

In this data analytic study, we tested the impact of applying the pairwise comparisons method to intensive care scales such as APACHE II. As far as we are aware, this is the first study to examine APACHE II's effectiveness while improving the predictability of a clinical assessment using a well-established method. Our results, although seemingly modest, have been consistent with improved psychometric properties of the questionnaire examined, as evidenced by the superior AUC classifier percentage after weights were added.

It is a progress report and a part of the MSc degree thesis of the first author. The presented hypothesis of changing the

weight for just one APACHE II scale item (Chronic Health Points) has been statistically proven and published in [9], [10], [1].

However, this is the first time of validation on the presented clinical data and every attempt will be made to use other clinical data in the future. We have proven the hypothesis that the proposed values in [7] in 1985 for Chronic Health Points should be changed from 2 and 5 to 5 and 7.5 respectively for elective post-operative patients and non-operative or emergency post-operative patients. This hypothesis is of considerable importance for health care planners. We hope that a more labor intensive analysis for the second level of the model would further improve the accuracy of the predictions of the presented scale. Similarly, adding the 50 principal diagnosis categories leading to ICU admission is a bit time consuming but will be incorporated in our approach for approximation of the mortality.

## ACKNOWLEDGMENT

We would like to acknowledge the endeavors of the sponsor, the Kingdom of Saudi Arabia, Ministry of Higher Education.

## REFERENCES

- [1] Adamic, P., Babiy, V., Janicki, R., Kakiashvili, T., Koczkodaj, W. W., Tadeusiewicz, R. *Pairwise comparisons and visual perceptions of equal area polygons*. *Perceptual and Motor Skills*, 108(1):37-42, 2009.
- [2] Arabi, Y., Abbasi, A., Goraj, R., Al-Abdulkareem, A., Al-Shimemeri, A., Kalayoglu, M., Wood, K. *External validation of a modified model of Acute Physiology and Chronic Health Evaluation (APACHE) II for orthotopic liver transplant patients*. *Critical Care Medicine*, 6(3), 2452-50, 2002.
- [3] Arabi, Y., Haddad, S., Goraj, R., Al-Shimemeri, A., Al-Malik, S. *Assessment of performance of four mortality prediction systems in a Saudi Arabian intensive care unit*. *Critical Care Medicine*, 6(2):166-74, 2006.
- [4] <http://www.trauma.org/archive/scores/rt.html> (accessed on 2011-06-24)

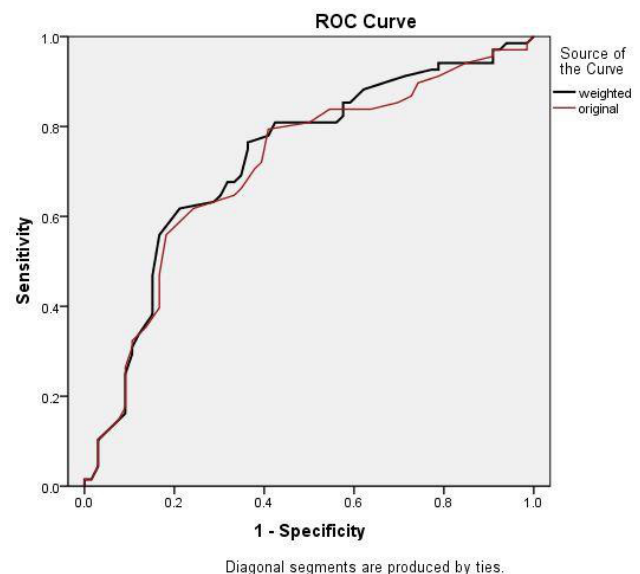


Fig. 6. Comparison between ROC results on both set of data

[5] Fawcett, T. *An introduction to ROC analysis*. Pattern Recognition Letters.27(8),861-874, 2006.

[6] Fischer J, Mathieson C. *The history of the Glasgow Coma Scale: Implications for practice* Critical Care Nurse,23(4):52-8, 2001

[7] Knaus, W., Draper, E., Wagner, D., Zimmerman, J. *APACHE II: a severity of disease classification system.*, Critical Care Medicine.13(10):818-29, 1985.

[8] Koczkodaj, W.W. *A new definition of consistency of pairwise comparisons*, Mathematical and Computer Modelling, (18)7,79-84,1993.

[9] Koczkodaj, W.W. *Statistically Accurate Evidence of Improved Error Rate by Pairwise Comparisons*, Perceptual and Motor Skills, 82,43-48, 1996.

[10] Koczkodaj, W.W., *Testing the Accuracy Enhancement of Pairwise Comparisons by a Monte Carlo Experiment*, Journal of Statistical Planning and Inference, 69(1), 21-32, 1998.

[11] Koczkodaj, W.W, LeBrasseur, R., Wassilew, A. ,Tadeuszewicz, R. *About Business Decision Making by a Consistency-Driven Pairwise Comparisons Method.*, Journal of Applied Computer Science.2009

[12] Koczkodaj, W.W., Szarek,S.J. *On distance-based inconsistency reduction algorithms for pairwise comparisons.*, Logic Journal of the IGPL 18(6):859-869, 2010.

[13] Poses, R., McClish, D., Smith, W., Bekes, C. , Scott, W. *Prediction of survival of critically ill patients by admission comorbidity.*, J Clin Epidemiol, 49(7):743-7, 1996.

[14] Saaty, L.T. *A Scaling Method for Priorities in Hierarchical Structures*, Journal of Mathematical Psychology 15(3),234-281,1977.

[15] Statistics Canada official web page, <http://www40.statcan.gc.ca/01/cst01/demo02a-eng.htm?sdi=population>, data retrieved on 2011-05-27

[16] Statistics Canada official web page, <http://www40.statcan.ca/01/cst01/health30a-eng.htm>, data retrieved on 2011-05-27

[17] Teasdale G, Jennett B. *Assessment of coma and impaired consciousness. A practical scale*. Lancet,2(7872):81-4, 1974.

[18] Zhai, Y., Janicki, R. *On Consistency in Pairwise Comparisons Based Numerical and Non-Numerical Ranking.*, Proceedings of the International Conference on Foundations of Computer Science, FCS 2010: 183-186, 2010.

APPENDIX

Using pairwise comparisons is a powerful method for synthesizing measurements and subjective assessments. From the mathematical point of view, the pairwise comparisons method generates a matrix (say  $A$ ) of ratio values ( $a_{ij}$ ) of the  $i$ th entity compared with the  $j$ th entity according to a given criterion. Entities/criteria can be both quantitative or qualitative allowing this method to deal with complex decisions. Comparing two entities in pairs to assess which of them is preferred, or has a greater amount of some property is irreducible since having one entity compared with itself has very little or practical meaning. However, subjective assessments often involve inconsistency, which is usually undesirable. The assessment can be refined via analysis of inconsistency, leading to reduction of the latter.

Making one comparison at a time is simpler than simultaneously assessing *all* items of a scale according to their contribution to the overall score. However, we need a method of synthesizing these partial assessments. The pairwise comparisons method, used since 1785, serves exactly this purpose, with the inconsistency analysis allowing us to localize the most questionable partial assessments and revise them if necessary.

From the mathematical point of view, the pairwise comparisons method creates a matrix (say  $A$ ) of values ( $a_{ij}$ ) of the  $i$ th entity compared with the  $j$ th entity:

$$A = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ \frac{1}{a_{12}} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \cdots & 1 \end{bmatrix}$$

A scale  $[\frac{1}{c}, c]$  is used for ‘ $i$  to  $j$ ’ comparisons where  $c > 1$  is a not-too-large real number (5 to 9 is used in most practical applications). It is usually assumed that all the values  $a_{ii}$  on the main diagonal are 1 (the case of ‘ $i$  compared with  $i$ ’, that is with itself) and that matrix  $A$  is *reciprocal* :  $a_{ij} = \frac{1}{a_{ji}}$  since ‘ $i$  to  $j$ ’ is (or at least, is expected to be) the reciprocal of ‘ $j$  to  $i$ ’. (In other words, for  $x, y \neq 0$ ,  $\frac{x}{y} = \frac{1}{\frac{y}{x}}$ .) However, in practice even the reciprocity condition is not always guaranteed. For example, in blind wine testing we may conclude that *wine  $i$*  is better than *wine  $j$*  if it is served in unmarked glasses.

Since 1996, a *distance-based* adjective has been used by other researchers for the new inconsistency defined in 1993 in [8]. The distance-based adjective reflects the nature of the *inconsistency indicator*, which is defined, in essence, as a function of a distance from the nearest consistent *triad* in matrix  $A$ . Unlike the eigenvalue-based inconsistency, introduced in [14]), which is of a *global* indicator, and as such a non-identifying, the distance-based inconsistency identifies the most inconsistent triad (or triads). It is the maximum over all triads  $\{a_{ik}, a_{kj}, a_{ij}\}$  of elements of  $A$  (say, with all indices  $i, j, k$  distinct) of their inconsistency indicators, which in turn are defined as  $ii := \min(|1 - \frac{a_{ij}}{a_{ik}a_{kj}}|, |1 - \frac{a_{ik}a_{kj}}{a_{ij}}|)$ .

The inconsistency indicator of  $A$  equals zero if and only if  $A$  is fully consistent as it was (in all likelihood shown for the first time in [14]. Consistent matrices correspond to the ideal situation in which we know all exact values of all properties (or at least it seems to be a reasonable assumption to make). However, a realistic situation which is complex enough, nearly always involves inconsistency and we need to deal with it. In fact, when we are able to locate it, our comparisons can be reconsidered to reduce the inconsistency in the next round.

Certainly, inconsistency is undesirable in a system. On the other hand, although this may sound strange, it is not easy (we suspect, impossible) to construct a non-trivial *fully* inconsistent system: an “ideal” system where everything contradicts everything else. This question (or a family of questions, which we suggest only vaguely here) seems quite important as such impossibility would imply that *every* scenario of answers to pairwise comparison queries (even deliberately false) would necessarily create “apparent” consistencies.

In practical applications, a high value of the inconsistency indicator is a “red flag,” or a sign of potential problems. A distance-based inconsistency reduction algorithm focuses, at each step, on an inconsistent triad and “corrects” it by replacing it with a consistent (or, more generally, less inconsistent) triad. It resembles “whac-a-mole,” a popular arcade game. One difference is that instead of one mole, we have three array elements as explained above. After “hitting the mole” (which generally results in some other “moles” coming out), the next triad is selected according to some rule (which may be for example the greedy algorithm), and

the process is repeated. Numerous practical implementations (e.g., a hazard rating system for abandoned mines in Northern Ontario) have shown that the inconsistency converges relatively fast. However, the need for rigorously *proving* the convergence (that is, showing that whacked moles *always* have the tendency of coming out less and less eagerly) was evident.

The distance-based inconsistency locates the most inconsistent triad or triads. This allows the user to reconsider the assessments included in the most inconsistent triad.

$$\begin{array}{c|cccc} & \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{D} \\ \hline \mathbf{A} & 1 & \boxed{1} & \boxed{5} & 4 \\ \mathbf{B} & 1 & 1 & \boxed{2} & 2\frac{1}{2} \\ \mathbf{C} & \frac{1}{5} & \frac{1}{2} & 1 & \frac{1}{2} \\ \mathbf{D} & \frac{1}{4} & \frac{2}{5} & 2 & 1 \end{array} \quad (1)$$

Changing the value 1 in the above triad to 2.5 makes this triad fully consistent since  $2.5 \cdot 2 = 5$ . Unfortunately, this is not the end of our problems since there is another triad  $[2, 2\frac{1}{2}, \frac{1}{2}]$  that is inconsistent and “boxed” below:

$$\begin{array}{c|cccc} & \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{D} \\ \hline \mathbf{A} & 1 & 2\frac{1}{2} & 5 & 4 \\ \mathbf{B} & \frac{2}{5} & 1 & \boxed{2} & \boxed{2\frac{1}{2}} \\ \mathbf{C} & \frac{1}{5} & \frac{1}{2} & 1 & \boxed{\frac{1}{2}} \\ \mathbf{D} & \frac{1}{4} & \frac{2}{5} & 2 & 1 \end{array} \quad (2)$$

Assume that we have good reason (coming from the knowledge domain; not from mathematics), to change the value of  $2\frac{1}{2}$  to 1. It is an arbitrary decision since 2 could have been changed to 5 or  $\frac{1}{2}$  to  $1\frac{1}{4}$ , also making this triad consistent. Only the domain knowledge can determine the change of the value (or values) in a triad. However, changing 2 may not be wise since it belongs to a consistent triad altered in the previous step. In our case, the only reason why we have chosen to change  $2\frac{1}{2}$  to 1 was to illustrate how the inconsistency procedure works and the reader may be disappointed to find that there is yet another triad “boxed” below which is inconsistent:

$$\begin{array}{c|cccc} & \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{D} \\ \hline \mathbf{A} & 1 & \boxed{2\frac{1}{2}} & 5 & \boxed{4} \\ \mathbf{B} & \frac{2}{5} & 1 & 2 & \boxed{1} \\ \mathbf{C} & \frac{1}{5} & \frac{1}{2} & 1 & \frac{1}{2} \\ \mathbf{D} & \frac{1}{4} & 1 & 2 & 1 \end{array} \quad (3)$$

Finally, we change 4 to  $2\frac{1}{2}$  making the entire table fully consistent.

$$\begin{array}{c|cccc} & \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{D} \\ \hline \mathbf{A} & 1 & 2\frac{1}{2} & 5 & 2\frac{1}{2} \\ \mathbf{B} & \frac{2}{5} & 1 & 2 & 1 \\ \mathbf{C} & \frac{1}{5} & \frac{1}{2} & 1 & \frac{1}{2} \\ \mathbf{D} & \frac{1}{4} & 1 & 2 & 1 \end{array} \quad (4)$$

In practice, inconsistent assessments are unavoidable when at least three factors are independently compared against each other. The corrections for real data are done on the basis of

professional experience, the case-based knowledge, and by the careful examination of all criteria involved (not necessarily in the current triad).

An acceptable threshold of inconsistency, for most practical applications, turns out to be  $\frac{1}{3}$ . This is so because one value in a triad is not more than two grades off the scale from the remaining two values. This heuristic was introduced in [8] and it seems more mathematically sound than 10% proposed in [14].

There is no need to continue decreasing the inconsistency indefinitely to zero, as only a high value of it is harmful. In fact, a zero or a small inconsistency value may indicate that artificial data were entered hastily without reconsideration of former assessments, which is an unacceptable practice.

For the improved matrix, the normalized vector of weights is:

$$w = [0.5, 0.2, 0.1, 0.2]$$

It is identical for both the geometric means method, and the eigenvector method, since the eigenvector of a consistent pairwise comparisons matrix is always equal to the geometric means. For the original input matrix, which is inconsistent, the solutions are,

for the eigenvector method:

$$w = [0.441, 0.317, 0.101, 0.140]$$

and for geometric means method (computed as  $\sqrt[4]{\prod_{j=1}^N a_{ij}}$ ):

$$w = [0.445, 0.315, 0.100, 0.141]$$

The difference between both solutions is negligible. However, both solutions for the inconsistent matrix vary drastically from the solution for the consistent matrix.

It is important to note the difference between inaccuracy and inconsistency. For example, in a triad  $[2, 5, 3]$ , a rash approach may lead us to believe that  $A/C$  should indeed be 6 since it is  $2 \cdot 3$ , but we do not have any reason to reject the estimation of  $B/C$  as 2.5 or  $A/B$  as  $5/3$ . This is what inconsistency is about. It is not inaccuracy, but when used wisely, it may help to decrease inaccuracy.

The reader will notice that while the three-step inconsistency-reduction procedure performed above does not offend the common sense, it is rather *ad hoc*, hence not fully satisfactory. This remark applies both to the choices of triads to be corrected, and to the choices of the particular members of each such triad that is being modified. The algorithm analyzed in [12] (and, by extension, the present note) is more canonical with respect to the second point. In general, it replaces the triad  $\{a_{ik}, a_{kj}, a_{ij}\}$  by  $\{a_{ik}/r, a_{kj}/r, ra_{ij}\}$ , where  $r := \sqrt[3]{a_{ik}a_{kj}/a_{ij}}$ . This corresponds to subtracting from the matrix  $(\log a_{uv})$  its orthogonal projection onto the direction of the skew-symmetric matrix  $B = (b_{uv})$  defined by the requirement that  $a_{ik} = 1 = a_{kj}$ ,  $a_{ij} = -1$  and that all other super-diagonal entries are 0; the corresponding subspace in the context of Theorem is  $U = \{X : X \text{ is an } n \times n \text{ skew-symmetric matrix such that } \text{tr}BX = 0\}$ . In particular, for the first triad  $[1, 2, 5]$



considered above, we have  $r = 2/5$  and the corrected triad is  $[\sqrt[3]{5/2}, 2\sqrt[3]{5/2}, 5\sqrt[3]{2/5}] \approx [1.36, 2.71, 3.68]$ .

Monte Carlo studies have shown that approximations of highly inconsistent pairwise comparisons matrices yields high errors. Finding consistent approximations of such matrices makes little practical sense. From mathematical logic, we know that *only* falsehood can generate both truth or falsehood. However, the old adage that *one bad apple spoils the barrel* seems to be more applicable here: even a little bit of falsehood may contribute to massive errors and misjudgments. An approximation of a pairwise comparisons matrix is meaningful only if the initial inconsistency is acceptable (that is, located, brought under control and/or reduced to a certain predefined minimum; in our analogy, *always remove overripe fruit promptly if it is possible to find it*).

The new results and applications of pairwise comparisons show the importance of the consistency-driven approach. The

inconsistency concept still remains enigmatic and more research needs to be done. In particular, inconsistency in a general system needs to be defined and this study is a step forward. The idea of improving inaccuracy by controlling inconsistency cannot be wrong and a new approach to it is presented in [18]. Knowing what we do not know is essential to managing the knowledge and improving it. On the other hand, it is hard to change our knowledge if we choose not to know what we know or even should know.

The method of pairwise comparisons was used by a research team, lead by W.W. Koczkodaj, to develop AMIS (Abandoned Mines Hazard Rating System) for the government of Ontario (The Ministry of Northern Ontario and Mines). The system ranked an abandoned mine, located in Northern Ontario, as one of the most dangerous from a public safety point of view. Its eventual collapse convinced the government that its research founding was well spent.





# A Rough K-means Fragile Watermarking Approach for Image Authentication

Lamiaa M. El Bakrawy<sup>1,\*</sup>, Neveen I. Ghali<sup>1,\*</sup> and Aboul ella Hassanien<sup>2,\*</sup>, Tai-hoon Kim<sup>3</sup>

<sup>1</sup>Al-Azhar University, Faculty of Science, Cairo, Egypt

Email: lamiaabak@yahoo.com, nev\_ghali@yahoo.com

<sup>2</sup>Cairo University, Faculty of Computers and Information, Cairo, Egypt

Email: aboitcairo@gmail.com

\*Abo Research Laboratory (ARL), Cairo, Egypt

Email: aboitcairo@gmail.com

<sup>3</sup>Hannam University, Korea. Email: taihoonn@hannam.ac.kr

**Abstract**—In the past few years, various fragile watermarking systems have been proposed for image authentication and tamper detection. In this paper, we propose a rough k-means (RKM) fragile watermarking approach with a block-wise dependency mechanism which can detect any alterations made to the protected image. Initially, the input image is divided into blocks with equal size in order to improve image tamper localization precision. By combining image local properties with human visual system, authentication data are acquired. By computing the class membership degree of each image block property, data are generated by applying rough k-means clustering to create the relationship between all image blocks and cluster all of them. The embed watermark is carried by least significant bits (LSBs) of each pixel within each block. The effectiveness of the proposed approach is demonstrated through a series simulations and experiments. Experimental results show that the proposed approach can embed watermark without causing noticeable visual artifacts, and does not only achieve superior tamper detection in images accurately, it also recovers tampered regions effectively. In addition, the results show that the proposed approach can effectively thwart different attacks, such as the cut-and paste attack and collage attack, while sustaining superior tamper detection and localization accuracy. Furthermore, the results show that the proposed approach can embed watermark without causing noticeable visual artifacts.

## I. INTRODUCTION

**D**UE TO significant improvements in computer and Internet technology, a large amount of digital data is easily accessible to every one these days. This digital data, such as images, are commonly transmitted via the Internet. On the other hand, a variety of powerful image processing tools have also made digital image manipulations much easier. The ease and extent of such manipulations emphasize the need for image authentication techniques in applications where verification of integrity and authenticity of the image content is essential. Therefore, various authentication approaches have recently been proposed for verifying the integrity and authenticity of the image content [1], [2], [4].

The authentication approaches can be branched into two categories: digital signature-based approaches and digital watermark-based approaches. A digital signature can be either an encrypted or a signed hash value of image contents or

image features. The drawback of signature based approaches is that they cannot locate the regions where the image has been modified although they can detect if an image has been modified or not, also the transmission of a digital signature for each image requires additional bandwidth or storage that may not always be available [3], [11], [12]. Digital watermark-based approaches embed the authentication data directly into the original multimedia which provides an obvious solution to the preciously mentioned problems. Moreover, the digital watermark is capable of isolating manipulated image areas which is known as the tamper localization [5], [13]. So many researchers have proposed watermarking based approaches for image authentication [4], [5], [14].

The authentication digital watermark-based approaches can be classified as either fragile watermarking or semi-fragile watermarking. A fragile watermarking can detect any possible modification of the pixel values. On the other hand, semi-fragile watermarking can distinguish content-preserving operations from malicious manipulations, e.g., addition or removal of a significant element of the image [3], [15]. Various approaches for fragile watermarking [16][18] and semi-fragile watermarking [19], [20] have been proposed. since tamper detection and localization are well defined in fragile watermarking approaches, the work presented in this paper concentrates on the latter.

The remainder of this paper is ordered as follows. Brief introduction of rough sets and rough k-means clustering are introduced in Section (II). The details of the proposed approach is presented in Section (III). Section (IV) shows the experimental results. Conclusions are discussed in Section (V).

## II. PRELIMINARIES

### A. Rough sets

Due to space limitations we provide only a brief explanation of the basic framework of rough set theory, along with some of the key definitions. A more comprehensive review can be found in sources such as [10].

Rough sets theory proposed by Pawlak [9] is a new intelligent mathematical tool. It is based on the concept of

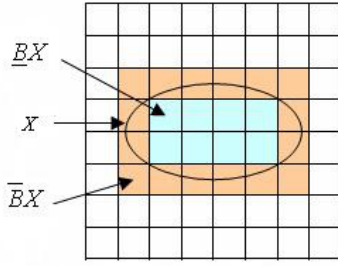


Fig. 1. Rough: boundary region

approximation spaces and models of the sets and concepts [7,8,9,11]. In rough sets theory, feature values of sample objects are collected in what are known as information tables. Rows of a such a table correspond to objects and columns correspond to object features.

Let  $\mathcal{O}, \mathcal{F}$  denote a set of sample objects and a set of functions representing object features, respectively. Assume that  $B \subseteq \mathcal{F}, x \in \mathcal{O}$ . Further, let  $[x]_B$  denote:  $[x]_B = \{y : x \sim_B y\}$ .

Rough sets theory defines three regions based on the equivalent classes induced by the feature values: lower approximation  $\underline{BX}$ , upper approximation  $\overline{BX}$  and boundary  $BND_B(X)$ . A lower approximation of a set  $X$  contains all equivalence classes  $[x]_B$  that are proper subsets of  $X$ , and upper approximation  $\overline{BX}$  contains all equivalence classes  $[x]_B$  that have objects in common with  $X$ , while the boundary  $BND_B(X)$  is the set  $\overline{BX} \setminus \underline{BX}$ , i.e., the set of all objects in  $\overline{BX}$  that are not contained in  $\underline{BX}$ . The approximation definition is clearly depicted in Fig. 1.

### B. Adaptation of K-means to rough set theory

K-means clustering is used for other segmentation than medical images [6], [10]. The name K-means originates from the means of the  $k$  clusters that are created from  $n$  objects. Let us assume that the objects are represented by  $m$ -dimensional vectors. The objective is to assign these  $n$  objects to  $k$  clusters. Each of the clusters is also represented by an  $m$ -dimensional vector, which is the centroid or mean vector for that cluster. The process begins by randomly choosing  $k$  objects as the centroids of the  $k$  clusters. The objects are assigned to one of the  $k$  clusters based on the minimum value of the distance  $d(v, x)$  between the object vector  $v = (v_1, \dots, v_j, \dots, v_m)$  and the cluster vector  $x = (x_1, \dots, x_j, \dots, x_m)$ . After the assignment of all the objects to various clusters, the new centroid vectors of the clusters are calculated as:

$$x_j = \frac{\sum_{v \in x} v_j}{|x|} \quad (1)$$

Where  $1 \leq j \leq m$ ,  $|x|$  is the size of cluster  $x$ . Incorporating rough sets into K-means clustering requires the addition of the concept of lower and upper bounds [7,8]. Calculation of the centroids of clusters from conventional K-means needs to

be modified to include the effects of lower as well as upper bounds. The modified centroid calculations for rough sets are given in Algorithm(1).

**Algorithm 1** The modified centroid calculations for rough sets

---

```

if  $\underline{BX} \neq \emptyset$  and  $\overline{BX} - \underline{BX} = \emptyset$  then
  Compute  $x_j = \frac{\sum_{v \in \underline{BX}} v_j}{|\underline{BX}|}$ 
end if
if  $\underline{BX} = \emptyset$  and  $\overline{BX} - \underline{BX} \neq \emptyset$  then
  Compute  $x_j = \frac{\sum_{v \in (\overline{BX} - \underline{BX})} v_j}{|\overline{BX} - \underline{BX}|}$ 
else
  Compute  $x_j = w_{lower} \times \frac{\sum_{v \in \underline{BX}} v_j}{|\underline{BX}|} + w_{upper} \times \frac{\sum_{v \in (\overline{BX} - \underline{BX})} v_j}{|\overline{BX} - \underline{BX}|}$ 
end if

```

---

Where  $1 \leq j \leq m$ . The parameters  $w_{lower}$  and  $w_{upper}$  correspond to the relative importance of lower and upper bounds, and  $w_{lower} + w_{upper} = 1$ . If the upper bound of each cluster were equal to its lower bound, the clusters would be conventional clusters. Therefore, the boundary region  $(\overline{BX} - \underline{BX})$  will be empty, and the second term in the equation will be ignored. Thus, the equation on Algorithm (1) will reduce to conventional centroid calculations.

The next step in the modification of the K-means algorithm for rough sets is to design criteria to determine whether an object belongs to the upper or lower bound of a cluster given as follows:

For each object vector,  $v$ , let  $d(v, x_j)$  be the distance between itself and the centroid of cluster  $x_j$ . Let  $d(v, x_i) = \min_{1 \leq j \leq K} d(v, x_j)$ . The ratio  $d(v, x_i)/d(v, x_j)$ ,  $1 \leq i, j \leq k$ , are used to determine the membership of  $v$ . Let  $T = \{j : d(v, x_i)/d(v, x_j) \leq \text{threshold } \epsilon \text{ and } i \neq j\}$ .

- 1) If  $T \neq \emptyset, v \in \overline{BX}_i$  and  $v \in \overline{BX}_j, \forall j \in T$ . Furthermore,  $v$  is not part of any lower bound. The above criterion guarantees that property (3) is satisfied.
- 2) Otherwise, if  $T = \emptyset, v \in \underline{BX}_i$ . In addition, by property (2),  $v \in \overline{BX}_i$ . It should be emphasized that the approximation space  $A$  is not defined based on any predefined relation on the set of objects. The upper and lower bounds are constructed based on the criteria described above.

## III. THE PROPOSED FRAGILE WATERMARKING APPROACH FOR IMAGE AUTHENTICATION

In this section, we explain the proposed watermarking approach. The system contains two procedures: watermark embedding procedure and tamper detection procedure. Details of the proposed system are described as follows.

### A. Watermark embedding

Let us consider,  $I$  is the host image of size  $M \times M$ , where  $M$  is assumed to be an even number. The original image is divided into non-overlapping  $2 \times 2$  blocks  $B_j (1 \leq j \leq \frac{M}{2} \times \frac{M}{2})$  which are arranged by the order from left to right and then top to bottom.

To generate the watermark, the two LSBs of all the pixels within each block of the host image are first set to zero. Each block  $B_j$  can be regarded as a 4-dimensional vector,  $B_j = (B_{j1}, B_{j2}, B_{j3}, B_{j4})$ . Then the RKM clustering is applied to classify all the blocks into  $k$  clusters. After performing the RKM clustering, for each block,  $B_j$ , let  $d(B_j, x_c)$  be the distance between itself and the centroid of cluster  $x_c$ ,  $1 \leq c \leq k$ . Let  $d(B_j, x_i) = \min_{1 \leq c \leq k} d(B_j, x_c)$ ,  $d(B_j, x_m) = \max_{1 \leq c \leq k} d(B_j, x_c)$ ,  $1 \leq i, m \leq k$ . Then we compute feature sequence  $F = \{f_1, f_2, \dots, f_{\frac{M}{2} \times \frac{M}{2}}\}$  by

$$f_j = \frac{d(B_j, x_i)}{d(B_j, x_m) - d(B_j, x_i)} \quad (2)$$

When  $f_j < 0.1$  then  $f_j = 0$  otherwise  $f_j = 1$

Assume that  $R = \{r_1, r_2, \dots, r_{\frac{M}{2} \times \frac{M}{2}}\}$  is a random sequence created by using a pseudorandom number generator (PRNG) seeded with a secret key SK, where  $r_j \in [0, 255]$ . For each block  $B_j$ , its corresponding authentication data  $a_j$  is constructed by the following Equation:

$$a_j = f_j \oplus r_j \quad (3)$$

where the symbol  $\oplus$  denotes the XOR operation. Each resultant 8-bit authentication data is embedded into the 8 LSBs of the corresponding image block, and the watermarked image  $I'$  is thus obtained. Finally, the set of cluster centers  $x$  acquired after performing the RKM clustering on image  $I$ , and the secret key SK should be kept securely by the image owner for further tamper detection.

### B. Tamper detection

The possibly distorted image  $I''$ , as in the authentication data embedding procedure, is first divided into non-overlapping  $2 \times 2$  blocks  $B_j''$  ( $1 \leq j \leq \frac{M}{2} \times \frac{M}{2}$ ). By verifying the watermark embedded in each image block, we can determine whether an image block has been tampered with.

To perform tamper detection, the embedded watermark sequence,  $A = \{a_1, a_2, \dots, a_{\frac{M}{2} \times \frac{M}{2}}\}$  is extracted from all the blocks of image  $I''$ , and then the two LSBs of all the pixels within each block are set to zero. Employing the set of cluster centers  $x$  kept by the image owner, to all blocks a feature sequence  $F'' = \{f_1'', f_2'', \dots, f_{\frac{M}{2} \times \frac{M}{2}}''\}$  can be derived by

$$f_j'' = \frac{d(B_j'', x_i)}{d(B_j'', x_m) - d(B_j'', x_i)} \quad (4)$$

When  $f_j'' < .1$  then  $f_j'' = 0$  otherwise  $f_j'' = 1$ . Let  $R = \{r_1, r_2, \dots, r_{\frac{M}{2} \times \frac{M}{2}}\}$  be a random sequence created by using the PRNG seeded with the secret key SK kept by the image owner, where  $r_j \in [0, 255]$ . The authentication data sequence  $A'' = \{a_1'', a_2'', \dots, a_{\frac{M}{2} \times \frac{M}{2}}''\}$  corresponds to image  $I''$  can be computed by applying  $f_j''$  and  $R$  to Eq. (3). Finally, the legitimacy of each block  $B_j''$  can be recognized by comparing  $a_j''$  with  $a_j$ . If they are the same,  $B_j''$  is a legitimate block; otherwise,  $B_j''$  is regarded as a tampered block.

## IV. EXPERIMENTAL RESULTS

Various experiments are carried out in this section to demonstrate the validity of the proposed fragile watermarking approach. We set the number of clusters  $k = 3$ ,  $w_{lower} = 0.75$ ,  $w_{upper} = 0.25$  and threshold  $\epsilon = 0.05$ . For quantitative evaluation, peak signal-to-noise ratio (PSNR) was used to measure the image quality of the watermarked image  $I'$  in comparison with the original image  $I$  which is given by the following form:

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE} (dB) \quad (5)$$

$$MSE = \frac{1}{M \times M} \sum_{i=1}^M \sum_{j=1}^M (I_{i,j} - I'_{i,j})^2, \quad (6)$$

where  $I_{i,j}$  denotes the pixel value of the original image,  $I'_{i,j}$  denotes the pixel value of the watermarked image,  $M \times M$  is the image size. Also, the true positive (TP) and false positive (FP) rates were used to measure the accuracy of tamper detection and localization, where the TP rate is the proportion of actual tampered pixels that were correctly reported as tampered pixels and the FP rate is the proportion of actual non-tampered pixels that were erroneously reported as tampered pixels.

### A. Performance evaluation

1) *Performance under cut-and-paste attack*: In this experiment, to simulate the cut-and-paste attack, the content of a watermarked image was modified by cutting regions from the same or another watermarked image and pasting them together to form a new image.

We used two 8-bit grayscale images, Pool image of size  $256 \times 256$  and field image of size  $320 \times 240$ . They were used to simulate the cut-and-paste attack. Figs. 2a and 3a show the original images and their corresponding watermarked images are shown in Figs. 2b and 3b, respectively. The simulations and their respective results are described in the following:

- The watermarked Pool image was tampered by copying one black ball and one white ball from the watermarked image and pasting them into the same image. The tampered image is shown in Fig. 2c. Fig. 2d shows the ground truth of tampered regions. The tamper detection result is shown in Fig. 2e.
- The modification made to the watermarked Field image, was done by cutting a white goat and a black cattle from other watermarked images and paste them in Fig. 3b. The tampered image is shown in Fig. 3c, and the ground truth of tampered regions is shown in Fig. 3d. Fig. 3e shows the tamper detection result.

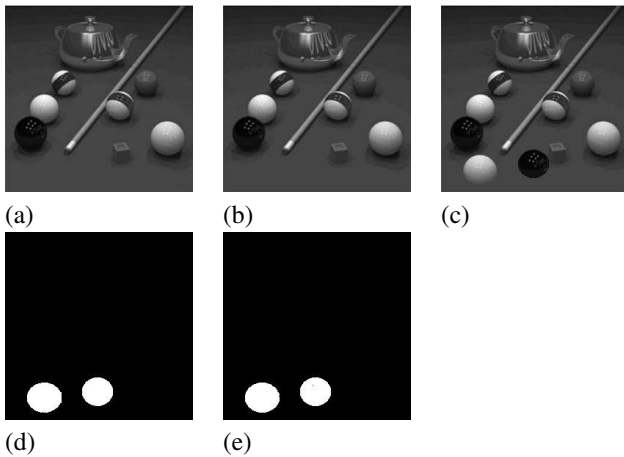


Fig. 2 (a) Original pool image; (b) watermarked pool image; (c) tampered pool image; (d) ground truth image; (e) tamper detection result.

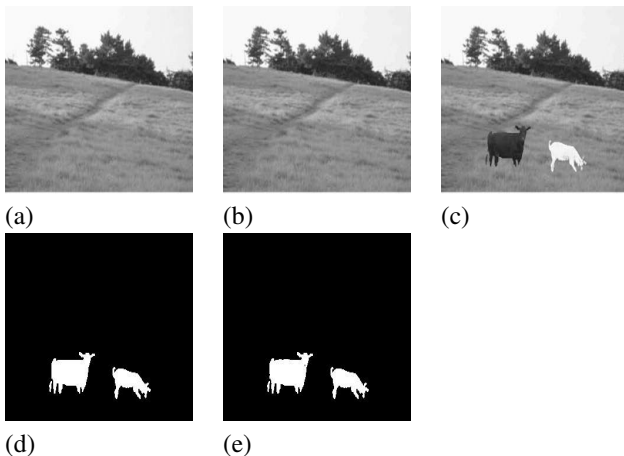


Fig.3. (a) Original field image; (b) watermarked field image; (c) tampered field image; (d) ground truth image; (e) tamper detection result.

2) *Performance under collage attack*: To evaluate the performance under collage attack, a counterfeit image is formed by combining the portions of multiple watermarked images, while preserving their relative spatial location within the target image. Sofa and Doll images given in Figs. 4a, 4b respectively are used to evaluate the performance under the collage attack. The size of both images are  $320 \times 240$  pixels. Figs. 4c and 4d show the corresponding watermarked images. The collage image, as shown in Fig. 4e, was created by copying the three dolls from Fig. 4d and pasting them in Fig. 4c. The ground truth of tampered regions and the tamper detection result are shown in Figs. 4f, 4g respectively.

TABLE I  
COMPARISON RESULTS OF PSNR OF WATERMARKED IMAGES.

Image	Li and Yuan's approach [16]	Chen and Wang's approach [3]	Proposed approach
Pool	44.46	44.48	46.53
Field	44.06	44.07	46.59
Sofa	44.14	44.14	45.99
Doll	44.18	44.20	46.03

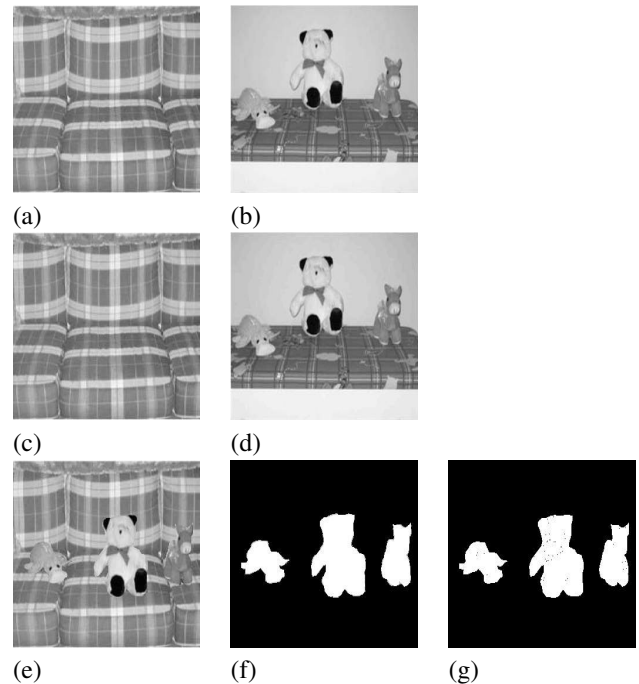


Fig. 4. (a) Original sofa image; (b) original doll image; (c) watermarked sofa image; (d) watermarked doll image; (e) tampered sofa image; (f) ground truth image; (g) tamper detection result.

### B. Performance comparisons and analysis

The two fragile watermarking approaches proposed by Li and Yuan in [15] and Chen and Wang in [3] were provided in this study for performance comparisons.

Table (I) shows the comparison results of image quality, the value of PSNR of watermarked images in the proposed method is greater than the value of PSNR of watermarked images in Li and Yuan's approach and Chen and Wang's approach. It means that the watermarked image of our proposed method have better image quality.

The comparison results of tamper detection are listed in Table (II). The results demonstrate that the proposed approach outperforms the other two approaches in both TP and FP rates. Furthermore, the results show that the proposed approach can completely resist collage attack, as all the blocks which have been modified by collage attack can be correctly identified by our approach.

TABLE II  
COMPARISON RESULTS OF TAMPER DETECTION.

Image	Li and Yuan's approach		Chen and Wang's approach		Proposed approach	
	TP (%)	FP (%)	TP (%)	FP (%)	TP (%)	FP (%)
Pool	74.30	0.91	99.71	0.32	99.72	0.004
Field	74.89	1.33	99.66	0.48	99.71	0.02
Sofa	13.34	2.43	99.30	0.81	98.77	0.11

## V. CONCLUSIONS

In this paper, the proposed fragile watermarking algorithm is presented. It can detect and locate any modification of the embedded image if it is tampered. Realizing that the basic requirement of thwarting counterfeiting attacks is to break blockwise independency, the proposed approach used the RKM clustering technique to create the relationship between image blocks. A series of experiments were conducted to illustrate the validity of the proposed approach.

Experimental results showed that the proposed approach not only can embed authentication data with acceptable visual quality, but also can effectively thwart diverse attacks, such as the cut-and-paste attack and collage attack, while sustaining superior localization accuracy. Furthermore, the results demonstrated that the proposed approach, compared with Li and Yuan's approach which exploited nondeterministic dependence information to resist counterfeiting attacks and Chen and Wang's [5], [11] approach which applied the fuzzy c-means clustering technique to cluster all the image blocks, provided more accurate tamper detection and localization accuracy.

## REFERENCES

- [1] Chang C., Chen K., Lee C., and Liu L., A secure fragile watermarking approach based on chaos-and-hamming code. *The Journal of Systems and Software*, vol.2, pp. 1-9, 2011.
- [2] Chan C. S., and Chang C. C., An efficient image authentication method based on Hamming code. *Pattern Recognition*, vol. 40, pp. 681-690, 2007.
- [3] Chen W. C., and Wang M. S., A fuzzy c-means clustering-based fragile watermarking approach for image authentication. *Expert Systems with Applications*, vol. 36, pp. 1300-1307, 2009.
- [4] Rawat S., and Raman B., A chaotic system based fragile watermarking approach for image tamper detection. *International Journal of Electronics and Communications (AE)*, vol. 16, pp. 1-8, 2011.
- [5] Wang M. S., and Chen W. C., A majority-voting based watermarking approach for color image tamper detection and recovery. *Computer Standards and Interfaces*, vol. 29, pp. 561-570, 2007.
- [6] Hassanien A., Abraham A., Peters J. F., and Kacprzyk J., Rough sets in medical imaging: foundations and trends. *Computational Intelligence in Medical Imaging: Techniques and Applications*, G. Schaefer et al. (Eds.), CRC Press, USA, ISBN 978-1-4200-6059-1, Chapter 3, pp. 47-87, 2008.
- [7] Lingras P., Applications of rough set based K-means, Kohonen SOM, GA clustering. *Transactions on Rough Sets, Lecture Notes in Computer Science*, vol. 2, pp. 120-139, 2007.
- [8] Lingras P., Interval set clustering of web users with rough K-Means. *Journal of Intelligent Information Systems*, vol. 23, pp. 5-16, 2004.
- [9] Pawlak Z., On Rough Sets, *Bulletin of the European Association for Theoretical Computer Science*, no.24, pp. 94-109, 1984.
- [10] Hassanien A., Milanova M., Smolinsk T., and Abraham A., Computational intelligence in solving bioinformatics problems: Reviews, Perspectives, and Challenges. *Computational Intelligence in Solving Bioinformatics Problems*, vol. 151, pp. 3-47, 2008.
- [11] Lin C., and Chang S., A robust image authentication method surviving JPEG lossy compression. *Proceedings of SPIE International conference on storage and retrieval of image/ video database*, vol. 3312, pp. 296-307, 1998.
- [12] Lu C., Liao H., and Sze C., Structural digital signature for image authentication: an incidental distortion resistant approach. *Proceedings of the multimedia security workshop 8th ACM international conference on multimedia*, pp. 115-8, 2000.
- [13] Chang C. C., Hu Y. S., and Lu T. C., A watermarking-based image ownership and tampering authentication approach. *Pattern Recognition Letters*, vol. 27(5), pp. 439-446, 2006.
- [14] Lin E., Podilchuk C., and Delp E., Detection of image alterations using semi-fragile watermarks. *Proceedings of SPIE conference on security and watermarking of multimedia contents*, vol. 2, pp. 152-63, 2000.
- [15] Li C., and Yuan Y., Digital watermarking approach exploiting nondeterministic dependence for image authentication. *Optical Engineering*, vol. 45(12), 2006.
- [16] Zhang X., Wang S., Qian Z., and Feng G., Reversible fragile watermarking for locating tampered blocks in JPEG images. *Signal Processing*, vol. 90, pp. 3026-3036, 2010.
- [17] He H., Zhang J., and Tai H., A neighborhood-characteristic-based detection model for statistical fragile watermarking with localization. *Multimed Tools Appl*, vol. 52, pp. 307-324, 2011.
- [18] Halder R., and Cortesi A., A Persistent Public Watermarking of Relational Databases. *Lecture Notes in Computer Science*, vol. 6503, pp. 216-230, 2010.
- [19] Penga F., Guoa R., Li C., and Long M., A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar transformation. *Computer-Aided Design*, vol. 42, pp. 1207-1216, 2010.
- [20] Qi X., and Xin X., A quantization-based semi-fragile watermarking approach for image content authentication. *J. Vis. Commun. Image R.*, vol. 22, pp. 187-200, 2011.



# Sparse PCA for gearbox diagnostics

Anna Bartkowiak

Institute of Computer Science, University of Wrocław,  
 and Wrocław High School of Applied Informatics  
 Wrocław, Poland

Email: aba@ii.uni.wroc.pl

Radosław Zimroz

Wrocław University of Technology  
 Vibro-Acoustic Science Laboratory  
 Wrocław, Poland

Email: Radoslaw.Zimroz@pwr.wroc.pl

**Abstract**—The paper presents our experience in using sparse principal components (PCs) (Zou, Hastie and Tibshirani, 2006) for visualization of gearbox diagnostic data recorded for two bucket wheel excavators, one in bad and the other in good state. The analyzed data had 15 basic variables. Our result is that two sparse PCs, based on 4 basic variables, yield similar display as classical pair of first two PCs using all fifteen basic variables. Visualization of the data in Kohonen’s SOMs confirms the conjecture that smaller number of variables reproduces quite well the overall structure of the data. Specificities of the applied sparse PCA method are discussed.

## I. INTRODUCTION

NOWADAYS we are witnessing a growing interest in the predictive assessment of industrial machinery. Machines work everywhere and are critical in maintenance of environmental and life conditions of humans. Machines use gearboxes, which in turn are substantial for functioning of the devices. Bad functioning is connected with huge economic losses; therefore early detection of malfunctioning is crucial both from economic and vital aspect. Although there are many general rules how to assess the problem, the working devices are quite different and different approaches are needed. What concerns machinery faults, vibration analysis has become almost the universal method to assess the state of a machine. See [1], [2], [3], [4], [5], [8], [6], [7] for methods of fault analysis used in this domain. Further references may be found in [10].

We will be concerned with diagnosis of gearboxes, used in bucket wheel excavators, working in surface mining. These are huge and expensive machines. In the following we will analyze vibration sounds obtained from two machines: one in bad state (machine A), and one in good state (machine B). We will show that, on the basis of vibration signals, it is possible to assess, what is the state of the machine: good or bad. This will be done by two combined methods: using self-organizing maps (SOMs) and sparse principal components (sPCs). Next section describes how the data were acquired; it contains also some statistical characterization of the recorded data. A preliminary approach to dimensionality reduction - by using Kohonen’s self-organizing maps (SOMs) is shown in Section 3. The principles of constructing classical and sparse PCs are presented in Section 4. Results of applying sparse

PCA to the recorded data, and discussion on the results may be found in Section 5. An overall discussion on the applied methods and their validity is presented in Section 6.

## II. THE DATA

### A. Collection of the data

Data were recorded in an experiment carried out during a research conducted in the Vibro-Acoustics Science Laboratory of Wrocław University of Technology [9]. Two complex multistage gearboxes used in two bucket wheel drive units – working in surface mining – were investigated. The Bruel&Kjaer Pulse system was used in the experiment. The vibration signals were measured in two auxiliary channels (tachometer signal and electric current signal) and four vibration channels. The signal duration was 60s and the sampling frequency 16384 Hz. After data acquisition, the speed profile and the vibration signal data were processed. This was done by (a) signal segmentation (according to digging process cyclicity), and (b) feature extraction.

The recorded vibration signal is shown in the top exhibit of Fig. 1. The exhibit contains two graphs depicting *speed [RPM]* and *acc [m/sec<sup>2</sup>]* (acceleration) viewed as function of *time [sec]*. To achieve stationarity, the recorded signal was cut into segments (two cuts are also shown in the top exhibit of Fig. 1).

Next, each of the obtained segments was subjected to spectral analysis using Matlab function `psd`. This yielded for each segment a power spectrum. Two such power spectra are shown in Fig. 1, bottom exhibit. One may notice some periodicity in appearing of the peaks of the spectrum: all they appear at equally spaced values of mesh frequency (expressed in Hz). This can be explained, taking into account the architecture of the gearbox and its way of working (to learn more, see [9], [8], where also a preliminary analysis of the data may be found).

As result of this stage of data recording and preprocessing, two sets of data vectors were obtained, each with 15 components. Each data vector corresponds to the recorded variables `pp1`,  $\dots$ , `pp15` obtained from the spectral analysis carried out for one segment of the vibration recording. The bad data yielded  $n_A = 1232$ , and the good data  $n_B = 951$  segments. The obtained two sets of data vectors constitute two matrices named A and B. These matrices, of size  $[n_A \times 15]$  and  $[n_B \times 15]$ , are the basis of our further analysis.

\* This work was partially supported by (Polish) State Committee for Scientific Research in 2009 - 2012 as research project by R. Zimroz



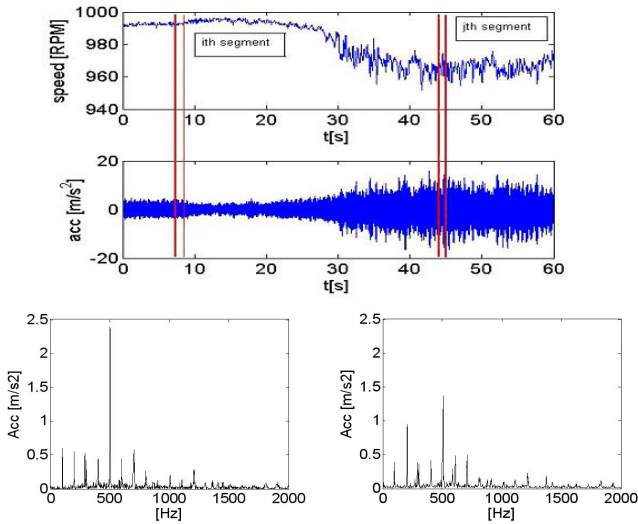


Fig. 1. Data acquisition. Top exhibit: Recorded vibration signals shown as dependence of speed [RPM] and acceleration [m/s<sup>2</sup>] from time [s]. Bottom: Two samples of power spectra, obtained from the Matlab PSD procedure, the amplitude of the spectra is depicted as function of frequency [Hz].

The number of spectral components extracted from a spectrum depends on many factors (design of machine, its condition, external load, signal-to-noise ratio etc). In literature usually 5-6 components were used.

Generally, the obtained variables  $pp1, \dots, pp15$  are expected to indicate whether the device is in bad or good state; in our case they are expected to be associated with state of particular element of the working device, i.e. the planetary stage. It seems that so far this problem (how many power spectra take for the analysis) has not been investigated in a systematic way.

What concerns the damages of the gearbox A, it was found that all the rolling bearings had exceeded the allowable radial backlash and most of the gears had scuffs and micro cracks on their teeth [8].

### B. Preliminary analysis of the data: simple statistics and visualization

Firstly, univariate characteristics such as means, medians and boxplots were calculated. This was done both for raw and normalized data. Each data matrix ( $\mathbf{X}$ ) has been firstly centered to have the mean of each column equal zero, and next normalized so to have the length of each column equal to 1 (this was the way used by [18] and [19] in their sparse PCA). In consequence, every column of  $\mathbf{X}$  has variance equal to  $1/n$ . The boxplots obtained for both raw and normalized data matrices are shown in Fig. 2.

Looking at the boxplots constructed from raw data of sets A and B, one may notice that their distributions differ (in absolute value) considerably in these two sets. For example, in set A, the most characteristic are very large values in variable 5 and 7; while in set B the largest values are exhibited by variable 2 (contains exceedingly large part of outliers) and variable 4.

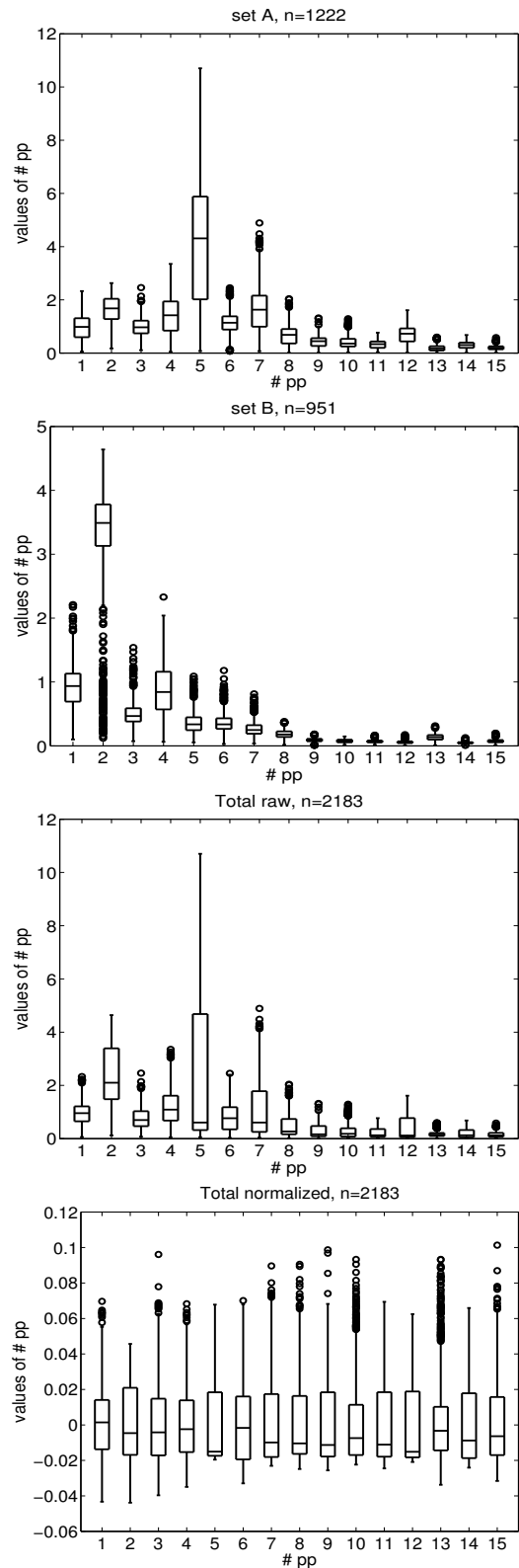


Fig. 2. The top two boxplots depict groups A (bad) and B (good) with raw data, i.e. non-standardized. The bottom two exhibits depict set C containing the sets A and B merged into one common group. Two boxplots are shown for the set C: firstly for raw data, next for normalized data.



On the other hand, looking at the variables no. 9–15, which have very small amplitudes, but differentiated distributions, it is difficult to say something definite about their abilities of differentiation between groups A and groups B.

The medians calculated for all the 15 variables are shown in Fig. 3.

There are two plots constructed for raw and standardized data of both groups. The bottom plot, exhibiting medians for normalized data, is especially interesting. One may notice that:

- (i) variable no. 1 has practically the same value in both groups, therefore it is hard to expect that this variable will contribute to the differentiation between groups A and B; the same can be said about variable no. 13
- (ii) variable no. 2 has inverted value as compared to remaining variables; why does it happen? needs further exploring
- (iii) remaining variables No.s 3 – 12, 14 – 15 exhibit approximately the same difference between their medians, thus any one of them is a candidate for a good discriminator.

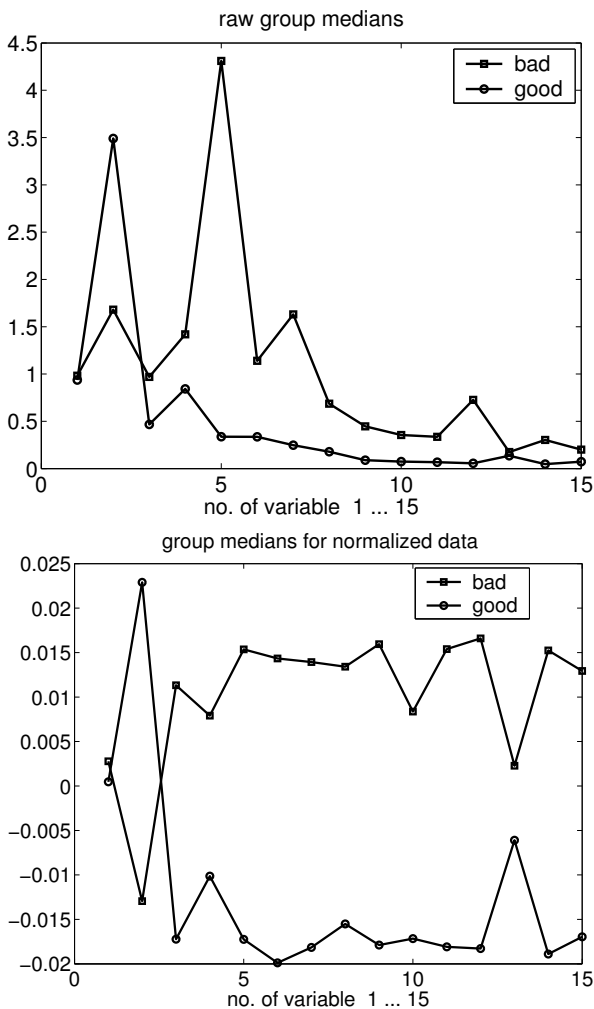


Fig. 3. **Medians** for the raw (top) and normalized (bottom) groups A and B of the recorded data.

### III. MULTIVARIATE DATA VISUALIZATION USING KOHONEN’S SELF-ORGANIZING MAPS

To visualize the considered 15-dimensional data, we use two methods: Canonical discriminant analysis and Kohonen’s self-organizing maps.

Canonical discriminant analysis is a supervised method, it needs as input two data matrices, A and B, containing the two groups of data. The question to answer is: Do the variables pp1 - pp15 have some discriminative power with respect to the bad and good state of the machine? The problem was considered in [10] by using all the 15 variables pp2–pp15 and considering so called Fisher’s LDA criterion. The method yields one canonical discriminant variate which - combined with another one without discriminative power - permits to construct a scatterplot, in which one may see the location of data points belonging to the two considered groups A and B. It happens that these two sets are separated nearly perfectly: only 5 data points are misclassified [10].

Kohonen’s self-organizing maps (the SOM method) are constructed in an unsupervised way. The algorithm obtains as input only one set of data points and tries to depict their mutual neighborhoods in a predefined map. Maps obtained for set C of our data, when considering all 15 variables, and a reduced set of 7 variables, are shown in Fig. 4. First row of maps is based on 15, and the second row on 7 variables.

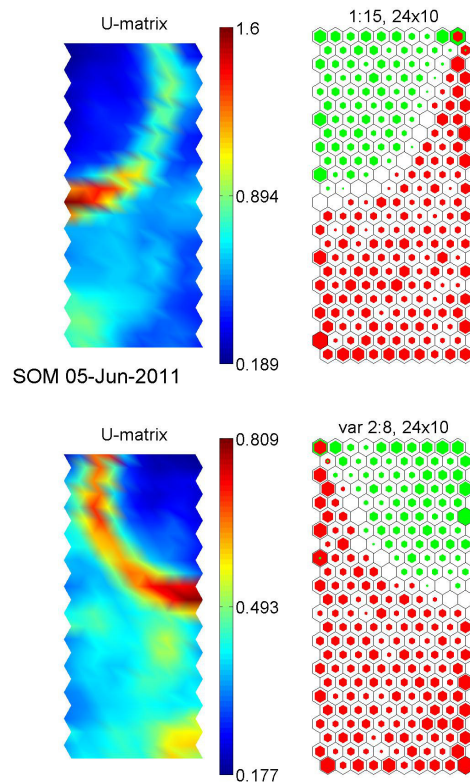


Fig. 4. Kohonen’s self-organizing maps visualizing groups A and B when taking (top) all 15 variables, and (bottom) only variables 2–8. Notice the topological similarity between the two rows of maps.

For construction of the maps we have used the Matlab SOM Toolbox [11]. The architecture of the map is hexagonal; the map is composed from  $24 \times 10$  hexagons. The data were standardized statistically, that is to have zero mean and unit variance for each column.

The maps displayed in Fig. 4 are based on  $24 \times 10$  nodes, which subdivide the total area of the map into 240 sub-areas having the shape of hexagons organized in a grid having 24 rows and 10 columns. To each hexagon a *prototype* vector called by Kohonen a *codebook* vector is found during training of the map. Each codebook represents a certain number ( $n_{kl}$ ,  $k = 1, \dots, 24$ ;  $l = 1, \dots, 10$ ) of input vectors located nearby in the data space. We may find which ones of the data vectors are represented by the given codebook. We may also split each count  $n_{kl}$  into the sum  $n_{kl}^A + n_{kl}^B$ , which shows how many of data points localized at the node  $\langle k, l \rangle$  belong to group A and how many to group B. Having the counts  $n_{kl}^A$  and  $n_{kl}^B$ , we may insert into each hexagon constituting the map two others with size proportional to the counts  $n_{kl}^A$  and  $n_{kl}^B$ , each inserted hexagon painted with different color. The respective maps constructed that way are shown in the right exhibits of Fig. 4. One may notice that data coming from group A and group B are differentiated quite nice.

In the upper map only two nodes (no. 217 and 218) have mixed group content, i.e. represent data vectors belonging either to A or to B.

In the bottom map, the topological locations are inverted with respect to the y-axis (this happens quite often, both maps are topologically equivalent). In this map only two hexagons (no. 1 and 2) have mixed group content, the others are perfectly disjoint. Apart from this, both maps show a decided separation of hexagons containing either bad (group A, color red) or good (group B, color green) data.

We have constructed also another kind of maps, showing directly - by color - the distances of neighboring prototype vectors in the data space. These maps are shown in left exhibits of Fig. 4. The distances are indicated by colors as shown by the adjacent colorbars. One may notice that the two groups are separated quite distinctly by an empty space; also: group A is more homogenous than group B.

The quality of the maps is measured by two indices:  $q$  - *quantization error* and  $t$  - *topographic error* (see [11] for definitions). The respective errors amount:

when using all variables:	$q = 1.075,$	$t = 0.041$
when using variables 2-8:	$q = 0.586,$	$t = 0.040$

The quantization error  $q$  is measuring an average Euclidean distance of data points (located in one node) to their prototype, that is, to their codebook vector. It is obvious, that with more components (variables taken for analysis) this distance is larger. The topographic error  $t$  reflects the fidelity of the topographic representation in the map compared to the true topology in the data space. Surprisingly or not, this error is practically the same for the maps based on 15 and 7 variables.

The above considerations have shown, that the number of considered variables can be reduced, without losing the

essential information about the topological location of the data points. A principled way for finding relevant variables is presented in next section.

#### IV. ORDINARY AND SPARSE PRINCIPAL COMPONENTS

##### A. The classical PCA

Principal component analysis (PCA) is one of the most common and widespread methods for multivariate linear data analysis. It serves for investigating data structure, data mining, data smoothing and approximation, also for exploring data dimensionality. The method permits to build new features, called principal components (PCs), which may serve for visualization of the data [12].

Let  $\mathbf{X}$  of size  $n \times d$  denote the observed data matrix. For simplicity of presentation, assume that  $n > d$ , and that  $\mathbf{X}$  is of full rank. It is advised to standardize or normalize the matrix  $\mathbf{X}$ . Following [18], [19], we assume that the data matrix is column wise centered and has columns of unit length. This means that all columns have means = 0 and variances equal  $1/n$ . The PCA starts from computing the eigenvalues ( $\lambda$ ) and eigenvectors ( $\mathbf{v}$ ) of the cross-product matrix  $\mathbf{S} = \mathbf{X}^T \mathbf{X}$  satisfying the matrix equation  $(\mathbf{S} - \lambda \mathbf{I})\mathbf{v} = \mathbf{0}$ . This results in  $d$  eigenvalues

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d \quad (1)$$

and associated with them  $d$  eigenvectors

$$\mathbf{v}_j = (v_{1j}, \dots, v_{dj})^T, \quad j = 1, \dots, d. \quad (2)$$

The eigenvectors constitute the loading matrix  $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_d]$ . The two fundamental PCA paradigms are:

##### 1) Feature construction:

$$\mathbf{Z}_{n \times K}^{(K)} = \mathbf{X} * [\mathbf{v}_1, \dots, \mathbf{v}_K], \quad 1 \leq K \leq d. \quad (3)$$

The new features identified as columns of  $\mathbf{Z}^{(K)}$  are called **Principal Components** (PCs).

##### 2) Data reconstruction:

$$\hat{\mathbf{X}}_{n \times d}^{(K)} = \mathbf{Z}^{(K)} * ([\mathbf{v}_1, \dots, \mathbf{v}_K]^T). \quad (4)$$

Taking  $K = d$ , the full original data matrix  $\mathbf{X}_{n \times d}$  is reconstructed. For  $K < d$  the best linear approximation of  $\mathbf{X}_{n \times d}$  by a rank- $K$  matrix  $\hat{\mathbf{X}}_{n \times d}^{(K)}$  is obtained - it is best in the meaning of the L2 norm.

The constructed PCs have the major advantage that they are uncorrelated, which permits to analyze each of them separately, without referring to the others.

Principal components can be also computed via SVD:

$$\mathbf{X} = \mathbf{U} \mathbf{D} \mathbf{V}^T$$

where  $\mathbf{Z} = \mathbf{U} \mathbf{D}$  contains the PCs, and  $\mathbf{V}$  the loadings.

##### Applications.

Concerned our domain of interest, that is machine condition monitoring and fault detection: the authors [13] considered some statistical characteristics of vibration signals recorded from an internal-combustion engine sound analysis and an automobile gearbox vibration analysis; after performing a

reduction of dimensionality of the data by PCA they obtained indices suitable for machine condition monitoring. PCA proved also to be very useful in vibration-based damage detection in an aircraft wing and in implementing a procedure for automatic detection and identification of ball bearing faults of some rotating machinery [14], [15].

A detailed analysis of the considered data (sets A, B, C described in Section 2) is shown in [10]. Among others it was found that the first three PCs calculated from set C explain 86 % of total variance of the data. Displaying a scatterplot of the first two PCs (i.e. PC1 and PC2) the points-projections of sets A and B are practically separated and it is possible to draw a linear line separating the two sets with only 5 misclassified data points (out of total  $n_A = 1232$  and  $n_B = 951$  data points).

### B. Sparse PCA

*The idea of sparse PCA:* To construct a PC, one needs all the variables contained in the data set. Is this necessary? It would be desirable to reduce the number of explicitly used variables. This idea was floating for some time among the data analysts and some proposals were offered (see [12], [19], [18]). A reasonable approach with a fast implementation algorithm was proposed in [18]. The authors proposed a regression approach. The idea starts from the observation that each PC is a linear combination of all  $d$  variables. Thus, for known values of a given PC (given as vector  $\mathbf{z}$ , being a column of the matrix  $\mathbf{Z}$  defined in formula 3), and for known values of the matrix  $\mathbf{X}$ , we are able to recover the coefficients of that combination by applying a regression method. If we want a sparse PC (which means that it is based only on few original variables), we should apply here a sparse regression method. The authors of [18] proposed for this purpose the *sparse elastic net regression* [17], [19].

*The computational algorithm proposed by Zou, Hastie et Tibshirani:* The sparse PCs are computed in an iterative way. The computations are for fixed  $K$  ( $1 \leq K \leq d$ ) chosen by the user. The starting point of the reasoning is the observation that the eigenvectors  $\mathbf{v}_j$ ,  $j = 1, 2, \dots, K$  constituting the matrix  $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_K]$  appear in PCA in two roles: when constructing the features (formula 3)

$$\mathbf{Z}^{(K)} = \mathbf{X}\mathbf{V}^{(K)}, \quad (5)$$

or reconstructing the original data matrix (formula 4).

$$\hat{\mathbf{X}}^{(K)} = \mathbf{Z}^{(K)} * (\mathbf{V}^{(K)})^T. \quad (6)$$

Let  $\mathbf{B}$  denote the matrix  $\mathbf{V}^{(K)}$  used in construction of  $\mathbf{Z}^{(K)}$  (formula 5), and  $\mathbf{A}$  the matrix  $\mathbf{V}^{(K)}$  used in reconstruction of  $\hat{\mathbf{X}}$  (formula 6). To obtain sparse PCs, the authors [18] propose to estimate the loading matrix  $\mathbf{V}^{(K)}$  in an alternating way, working in two stages:

*Stage 1.* Given  $\mathbf{A}$ , compute  $\mathbf{B}$ ; and

*Stage 2.* Given  $\mathbf{B}$ , compute  $\mathbf{A}$ .

The initial estimate of  $\mathbf{A}$  may be obtained, e.g. by ordinary PCA.

*Stage 1. Given  $\mathbf{A}$ , compute  $\mathbf{B}$ .* Having  $\mathbf{A}$ , we may perform for each  $j = 1, \dots, K$  the following evaluations:

- compute the values (scores) of the  $j^{\text{th}}$  feature  $Z_j$  appearing in the  $j^{\text{th}}$  column of  $\mathbf{Z}^{(K)}$  as:

$$\mathbf{z}_j^* = \mathbf{X} * \mathbf{a}_j,$$

- notice that the values (scores) of the  $j^{\text{th}}$  feature  $Z_j$  may be also computed as

$$\mathbf{z}_j = \mathbf{X} * \mathbf{b}_j, \quad (7)$$

- notice, that at this moment we know  $\mathbf{z}_j^*$ , and so we may substitute in equation (7):  $\mathbf{z}_j = \mathbf{z}_j^*$ , obtaining

$$\mathbf{z}_j^* = \mathbf{X} * \mathbf{b}_j, \quad (8)$$

- since equation (8) may be viewed as a linear regression problem in unknown regression coefficients vector  $\mathbf{b}_j$ , obtain an estimate of  $\mathbf{b}_j$  by applying a sparse regression algorithm (e.g. `larsen` [17]).

In such way it is enforced that only few columns of  $\mathbf{X}$  will appear in the regression equation, because only few elements of the vector  $\mathbf{b}$  will have non-zero elements. The degree of sparseness of the regression depends on the algorithm. The `larsen` algorithm permits the user to declare, how many original variables (columns of  $\mathbf{X}$ ) should be retained.

Performing the estimation of sparse regression coefficients vectors  $\mathbf{b}_j$  for  $j = 1, \dots, K$  we obtain  $K$  successive vectors composing estimates of the sought matrix  $\hat{\mathbf{B}} = [\hat{\mathbf{b}}_1, \dots, \hat{\mathbf{b}}_K]$  – which was desired at this stage of the calculations.

Now we pass to stage 2.

*Stage 2. Given  $\mathbf{B}$ , compute  $\mathbf{A}$ .*

If  $\mathbf{B}$  is fixed, then the problem is to find such a matrix  $\mathbf{A}$ , that minimizes the quadratic form

$$\|\mathbf{X} - (\mathbf{X}\mathbf{B})\mathbf{A}^T\|^2$$

subject to the restriction that  $\mathbf{A}^T\mathbf{A} = \mathbf{I}_{K \times K}$ .

It is shown in [18] that this is obtained by a reduced rank form of the *Procrustes rotation* by computing the SVD of  $(\mathbf{X}^T\mathbf{X})\mathbf{B}$ :

$$(\mathbf{X}^T\mathbf{X})\mathbf{B} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

and substituting  $\hat{\mathbf{A}} = \mathbf{U}\mathbf{V}^T$ .

Stages 1 and 2 are repeated alternately until a final criterion of convergence is met. The convergence might be achieved if the maximal difference computed for the respective elements  $B_{ik}$  ( $i = 1, \dots, d$ ,  $k = 1, \dots, K$ ) is smaller than an assumed small number  $\epsilon$  (say,  $\epsilon = 1 * e^{-6}$ )

$$|B_{ik}^{old} - B_{ik}^{new}| < \epsilon,$$

where  $\mathbf{B}^{old}$  and  $\mathbf{B}^{new}$  denote the matrices of loadings obtained in two successive iterations. This condition might be combined with another condition that the total number of iterations carried out so far is smaller than a declared maximal number of iterations.

TABLE I

RESULTS OF SPCA FOR THE GEARBOX DATA: CHOSEN VARIABLES AND (ADJUSTED) EXPLAINED VARIANCE AV – FOR 3 VALUES OF  $\lambda$ . VARIANCE EXPLAINED BY 3 FIRST ORDINARY PCs EQUALS 86.05 %.

lambda=1				
chosen	7, 10	1, 4	12, 14	Inversion 3, 2, 1
AV	0.13	0.06	0.02	$\Sigma$ 20.66%
lambda=10				
chosen	12, 14	7, 10	5,8	Inversion 2, 3, 1
AV	0.12	0.03	0.01	$\Sigma$ 15.54%
lambda=100				
chosen	12, 14	7, 10	5,8	Inversion 2, 3, 1
AV	0.12	0.03	0.01	$\Sigma$ 15.54%

## V. SPARSE PCA – RESULTS FOR THE GEARBOX DATA

For calculations we have used the Matlab software implemented by Karl Skoglund and available at <http://www2.imm.dtu.dk/~ksjo/kas/software/index.html> [19]. The software works extremely fast, and being in the form of M-files, can be quite easily adopted to the wishes of its user. Sparse principal components are calculated by the function `sPCA`. The most important parameters to be declared by the user are:  $K$ , the number of desired sparse principal components,  $\lambda$ , parameter for the function `larsen` performing the elastic net regression (see description of stage 2 of the algorithm presented in Section 4), `stop`, a parameter specifying some stopping criteria, containing also the option how many non-zero components to retain in each loading vector (eq. 2) of the retained PCs. The algorithm relies heavily on the function `larsen` realizing the algorithm for elastic net regression [18], [19], being an extension of the least angle regression [16], [17].

Assuming  $K=3$  (compute 3 sparse principal components),  $stop=2$  (each PC uses only 2 original variables) and  $\lambda=1$  ( $\lambda$  is a parameter needed by the function `larsen` called in Stage 1 of the algorithm) we got for our data the (constructed) feature matrix  $\mathbf{Z}_{n \times 3}$  containing in its columns three sparse principal components:  $\langle PC1, PC2, PC3 \rangle$ . They are depicted in Fig. 5.

The displays look much similar to those obtained by ordinary PCA (shown in [10]).

Similar displays (not shown here) were obtained when using  $\lambda=10$  and  $\lambda=100$ . Details of the results, for the same values of  $K$  ( $=3$ ),  $stop$  ( $=2$ ) but different values of  $\lambda$  are shown in Table 1.

When analyzing the results, one should consider:

- (i) Which original variables were chosen by the applied `sPCA` algorithm to constitute the sparse principal components?
- (ii) How much of total variance is explained by the constructed sparse PCs?

Before discussing the results, we should emphasize that the applied method yields sparse PCs that have different properties than the classic ones. This happens because the classical PCs are set with the criterion of extracting the maximum amount of total variance of the data, while in construction of sparse PCs this principle is *de facto* not used. To say it plainly, the

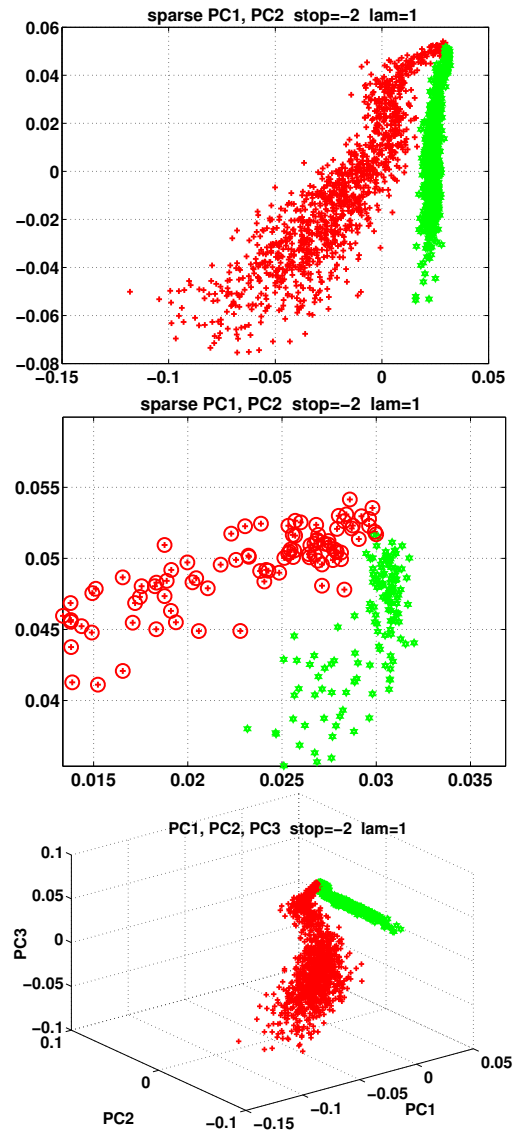


Fig. 5. Sparse principal components PC1, PC2 and PC3. Legend: a red '+' denotes 'bad' data point (set A); a green hexagon denotes 'good' data point (set B). Top: scatterplot of  $\langle PC1, PC2 \rangle$ . Middle: Zoom of the overlapping parts of the 'bad' and 'good' data clouds, the bad points, i.e. red pluses, are additionally circled. Bottom: 3D display of  $\langle PC1, PC2, PC3 \rangle$ .

columns of  $\mathbf{B}$  (used for construction of  $\mathbf{Z}$ ) are based on other principle: the sparse regression. This makes that the variances of the sparse PCs need not decrease with the no. of the PC; also that the variances of the sparse PCs do not need to sum to the total variance, given as  $trace$  of  $\mathbf{S} = \mathbf{X}^T \mathbf{X}$ . To make them act similarly like the classical PCs, after convergence of the 2-stage algorithm the following actions are undertaken :

- $\alpha$ ) normalization of the loadings appearing in  $\mathbf{B}$ , to have unit Euclidean length.
- $\beta$ ) ordering of the columns of the  $\mathbf{Z}$  matrix according to their decreasing variance, and next orthogonalizing them – to have adjusted increments (AV) of explained variance.

Now let us look at Table 1. We find that

- Considering point (i) formulated above: The variables

chosen for two pairs of the sparse PCs are the same for  $\lambda=1$  and  $\lambda=2$ , but one pair of chosen variables differs.

- Considering point (ii) formulated above: An inversion happened in finding the PCs with the largest variances. In the experiment using 3 different values of  $\lambda$ , the sparse PC explaining the maximum variance was found as the third (for  $\lambda=1$ ) or the second (for  $\lambda=10$  and  $\lambda=100$ ).

- Generally, the total percentage of explained variance called AV (adjusted variance) is low, however that the graphically displayed information of the data is practically the same as that obtained from first three classical PCs, which reconstruct together 86.05% of total variance. This leads to the conclusion that the sparse PCA is able to recover the proper information on the topological structure of the data. This is interesting and needs further elaboration. Probably this is due to the specificity of our data. It was noticed already in Section 2 that there are several variables which repeat the same information. The classical PCA is able to notice them at once and account all their information into the first PCs. The sparse PCA algorithm deals only with part of the variables and constructs dependent PCs. However, despite this, the sparse PC algorithm is able to recover the proper information on the topological structure of the analyzed gearbox data.

## VI. DISCUSSION AND CONCLUDING REMARKS

The necessity and usefulness of producing sparse PCs was considered since long as part of so called *factor analysis* seeking to group the analyzed variables into some subgroups which appear jointly indicating for a hidden, non-measurable variable which manifests itself through the observed and measurable phenomena [12] (e.g. 'intelligence' is such a hidden variable). One very popular method consisted of performing rotations of the loading matrix obtained from PCA (e.g., the varimax rotation). Other method acted by determining simple thresholds of elements of the loading matrix; e.g. elements smaller than 70% of the maximum from all elements were simply set to zero.

The sparse algorithm proposed in [18] is mathematically very sophisticated. It can be implemented to work effectively and fast when using Matlab [19]. The algorithm is relatively new and not very much in use. We have applied this algorithm with the aim of reducing the number of variables needed when making a diagnosis of gearbox functioning. It seems that we were successful and the set of used variables can be reduced. However, we have noticed that the results of analysis depend from few initial parameters needed by the spca algorithm. For the moment the parameters are found by a kind of cross-validation. The results may also depend on the structure of the data. Therefore, before advising the algorithm for an automated procedure, some further research on work of the algorithm is needed.

SCotLASS is another interesting method, developed about the same time [20]. It seems to be easier in determining the input parameters for a sparse PCA, however the numerical optimization of the applied criterion is more complex.

SCoTLASS has elucidated a difficult problem connected with structure of Yeast DNA [21], which could not be identified previously.

## REFERENCES

- [1] R.B. Randall, "State of the art in monitoring rotating machinery – Part 1", *Sound and Vibration*, March 2004, pp. 14–20.
- [2] R.B. Randall, "State of the art in monitoring rotating machinery – Part 2", *Sound and Vibration*, May 2004, pp. 10–16.
- [3] R.B. Randall and J. Antoni, "Rolling element bearing diagnostics – A tutorial", *Mechanical Systems and Signal Processing* Vol. 25, 2009, pp. 485–512.
- [4] K. Worden, W.J. Staszewski, J.J. Hensman, "Natural computing for mechanical system research: A tutorial overview". *Mechanical Systems and Signal Processing* Vol. 25, 2009, pp. 4–111.
- [5] D.A. Clifton, L.A. Clifton and P.R. Bannister, "Automated novelty detection in industrial systems". *Studies in Computational Intelligence (SCI)* Vol. 116 (www.springerlink.com) 2008, pp. 269–296.
- [6] A. Bartkowiak and R. Zimroz, "Outlier analysis and one class classification approach for planetary gearbox diagnosis", (*9th Int. Conf. on Damage Assessment of Structures DAMAS 2011*, Oxford UK.), *J. Phys. Conf. Ser.* vol. 305, 2011, 012031, pp. 1–10, IOP Publishing, doi:10.1088/1742-6596/305/1/012031
- [7] R. Zimroz and A. Bartkowiak, "Investigation on spectral structure of gearbox vibration signals by principal component analysis for condition monitoring purposes" *J. Phys. Conf. Ser.* vol. 305, 2011, 012075, pp. 1–10, IOP Publishing, doi:10.1088/1742-6596/305/1/012075
- [8] W. Bartelmus, F. Chaari, R. Zimroz and M. Haddar, "Modelling of gearbox dynamics under time-varying nonstationary load for distributed fault detection and diagnosis", *European J. of Mechanics A/Solids* Vol. 29, 2010, pp 637–646.
- [9] W. Bartelmus and R. Zimroz, "A new feature for monitoring the condition of gearboxes in non-stationary operating conditions", *Mechanical Systems and Signal Processing* Vol. 23, 2009, pp. 1528–1534.
- [10] R. Zimroz and A. Bartkowiak, "Two simple multivariate procedures for monitoring planetary gearboxes in nonstationary operating conditions". Manuscript 2011, pp. 1–18, unpublished.
- [11] J. Vesanto, J. Himberg, E. Alhoniemi and J. Parhankangas, SOM Toolbox for Matlab 5. Som Toolbox team, Helsinki University of Technology, Finland, Libella Oy, Espoo 2000, pp. 1–54.
- [12] I.T. Jolliffe, *Principal Component Analysis*, 2nd Edition, Springer, New York 2002.
- [13] Q. He, R. Yan, F. Kong and R. Du, "Machine condition monitoring using Principal Component representations", *Mechanical Systems and Signal Processing* Vol. 23, No. 2, 2009, pp. 446–466.
- [14] I. Trendafilova, M. Cartmell, W. Ostachowicz, "Vibration based damage detection in an aircraft wing scaled model using principal component analysis and pattern recognition", *Journal of Sound and Vibration* 313, 3-5, 2008, pp. 560-566.
- [15] I. Trendafilova, "An automated procedure for detection and identification of ball bearing damage using multivariate statistics and pattern recognition", *Mechanical Systems and Signal Processing* Vol.24, 2010, pp. 1858-1869.
- [16] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani "Least angle regression", *Annals of Statistics*, Vol. 32, No. 2, 2004, pp. 407–451.
- [17] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning. Data Mining, Inference and Prediction*. 2nd Edition, New-York, Springer 2010.
- [18] H. Zou, T. Hastie, R. Tibshirani. "Sparse principal component analysis". *J. of Computational and Graphical Statistics*, Vol. 15, no. 2, 2006, pp. 265–286.
- [19] K. Sjöstrand, M.B. Stegmann and R. Larsen, "Sparse principal component analysis in medical shape modeling". *International Symposium on Medical Imaging 2006*, San Diego, CA, USA, Proc. SPIE 6144, 61444X, 2006, pp. 1–12, doi:10.1117/12.651658
- [20] N.T. Trendafilov and I.T. Jolliffe, "Projected gradient approach to the numerical solution of the SCoTLASS". *Computational Statistics and Data Analysis* Vol. 50, 2006, pp. 242–253.
- [21] A. Bartkowiak and N.T. Trendafilov, "Feature extraction by the SCoTLASS: an illustrative example". In: *Intelligent Information Processing and Web Mining*, Springer Series 'Advances in Soft Computing' Vol. 31, 2005, pp. 3-11, DOI: 10.1007/3-540-32392-9\_1.





# CWJess: Implementation of an Expert System Shell for Computing with Words

Elham S. Khorasani, Shahram Rahimi, Purvag Patel, Daniel Houle  
Department of Computer Science  
Southern Illinois University Carbondale  
Carbondale, IL, USA

**Abstract**—Computing with Words (CW) is an emerging paradigm in knowledge representation and information processing. It provides a mathematical model to represent the meaning of imprecise words and phrases in natural language, and to perform reasoning on perceptual knowledge. This paper describes a preliminary extension to Jess, CWJess, which allows reasoning in the framework of Computing with Words (CW). The resulting inference shell significantly enhances the expressiveness and reasoning power of fuzzy expert systems and provides a Java API which allows users to express various types of fuzzy concepts, including: fuzzy graphs, fuzzy relations, fuzzy arithmetic expression, and fuzzy quantified propositions. CWJess is fully integrated with jess and utilizes jess Rete network to perform a chain of reasoning on fuzzy propositions

**Index Terms**—Computing with Words; fuzzy Logic; Expert systems; knowledge representation

## I. INTRODUCTION

HUMAN mind has a limited capability for processing a huge amount of detailed information in his environment; thus, to compensate, the brain groups together the information it perceives by its similarity, proximity, or functionality and assigns to each group a name or a “word” in natural language. This classification of information allows human to perform complex tasks and make intelligent decisions in an inherently vague and imprecise environment without any measurements or computation. Inspired by this human capability, Zadeh introduced the machinery of CW as a tool to formulate human reasoning with perceptions drawn from natural language and argued that the addition of CW theory to the existing tools gives rise to the theories with enhanced capabilities to deal with real-world problems and makes it possible to design systems with higher level of machine intelligence [1][2]. To do this, CW offers two principal components, (1) a language for representing the meaning of words taken from natural language, this language is called the Generalized Constraint Language (GCL), and (2) a set of deduction rules for computing and reasoning with words instead of numbers. CW is rooted in fuzzy logic; however, it offers a much more general methodology for fusion of natural language propositions and computation with fuzzy variables. CW inference rules are drawn from various fuzzy domains, such as fuzzy logic, fuzzy arithmetic,

fuzzy probability, and fuzzy syllogism. This paper reports a preliminary work on the implementation of a CW inference system on top of JESS expert system shell (CWJess). The CW reasoning is fully integrated with JESS facts and inference engine and allows knowledge to be specified in terms of GCL assertions.

The current fuzzy logic expert system shells, such as: fuzzyclips [3], fuzzyjess [4], FLOPS [5], and Frill [6] are much devoted to implementing Mamdani inference system and have left out reasoning with other fuzzy concepts such as: fuzzy relations, fuzzy arithmetic, fuzzy quantifiers, and fuzzy probabilities. This paper presents a roadmap to implement a CW expert system shell capable of representing and reasoning with such concepts. The resulting CWJess expert system shell would allow users to express their knowledge in form of fuzzy quantified propositions, discrete fuzzy relations, and fuzzy arithmetic expressions as well as fuzzy if-then rules and enables them to perform advanced fuzzy reasoning.

## II. PRELIMINARY: COMPUTING WITH WORDS

This section provides a very brief introduction to computing with words, the generalized constraint language, and CW inference rules. More detailed information can be found in Zadeh’s paper [7].

The core of CW is to represent the meaning of a proposition in form of a generalized constraint (GC). The idea is that a majority of the propositions and phrases used in natural language can be viewed as imposing a constraint on the values of some linguistic variables such as: time, price, taste, age, relation, size, appearance, and etc. For example the sentence: “most Indian foods are spicy” constrains the two variables: (1) the taste of Indian food, and (2) the portion of the Indian foods that are spicy. In general, a GC is in form of:

$$X \text{ is } r \text{ R}$$

Where X is a linguistic (or constrained) variable whose values are constrained by the linguistic term R, and the small r shows the semantic modality of this constraint, i.e., how X is related to R. Various modalities are introduced in the literature of CW, among them are:

- possibility ( $r=blank$ ): where  $R$  is a fuzzy set which denotes the possibility distribution of  $X$  [8], e.g., “ $X$  is large”.
- fuzzy graph ( $r=fg$ ): where  $X$  is a function of another variable, say  $Y$ , and  $R$  is a fuzzy estimation (or granulation) of that function. This modality corresponds to a collection of fuzzy if-then rules that share the same variables in their premises and consequences. e.g., “ $X=f(y)$  isfg (small  $\times$  large + medium  $\times$  medium + large  $\times$  small)”, which is equivalent to three fuzzy rules: *if  $Y$  is small then  $X$  is large, if  $Y$  is medium then  $X$  is medium and if  $Y$  is large then  $X$  is small.*
- probability ( $r=p$ ): where  $X$  is a random variable and  $R$  is the probability distribution of  $X$ , e.g., “( $X$  is large) isp likely”.
- usuality ( $r=u$ ): where  $X$  is a random variable and  $R$  is the usual (or typical) value of  $X$ , e.g., “ $X$  isu big”

A collection of GCs together with a set of logical connectives (such as: and, or, implication, and negation) and a set of inference rules form the generalized constraint language (GCL). The inference rules regulates the propagation of GCs. Table 1 lists instances of inference rules introduced for GCL. As shown in this table, each rule has a symbolic part and a computational part. The symbolic part shows the general abstract form of the GCs that appear in the premises and conclusion of a rule, while the computational part calculates the fuzzy value of the consequent of the rule based on its premises.

TABLE I. INSTANCES OF CW INFERENCE RULES

Inference rule	Symbolic part	Computational part
Conjunction Rule	$x$ is $A$ $y$ is $B$ $(x, y)$ is $C$	$\mu_C(u, v) = \mu_A(u) * \mu_B(v)$
Extension Principle	$X$ is $A$ $f(X)$ is $B$	$\mu_B(z) = \sup_u (\mu_A(u))$ subject to : $z = f(u)$
Compositional Rule of Inference	$X$ is $A$ $(Y, X)$ is $B$ $Y$ is $C$	$\mu_C(v) = \sup_u (\mu_A(u_i) * \mu_B(v, u))$
Fuzzy graph Interpolation	$\sum_i$ if $X$ is $A_i$ then $Y$ is $B_i$ $X$ is $A$ $Y$ is $B$	$\mu_B(v) = \sup_i (m_i * B_i)$ $m_i = \sup_u (\mu_A(u) * \mu_{A_i}(u))$ $i = 1 \dots n$
Fuzzy Syllogism	$Q_1 A$ 's are $B$ 's $Q_2(B)$ 's are $C$ 's $Q_3 A$ 's are $(C)$ 's	$\mu_{Q_3}(z) = \sup(\mu_{Q_1}(w_1) * \mu_{Q_2}(w_2))$ subject to : $z = w_1 * w_2$ $w_1, w_2$ , and $z$ are the universes of discourse of $Q_1, Q_2$ , and $Q_3$ , respectively

### III. INCORPPRATING CW REASONING IN JESS

Figure 1 shows a schematic view of CWJess Inference System.

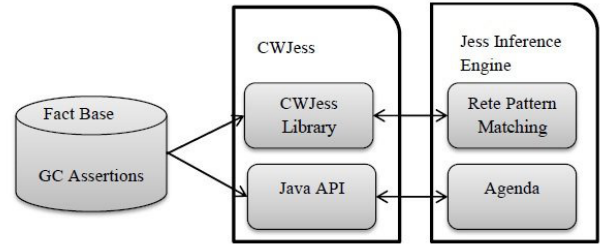


Figure 1. The schematic view of CWJess Inference System

CWJess consists of two components: the Java API and the CWJess library. The Java API is the core element which implements CW concepts, such as linguistic variables, various types of generalized constraints, and *the computational part* of CW inference rules. The CWJess library consists of a list of production rules which define the *symbolic part* of CW inference rules. When a user adds a GCL assertion to the fact base, Jess Rete algorithm checks the symbolic part of CW inference rules in CWJess library to find a production rule whose premises match with the facts and place the consequents of such rules into the agenda. When a CW inference rule is fired, its consequent calls the java method which performs the computational part of the rule and asserts the result back to the fact base.

#### A. GC Assersions in CWJess Fact Base

The CWJess fact base consists of a set of GC assertions. A GC assertion may have one of the following forms:

- (1) a simple generalized constraint,
- (2) a fuzzy graph,
- (3) a quantified generalized constraint
- (4) an arithmetic generalized constraint.

Different forms of generalized constraints are implemented as java classes in the java API. The GC facts are then created as jess shadow facts connected to the corresponding java object.

A simple generalized constraint consists of four elements:

- An atomic or composite linguistic variable, such as: “age”, “size”, “weight”, “price”, “distance”, etc.
- A list of objects of linguistic variables, for example object “Mary” for the linguistic variable “Age”, or object “McDonald’s”, for the linguistic variables “service-quality”.
- A semantic modality. The semantic modality may be: “possibistic”, “probabilistic” “verisitic”. The semantic modality is by default possibilistic. At this point, we have only implemented the possibilistic semantic modality. The implementation of probabilistic and veristic modalities and their combination with possibilistic modality requires the availability of more complex theories and will be considered as future work of this study.



- A linguistic value associated with the linguistic variable and constraint its values.

For example, the generalized constraint: *service-quality(McDonald's) is good* consists of the linguistic variable: *service quality*, object: *McDonald's*, semantic modality: *possibilistic*, and linguistic value: *good*.

Two types of linguistic variables are considered: *atomic* and *composite*. The atomic linguistic variable consists of a name (e.g., oil-price), a unit (e.g., \$ per gallon), a range of values (e.g. 1.0-5.0), and a set of linguistic terms (e.g., “cheap”, “mid-price”, and “expensive”). The linguistic terms are defined using a term name and a membership function that is a fuzzy set over the range of values for the linguistic variable. The membership functions may be defined to have a standard shape, such as : triangular, trapezoid, Gaussian, Pi shape, S shape, Z shape, crisp interval, or it may be a piecewise linear, or a discrete function specified by a set of singletons. Figure 2 shows three linguistic terms with different type of membership function for the linguistic variable: “oil-price”.

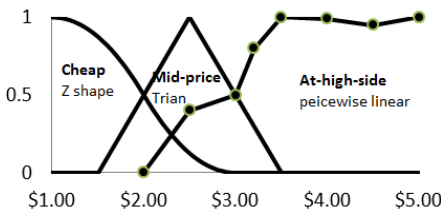


Figure 2. Examples of membership functions for the linguistic variable: oil\_price

A composite linguistic variable is a fuzzy relation between two or more linguistic variables. To reduce the complexity of computation, the membership function of composite linguistic variable is assumed to be discrete. The inclusion of continuous fuzzy relation requires complex non-linear computation, and will be considered as the future developments of CWJESS. Figure 3 shows the membership function for the linguistic term “petite” of the composite linguistic variable, “size”, where “size” is a fuzzy relation of two atomic linguistic variables: “height” and “weight”.

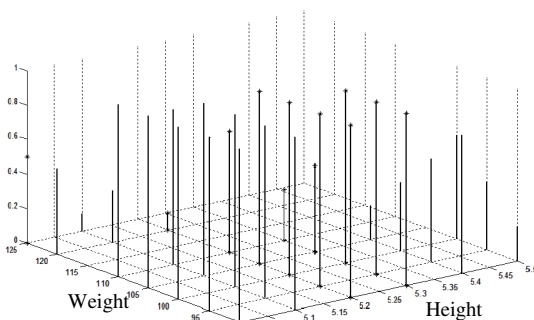


Figure 3. The membership function for the linguistic term: “petite” of the composite linguistic variable “size”.

A linguistic value in a generalized constraint may be modified by a fuzzy modifier. The modifiers implemented in

CWJESS are: *not*, *more-or-less*, *a-little-more*, *slightly-more*, *somewhat*, *very*, *extremely*, *indeed*, and their combinations.

A Fuzzy graph is a collection of fuzzy if...then rules in which all the premises and the conclusion share the same linguistic variables. The general form of a fuzzy graph in CWJESS is as follows:

$$\sum_{i=1..n} \text{if } (GC_{i1} \wedge GC_{i2} \dots \wedge GC_{in}) \text{ then } GC_{ik}$$

Where  $GC_{i1}, GC_{i2}, \dots, GC_{in}$  denote the generalized constraints in the premise of the rule, and  $GC_{ik}$  is the generalized constraint in the consequent of the rule. For example:

*if age(x) is young  $\wedge$  health(x) is good then insurance(x) is very low*  
*if age(x) is middle-age  $\wedge$  health(x) is good then insurance(x) is average*  
*if age(x) is old  $\wedge$  health(x) is moderate then insurance(x) is relatively high*  
*if age(x) is old  $\wedge$  health(x) is poor then insurance(x) is very high*

Although fuzzy graph may seem as the conjunction of a set of fuzzy if-then rules (or fuzzy implications), it has a very different meaning. From the mathematical point of view, a fuzzy graph expresses a functional dependency between two or more linguistic variables and provides a fuzzy description of such function when the point to point data is not available. Hence, fuzzy graph has a completely different semantics than the fuzzy implication and must be treated differently by the inference engine.

A quantified generalized constraint (QGC) consists of a fuzzy quantifier and a generalized constraint with an anonymous object (variable) which is bounded by the quantifier. For example: “most<sub>x</sub> price(x) is expensive” is a quantified generalized constraint in which  $x$  is an anonymous object bounded by the quantifier “most”. Fuzzy quantifiers are defined as fuzzy sets and are assigned a membership function:  $\mu : [0,1] \rightarrow [0,1]$ . In this version of CWJESS we limited ourselves to monotone increasing quantifiers, such as : “most”, “many”, “several”, “a few”, etc., for applying Zadeh’s syllogism reasoning [9].

We have implemented two types of QGC: unary and binary. A unary QGC puts constraint on the proportion of objects that satisfy a single generalized constraint, e.g., “most<sub>x</sub> (age(x) is young)” whereas a binary QGC imposes a constraint on the proportion of number of objects that satisfy one. For example: “most<sub>x</sub> ( age(x) is young, health(x) is good)” is a binary QGC which states that the number of “x” s who are young and healthy over the number of “x”s who are young, is most.

An arithmetic generalized constraint (AGC) is a generalized constraint which has an arithmetic fuzzy expression as its linguistic value. The arithmetic expression may consist of linguistic terms or other linguistic variables. For example: “gas-price(Europe) is gas-price(US) + approximately \$4 per gallon”, which states that the gas price in Europe is approximately \$4 more per gallon than in the United States.

**B. Implementation of CW Inference Rules**

The computational part of all CW rules is implemented in the java API as the following java methods. The result of the each method is a normalized fuzzy set.

- **Conjunction method:** The conjunction method takes a number of linguistic terms associated with a linguistic variable, and computes their minimum intersection. (The minimum operation can be easily replaced with a t-norm [10])
- **Disjunction method:** The disjunction method takes a number of terms associated with a linguistic variable and computes their maximum union. The maximum operation can be replaced with a t-conorm.
- **Compositional method:** The compositional method takes two linguistic terms: one associated with a composite linguistic variable and the other one associated with an atomic linguistic variable, which occurs in the composite linguistic variable, and calculates their max-min composition. As an example let us assume that the composite linguistic variable size of a woman consists of the atomic linguistic variables, height and weight, i.e.,

$$size(x) = (height(x), weight(x))$$

let's also assume that the membership function for the term "petite" associated with size, and the term "about 5.2" associated with the height are given. The compositional method calculates the membership function of the weight of a petite woman who is about 5.2 tall, as shown in the following figure.

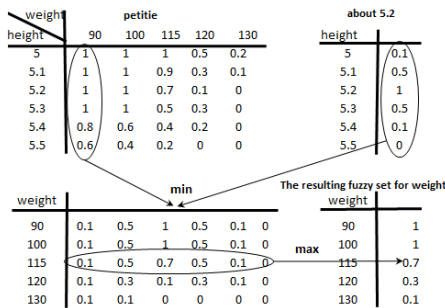


Figure 4. Compositional rule of inference. Composition of the fuzzy relation "petite" with the fuzzy set "about 5.2". the

- **Fuzzy graph Interpolation method:** This method corresponds to Mamdani method of inference. It takes a fuzzy graph as well as a set of linguistic terms associated with the linguistic variables in the premises of a fuzzy graph and computes the corresponding fuzzy set for the linguistic variable in the conclusion. Unlike Mamdani inference method, where the input is a singleton, the fuzzy interpolation method accepts a linguistic term (fuzzy set) as input and, for each rule, computes the match score as the degree of similarity between the input fuzzy set and the ones in the premise of the rule. For instance, consider the fuzzy graph mentioned as an example in the previous subsection, and let us assume that the input it: "age is about 55" and "health is "excellent", then the fuzzy graph interpolation

method would calculate the fuzzy set for the linguistic variable "insurance" as illustrated in figure 5.

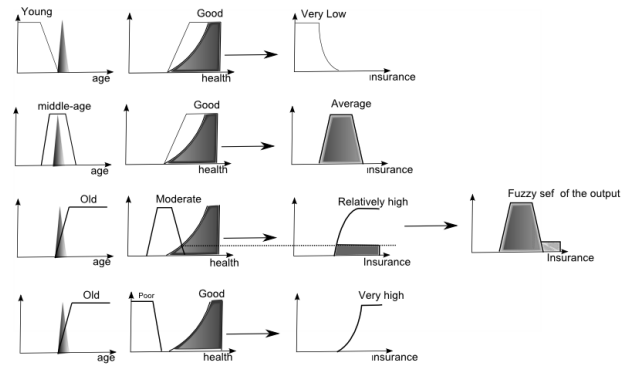


Figure 5. Example of fuzzy graph interpolation rule. The gray areas in the first and second column show the membership functions of "about 55", and "excellent" for the linguistic variables "age" and "health", respectively.

- **Fuzzy Extension method:** The fuzzy extension method, takes an arithmetic expression on a number of linguistic terms associated with a linguistic variable, and returns the fuzzy set resulted from performing the arithmetic operations. The linguistic terms appear in the arithmetic expressions are required to have a normal convex fuzzy set.

The implementation of the extension principle is not trivial and involves nonlinear optimization. Thus approximation methods are usually used to obtain the membership function of the resulting fuzzy set. A common practice is to discretize the membership interval [0,1] into a finite number of values and, for each value, take the  $\alpha$ -cut of all the operands. The arithmetic operations then may be performed on the resulting intervals, using the interval arithmetic, in order to come up with the  $\alpha$ -cut of the output fuzzy set. Finally, these  $\alpha$ -cuts are put together to obtain the output of the arithmetic operations. This approach can be efficiently implemented and provides a good approximation to the exact solution of the extension principle [11]. Figure 6 demonstrates the  $\alpha$ -cut method for computing the result of the arithmetic expression: "around3 + approximately4 \* about-half", for the linguistic variable "oil\_price".

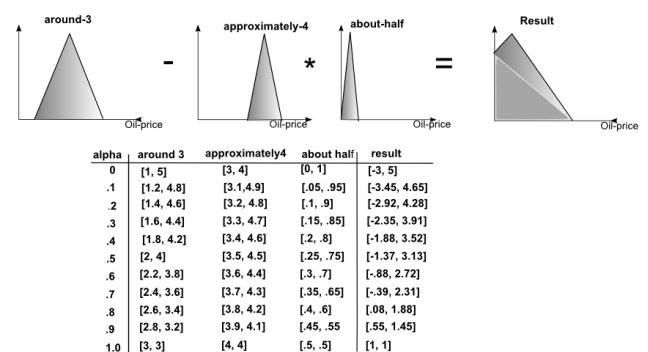


Figure 6. An example of alpha-cut implementation of the extension principle. The table shows the  $\alpha$ -cuts of the operands and the output of the arithmetic expression

- **Fuzzy Syllogism:** The fuzzy syllogism method takes two monotone increasing fuzzy quantifiers, and calls the extension method to calculate and return their multiplication.

The symbolic parts of the above rules are implemented as production rules in the CWJESS library. For example, the symbolic part of the conjunction rule may be defined as follows:

```
;; The conjunction rule
(defrule conjunction
  (declare (salience 100))
  ?c1<-(GC (linguistic_var ?x ) (object ?a)
    (linguistic_value ?w))
  ?c2<-(GC (linguistic_var ?x) (object ?a)
    (linguistic_value ?z&:(<> ?z ?w)))
=>
  (assert (GC (linguistic_var ?x) (object ?a)
    (value (CWRULES.conjunction ?w ?z))))
  (retract ?c1 ?c2))
```

This rule states that if there are two different generalized constraints in the fact base sharing the same linguistic variable and the same object, they are combined to one generalized constraint whose value is the conjunction of the linguistic values of the premises. GC is a template automatically created from the generalized constraint class in the java API. GCWRULES is a java class which implements the computational parts of CW rules as described earlier.

As another example, consider the compositional rule of inference, the symbolic part of this rule is defined in CWJESS library as follows:

```
;; The compositional rule of inference.
(defrule composition
  (GC (linguistic_var ?u &:(instanceof ?u
    COMPOSITE_LINGUISTIC_VARIABLE))
    (object ?o) (linguistic_value ?w))
  (GC (linguistic_var ?x &:(instanceof ?x
    ATOMIC_LINGUISTIC_VARIABLE))
    (object ?o) (linguistic_value ?z))
  (test (occurs ?x ?u )
    =>
  (assert (GC (linguistic_var (CWRULES.composedVar ?u
    ?x)) (object ?o) (linguistic_value
    (CWRULES.composedValue ?w ?z ))))
```

This rule states that if there are two generalized constraints in the fact base, one with a composite linguistic variable ?u, and another one with an atomic linguistic variable ?x, and ?x occurs in ?u, then assert a new generalized constraint which is the composition of the two generalized constraints in the premises. The methods composedVar, and composedValue are static methods defined in CWRULES and return, respectively, the linguistic variable and the fuzzy set resulted from the composition.

The symbolic parts of the fuzzy graph interpolation rule, fuzzy syllogism, and fuzzy extension principle, are defined in a similar way but are excluded from the paper due to their complexity and length.

#### IV. AN EXAMPLE

To demonstrate the CWJESS we present a simple example.

Suppose that we would like to encode the following information in a CWJESS fact base:

- *The average chance that a woman is diagnosed by breast cancer depends directly on her age. The younger a woman is, the lower is her risk of developing a breast cancer.*
- *Many obese women have higher risk of developing a breast cancer and a Mammogram is strongly recommended for most women at high risk of breast cancer.*
- *Mary has a son who is about 15. She gave birth to her son when she was in her 20's. Also She is few years younger than Ann who is in her mid-50.*

The dependency between the age and the risk of breast cancer should be represented as a fuzzy graph. To create a fuzzy graph, the related linguistic variables and their associated terms must be first defined. In the following Jess code, "trap-mf", "tri-mf", "z-mf", "p-impf", "s-mf", and "interval-mf" denote trapezoid, triangular, z shape, pi shape, s shape, and the crisp interval membership functions, respectively.

```
;; Defining the linguistic variables and terms that
;; appear in the fuzzy graph
(bind ?age (new ATOMIC_LINGUISTIC_VARIABLE "age"
  "year" 0 120 ))
(?age addTerm "young" "(0,1) (25,1) (50,0)" )
(?age addTerm "middle-age" "trap-mf" "30 40 50 60")
(?age addTerm "old" "(40, 0) (65 1) (120,1)" )

(bind ?riskbc (new ATOMIC_LINGUISTIC_VARIABLE
  "risk-breast-cancer" "percentage" 0 100)
  (?riskbc addTerm "low" "z-mf" "5 10" )
  (?riskbc addTerm "average" "pi-mf" "5 10 15 20")
  (?riskbc addTerm "high" "s-mf" "15 30"))
```

Now the fuzzy graph may be created to show the dependency between the linguistic variables: ?age and ?riskbc, and fuzzy rules may be created and added to the fuzzy graph. A fuzzy rule consists of a list of generalized constraints that makes its premises and its conclusion. The parameter "?x" denotes an anonymous object which may be instantiated by the object of a matching generalized constraint in the fact base. The fuzzy graph is added as a shadow fact [12] to the working memory.

```
;; Creating the fuzzygraph
(bind ?fg (new FUZZY_GRAPH)

;;Defining the fuzzy rule: "if age(x) is young then
;;riskbc(x) is low"
(bind ?gc1 (new GENERALIZED_CONSTRAINT ?age "?x"
  "young")
  (bind ?gc2 (new GENERALIZED_CONSTRAINT ?riskbc
    "?x" "low"))
  (bind ?rule1 (new FUZZY_RULE ?gc1 ?gc2))

;;Defining the fuzzy rule: "if age(x) is middle-age
;;then riskbc(x) is average"
(bind ?gc3 (new GENERALIZED_CONSTRAINT ?age "?x"
  "middle-age")
  (bind ?gc4 (new GENERALIZED_CONSTRAINT ?riskbc
    "?x" "average"))
  (bind ?rule2 (new FUZZY_RULE ?gc3 ?gc4))

;;Defining the fuzzy rule: "if age(x) is old then
;;riskbc(x) is high"
(bind ?gc5 (new GENERALIZED_CONSTRAINT ?age "?x"
```

```

"old"))
(bind ?gc6 (new GENERALIZED_CONSTRAINT ?riskbc
  "?x" "high"))
(bind ?rule3 (new FUZZY_RULE ?gc3 ?gc4))
(?fg addRule ?rule3)

;;Adding the rules to the fuzzy graph
(?fg addRule ?rule1)
(?fg addRule ?rule2)

;; adding the fuzzy graph to the working memory
(add ?fg)

```

The statement with fuzzy quantifiers may be represented as a binary quantified generalized constraint. In the following parameter “?x” is a variable bounded by the quantifier.

```

;; creating the generalized constraints that appear
;; in the quantified statements
(bind ?weight (new ATOMIC_LINGUISTIC_VARIABLE
  "weight" "Kilogram" 40 250))
(?weight addTerm "obese" "s-mf" "85 100")
(bind ?gc1 (new GENERALIZED_CONSTRAINT ?weight "?x"
  "obese" ) )
(bind ?gc2 (new GENERALIZED_CONSTRAINT ?riskbc "?x"
  "high"))
(bind ?recom (new ATOMIC_LINGUISTIC_VARIABLE
  "mammogram-recommendation" "percentage" 0 1))
(?recom addTerm "strongly-recommended" "s-mf"
  ".85 1")
(bind ?gc3 (new GENERALIZED_CONSTRAINT ?recom "?x"
  "strongly-recommended" ))

;; Creating the quantified generalized constraints
(bind ?qgc1 (new QUANTIFIED_GENERALIZED_CONSTRAINT
  "many" "?x" ?gc1 ?gc2))
(bind ?qgc2 (new QUANTIFIED_GENERALIZED_CONSTRAINT
  "most" "?x" ?gc2 ?gc3))
;; Adding the quantified generalized constraints to
;; the working memory
(add ?qgc1)
(add ?qgc2)

```

Given that the age of mother is equal to the age of her son plus the age that she gave birth to him, we have two pieces of information regarding Mary’s age which we can be encoded as the following arithmetic generalized constraints.

```

;; Defining the linguistic terms that appear in the
;; arithmetic generalized constraints
(?age addTerm "in-20's" "Interval-mf" "20 30" )
(?age addTerm "about 15" "tri-mf" "13 15 17" )
(?age addTerm "few-years" "trap-mf" " 2 3 5 7")

;; Defining the arithmetic generalized constraints
;; and adding them to the working memory
(bind ?agc1 (new ARITHMETIC_GENERALIZED_CONSTRAINT
  ?age "Mary" "in-20's + about 15" ))
(bind ?agc2 (new ARITHMETIC_GENERALIZED_CONSTRAINT
  ?age "Mary" "?age(Ann) - few-years"))

;; Adding the arithmetic generalized constraints to
;; the working memory
(add ?agc1)
(add ?agc2)

```

And finally, the information regarding Ann’s age can be defined as a simple generalized constraint:

```

(?age addTerm "mid-50" "tri-mf" "50 55 60")
(bind ?gc8 (new GENERALIZED_CONSTRAINT ?age "Ann"

```

```

  "mid-50"))
(add ?gc8)

```

Once the above Jess program is run, the pair of simple and arithmetic facts: ?gc8 and ?agc2 fire the extension principle rule. The extension principle rule replaces “?age(Ann)” with “mid-50” in the arithmetic expression “?age(Ann) – few years” to achieve a fuzzy value for Mary’s Age. The extension principle rule is also fired for the arithmetic fact ?agc1 to calculate the value “in-20’s + about-15”. At this point the fact base would consist of two generalized constraints regarding Mary’s age with different fuzzy values which leads to firing the conjunction rule to combine the constraints to obtain one fuzzy value for Mary’s age.

Now the resulting fact regarding Mary’s age together with the fuzzy graph fact ?fg activate the fuzzygraph interpolation rule, which in turn calculates the value of ?riskbc (Mary) and adds it as a simple generalized constraint to the fact base.

The fuzzy graph interpolation rule is also fired by the pair of facts: ?fg and ?gc8 to calculate the value of ?riskbc(Ann).

Furthermore, the pair of quantified facts: ?qgc1 and ?qgc2 activate the fuzzy syllogism rule which calculates the proportion of obese women for whom the Mammogram is highly recommended. The result of this calculation is asserted to the fact base as a quantified generalized constraint.

In order to show the output of a jess program to the user, the CWJess library includes a jess function: “print-GC”. This function allows users to query the fact base to find the value of a given linguistic variable. For example in the above program if the user wishes to know Mary’s age, or her risk of developing a breast cancer, she can add the following lines to the jess program:

```

;; Querying the fact-base
(print-GC ?age "Mary")
(print-GC ?riskbc "Mary")

```

The output is the two normalized fuzzy sets in figures 7 and 8, deduced by using the extension principle, the conjunction, and the fuzzy graph interpolation rules.

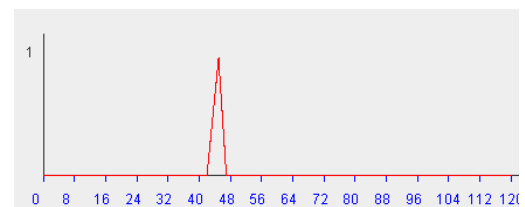


Figure 7. The value of the linguistic variable “age(Mary)”

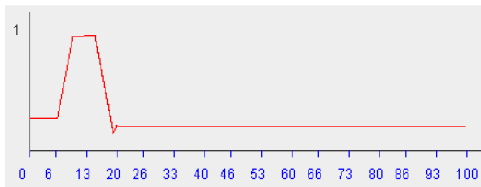


Figure 8. The value of the linguistic variable: "riskbc(Mary)"

## V. SUMMARY AND FUTURE WORK

The paper reports a preliminary work on the implementation of an expert system shell on top of Jess to perform CW reasoning. The resulting shell, CWJess, is a powerful tool which allows users to express fuzzy facts in form of generalized constraints of various types. It provides java classes which enable users to define atomic or composite linguistic variables, linguistic terms, fuzzy quantified statements, fuzzy rules, fuzzy arithmetic expressions, and fuzzy graphs. The CW inference rules, implemented in CWJess, along with Jess Rete algorithm render an inference engine which is able to perform forward reasoning over a set of generalized constraints.

We are working on making the CWJess tool available for download via [www.purvag.com/cwjess](http://www.purvag.com/cwjess). We will provide a complete set of instructions to help users to express their knowledge in terms of GCL. Nevertheless, to facilitate easier translation of linguistic knowledge, we are planning to develop a human readable intermediate language to hide the complexity of underlying GCL.

Several other developments to CWjess can be proposed to make it more expressive and to enhance its reasoning power, among which are:

- Developing necessary classes and methods to allow users to assert generalized constraints with probabilistic

modality, such as: "(age(Mary) is young) isp likely" and perform reasoning on such generalized constraints.

- Implementing the compositional rule of inference for fuzzy relations with a continuous membership function.

## REFERENCES

- [1] L. A. Zadeh, "Fuzzy logic = computing with words," *Fuzzy Systems, IEEE Transactions on*, vol. 4, no. 2, 1996, pp. 103-111.
- [2] L. A. Zadeh, "Computing With Words and Perceptions-A Paradigm-Shift," *Proc IEEE IRI-2009*.
- [3] *FuzzyCLIPS Version 6.04a User's Guide*, Nat. Res. Council, Ottawa, ON, Canada, 1998
- [4] R. Orchard, "Fuzzy reasoning in jess : The fuzzyj toolkit and fuzzyjess," *Proc. in ICEIS 2001, 3 rd International Conference on Enterprise Information Systems*, 2001, pp. 533-542.
- [5] W. Siler, D. Tucker, and J. Buckley, "A parallel rule firing fuzzy production system with resolution of memory conflicts by weak fuzzy monotonicity, applied to the classification of multiple objects characterized by multiple uncertain features," *International Journal of Man-Machine Studies*, vol. 26, no. 3, 1987, pp. 321-332; DOI 10.1016/s0020-7373(87)80066-4.
- [6] J. Baldwin, T. Martin, and B. Pilsforth, "Foil-Fuzzy and Evidential Reasoning in Artificial Intelligence", Research Studies Press, Taunton, Somerset, England, 1995.
- [7] L. A. Zadeh, "Toward a generalized theory of uncertainty (GTU): an outline," *Inf. Sci. Inf. Comput. Sci.*, vol. 172, no. 1-2, 2005, pp. 1-40.
- [8] D. Dubois and H. M. Prade, "Possibility theory : an approach to computerized processing of uncertainty", Plenum Press, 1988, p. xvi, 263 p.
- [9] L. A. Zadeh, "Commonsense reasoning based on fuzzy logic," *Book Commonsense reasoning based on fuzzy logic*, Series Commonsense reasoning based on fuzzy logic, ed., Editor ed.^eds., ACM, 1986, pp. 445-447.
- [10] P. Hájek, *Metamathematics of Fuzzy Logic*, Dordrecht: Kluwer, 1998
- [11] W. M. Dong and F. S. Wong, "Fuzzy weighted averages and implementation of the extension principle," *Fuzzy Sets and Systems*, vol. 21, no. 2, 1987, pp. 183-199.
- [12] E. F. Hill, *Jess in Action: Java Rule-Based Systems*, Manning Publications Co., 2003, p. 480.





## Visual Exploration of Cash Flow Chains

Jerzy Korczak, Walter Łuszczak  
Uniwersytet Ekonomiczny  
ul. Komandorska 118/120  
53-345 Wrocław, Poland  
Email: jerzy.korczak@ue.wroc.pl

□

**Abstract**—The paper proposes a new method for interactive visual exploration of the chains of financial transactions, assisting an analyst in the detection of money laundering operations. The method mainly concerns searching, displaying and annotating selected groups of transactions from a database. We show how one can programmatically and interactively reduce the volume of the chains surveyed and limit the analysis to the most suspicious transactions. In order to improve readability of the transaction graph, an evolution-based algorithm has been designed to optimize its visual representation. The system is verified on the real-life database of financial transactions. The experiments conducted have shown that allowing visual exploration, one can accelerate the search process and enrich the data analysis.

### I. INTRODUCTION

**M**ONEY laundering denotes the financial operations aimed at putting into circulation money from illegal sources, mainly from criminal activities. Detection of the transactions involved in the process of money laundering is extremely difficult and complex. This follows on the one side from the large amount of data that must be examined and, on the other from the difficulty of distinguishing the ordinary transactions from suspicious ones. Many systems of recording and monitoring transactions, type AML (Anti-Money Laundering), use predetermined rules to detect suspicious transactions [1]; [6]; [10]; [11]. However, despite the commitment of huge resources, the effectiveness of current solutions is very low. In general, the percentage of verified transactions as indicated by the system of suspicious transactions, measured by the index of TPR (True Positive Rate), is very small. For example, one of the largest financial institutions in the Balkans, after the purchase and implementation of software - some of the newest and most expensive software against money laundering - scored TPR equal to 0.02%. This result may seem daunting, but, as practice shows, the majority of institutions start with this level of accuracy and, after years of fine-tuning the system

and finding new patterns, TPR may reach about 5% [4] Government bodies involved in the issuance of regulations recommend to developers that the system should reach TPR ranged from 4% to 7% [4]; [6].

In this paper, we propose an interactive method for visual data mining, assisting an analyst in the detection of illegal transactions. Support consists mainly of visualization and annotation of some of the operations in a graph showing the money flow chains between accounts.

In real-world applications, the number of accounts and transactions is huge; therefore even a partial graph visualization poses problems of feasibility, readability and interpretation of financial operations. In order to assist the analyst to interactive interrogation of the database, we have introduced the functionality of defining SQL queries and graphical operations on the graph of transactions resulting in a significant reduction in the search space. We have also provided a number of editing and graph visualization features, including heuristics to visualize sequences of transactions, graphical aggregation of transactions corresponding to the same account, and the transaction graph optimization by evolutionary algorithm.

The fundamental characteristic which determines the ease of interpretation and pattern matching operations is the representation of the transactions. Typically in financial information systems, transactions are represented in a tabular form as shown in fig. 1, a report of program cash flow chains between accounts taken from the SART system (System for Analysis and Registration of Transactions, developed by TETA SA company) [12]. We see that the presentation of a large number of transactions is barely legible. The report contains information on only a few dozen transactions and yet these are difficult to display on a computer screen. An additional difficulty in visualizing the chain of transactions using the tables and data sheets is heterogeneity. For example, an account can have several incoming and dozens or even hundreds of outbound transactions. In such cases the transactions recorded in the traditional table are hardly legible and are cumbersome to handle because the analyst must often "scroll" the report.

The paper is organized as follows. The next section will show how one can interactively reduce the volume of the

This work has been performed in the framework of a research project on methods of mining anti-money laundering data, conducted by the Center for Intelligent Information Technology in Management of Wrocław University of Economics, Poland.

analyzed chains, namely how one can view in place of all the data only the most important transactions.

TEX_ID	TEX_ID	DT_SDATE	TEX_BNOS	TEX_IDNS	TEX_BNOD	TEX_IDND
2 743	2 738	2007-12-28	13 200 019	22 921	20 601 054	13 359
9 323	5 444	2008-01-04	20 600 002	13 359	12 401 789	26 830
16 763	6 959	2008-01-04	12 401 789	26 830	20 601 054	13 359
45 033	16 704	2008-01-08	20 600 002	13 359	10 101 010	26 010
55 792	18 214	2008-01-08	10 101 010	26 010	20 601 054	14 037
56 288	18 215	2008-01-08	10 101 010	26 010	20 601 054	14 037
10 654	5 455	2008-01-04	20 600 002	13 359	12 401 789	26 833
16 886	6 961	2008-01-04	12 401 789	26 833	20 601 054	13 359
45 156	16 704	2008-01-08	20 600 002	13 359	10 101 010	26 010
55 915	18 214	2008-01-08	10 101 010	26 010	20 601 054	14 037
56 411	18 215	2008-01-08	10 101 010	26 010	20 601 054	14 037
44 847	16 704	2008-01-08	20 600 002	13 359	10 101 010	26 010
2 745	2 739	2007-12-28	13 200 019	22 921	20 601 054	13 359
2 747	2 740	2007-12-28	13 200 019	22 921	20 601 054	13 359

Fig. 1. Tabular representation of chains of transactions

Then we show a simple, intuitive way in which one can find more detailed information. We will present a solution that optimizes the visual representation of transaction chains enabling easier analysis of graphic schemes, accelerating the search process, and enriching the functionality of the system. For example, by marking the selected transactions as suspicious, one can try to learn the system to recognize patterns of money laundering operations.

More details about the database structures in AML systems may be found in [7], [8], [9].

## II. REPRESENTATION OF TRANSACTION CHAINS

The main concern while designing a visualization algorithm of sequential operations is the complexity of the resulting graph. In the project, the main measure of assessment of the graph readability was the number of intersections of edges connecting vertices of the graph: the fewer the edges, the better the resulting graph. If the graph contains  $n$  accounts there could occur approximately  $n^2/2$  edges between them [13]. Let us note that in practice some of the edges can be repeated. It follows that the complexity of such a graph is  $O(n^2)$  [2], [3]. Fig.2 presents an example of a graphical representation of the transaction chains. The bold edges indicate that there are many transactions between linked accounts. Red color means that the edge, or at least one of multiple edges, represents a suspicious transaction.

To effectively minimize the number of cuts generated in the initial graph, the whole transformation process was divided into two stages. In the first the simple heuristics have been applied such as the aggregation of multiple edges, the clustering of accounts by type, the linking accounts with the same neighbor in the graph, and the displacement at the bottom of the graph the accounts that are related to transactions with one another account. The graph thus prepared leads to the second stage of the optimization process, which aims to reduce the number of edge intersections and thus improves the readability of the graph.

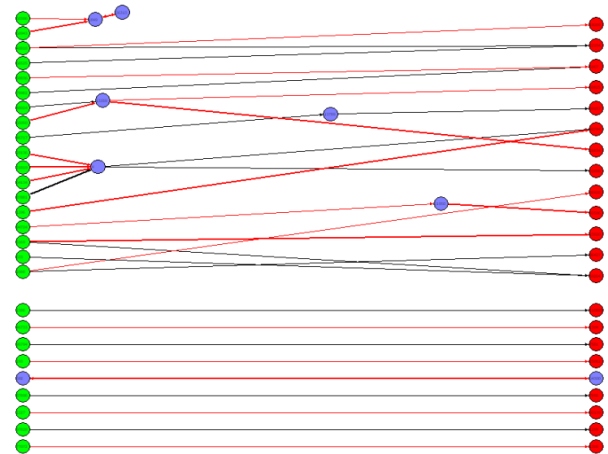


Fig. 2 Graph of transaction chains obtained by application

Let us now proceed to describe the evolutionary algorithm that minimizes the number of intersections of edges in the graph. The task of the algorithm is to find such repartition of vertices on the plane that the number of intersections of edges of the graph is as small as possible (see the specification of Algorithm 1).

*Algorithm 1. Minimization of the number of edge intersections*

```

C = 5000; // number of iterations without changes to
stop evolution
MutSrcPr = 0.45; // probability of source account mutation
MutInterPr = 0.1; // probability of intermediate account mutation
MutDstPr = 0.45; // probability of target account mutation
Fmin = infinity; // minimal number of intersections
F = infinity; // minimal number of intersections of individual in
one trial
FTemp = infinity; // number of intersections in currently mutated
individuals
IBest = NULL; // the best individual
ITemp = NULL; // temporary individual, similar for all trials
I = NULL; // evolved individual in the trial
Imut = NULL; // mutated individual in a given iteration
N = 3; // number of tested individuals
M = 3; // number of mutations of a given individual
FOR i = 1 TO N DO // number of tested individuals

BEGIN
ITemp = initiated new graph of transactions();
FOR j = 1 TO M DO // number of mutation of an individual
I = ITemp;
DO
Imut = I;
IF RAND() < MutSrcPr THEN Perform Imut mutation of source
accounts;
IF RAND() < MutInterPr THEN Perform Imut mutation of
interm. accounts;
IF RAND() < MutDstPr THEN Perform Imut mutation of target
accounts;
FTemp = compute a number of intersections in Imut;
IF FTemp > F THEN
BEGIN
F = FTemp;
I = Imut;
END
WHILE (F doesn't change in C last iterations)
IF F < Fmin THEN

```



```
BEGIN
  Fmin = F;
  lBest = l;
END
END
END
```

To explain the algorithm let us start by describing the process of evolution. The outer loop FOR algorithm indicates how many individuals will be considered in the initial population. Since the algorithm does not use crossover operator, the evolution of population is equivalent to the evolution of each of the individuals separately. The inner loop FOR controls the number of mutation attempts on the same individual. The algorithm may fall into a local minimum of the objective function, therefore repeating the evolution several times from the same starting point greatly reduces these cases.

The main part of the evolutionary algorithm is the most nested DO-WHILE loop. Mutations are performed until the number of iterations reaches *C* or the solution is not improving. We have proposed three types of mutation: displacement of source, intermediate and target nodes. Displacement of the source nodes changes the headings of two randomly selected groups of vertices (there may be a group containing one vertex). Here we use the heuristics that when the order of accounts in the group does not change, it is therefore permissible to move only the entire group. A similar mutation relates to the target accounts. Mutation of intermediate nodes is drawing the new position in the permitted area between the source and target accounts.

The second important feature concerns the user interaction related to the definition of the scope of data investigation. This is done in two ways. The first solution is the ability to filter transactions that are selected from a database by SQL query. The second way is a direct manipulation on the displayed graph of chains of transactions. For large databases, sequential viewing and analysis of all transactions would be very time-consuming or even impossible. To partially overcome these difficulties, the two phase commit was implemented, namely a display of the transactions characteristics that meet the search criteria; and acceptance by the analyst to draw the graph.

In the example in fig.3 the analyst launched a simple query searching for the transactions with the amount of more than 190 thousand zł. From the computing, *Characteristics*, we see that there are 809 such transactions involving 401 source, intermediate, and target accounts, and the area calculated for visualization is 4004 x 4004 pixels. This form lets the analyst select whether to view the graph, or modify the query to reduce or extend the set of transactions. The graph is displayed after optimization using an evolutionary algorithm

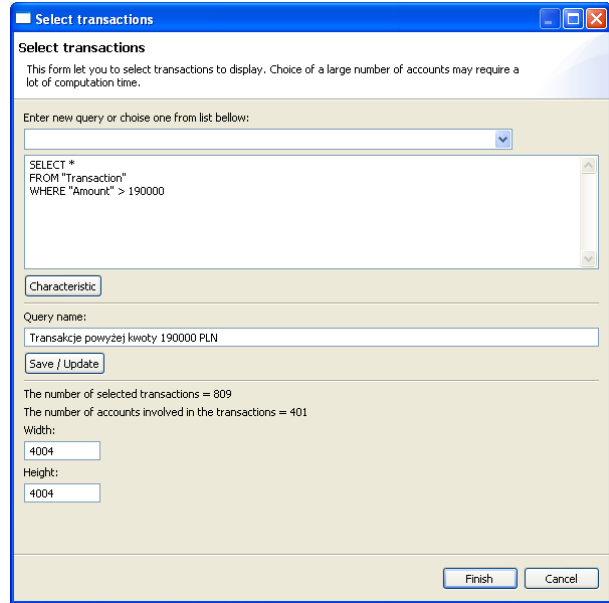


Fig. 3 Two-phase commit - the appearance of the window after query execution

In some cases the analyst may optionally re-arrange the graph manually. An example of the moving operation of the account ID 5169 is shown in fig.4.

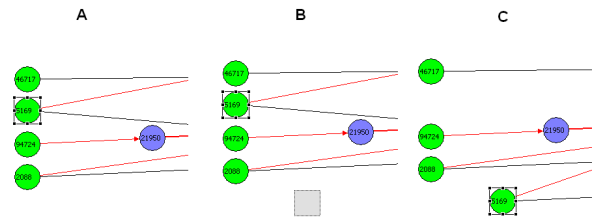


Fig.4. Moving a node: A - selection, B - dragging, C - a new position

Another operation is to remove vertices of the graph. This may happen when, after SQL filtering, the graph contains either too many accounts or accounts which have proved during the analysis to be uninteresting. To obtain a more readable graph, the analyst may then remove them manually. Let us note that the accounts are not removed physically from the database, but only from the currently displayed diagram. The steps of account deletion are shown in fig. 5.

An important operation is the interactive marking of transactions as suspicious. With this functionality, an analyst or pattern recognition program may mark the transaction as suspicious [cf. TABLE I]. Information about the suspicious transaction is stored in a database and can be used in subsequent studies.

In the system many other useful features have been designed, such as memorizing a query filtering transactions, zooming graph, *Undo* and *Redo* functions, and saving a trace

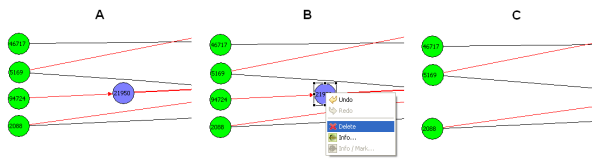


Fig.5. Remove a node: A - the initial situation, B – selection from the menu, *Delete*, C - after removal of the top figure

operations log. A detailed description of these functions along with examples is given in [9].

### III. DESCRIPTION OF EXPERIMENTS

This section contains descriptions of two experiments, namely: 1) advanced retrieval of database transactions, and 2) optimization of the graph using an evolutionary algorithm. In the experiments, the real-life data obtained from one of the Polish banks are stored in a relational database PostgreSQL 8.3.

The experimental database consists of three tables: *Transaction*, *Query* and *Trace*. The first one contains the input data in the form of transaction descriptions (TABLE I). *TransId* column is a unique key serving to identify unambiguously the transaction. The next two columns contain information about the accounts involved in the transaction: *SrcAcctId* is the identifier of the account from which money is transferred, and *DstAccl* indicates a debited account. In this example, identifiers are represented by small integers, but in real applications, the account identifier may be the IBAN or NRB. The *Date* and *Amount* columns contain the date of execution and the transfer amount, expressed in PLN. The *Suspected* column indicates whether the transaction is suspicious in terms of money laundering, and may be modified by the system.

TABLE I. EXAMPLE OF TRANSACTION DESCRIPTIONS

TransId	SrcAccl	DstAccl	Date	Amount	Suspected
677	21561	22924	01-02	73 095,67	
678	21561	22924	01-02	88 142,15	
2888	24756	16579	01-02	67 684,03	
730	558	22971	01-02	66 495,43	
1821	19202	23934	01-02	57 669,16	
1823	18677	22839	01-02	80 000,00	
1933	2135	24037	01-02	180 000,00	
1925	21561	24032	01-02	74 657,08	
1926	21561	24033	01-02	76 910,61	
1927	21561	24033	01-02	109 671,77	
1928	21561	24032	01-02	134 746,56	

Other tables are *Query* and *Trace*. The first of these tables holds query requests through which data about the transactions are loaded. The second, the *Trace* table, stores the user actions on the displayed graph of chains of

transactions. Data for application can be obtained from any bank's financial system which offers the functionality of transaction registration and thus at least partially satisfies the requirements of the legislature. If the data have a different structure than that shown in TABLE I, any tool of the class ETL (Extract, Transform, Load) can be used for importing data.

The system runs on PC computers, currently on the two leading operating systems, Windows and UNIX, using the popular database management system Postgres. To meet the requirement to run on different systems, the system was implemented using Java environment and a library Eclipse RCP GEF for graphics editing. Eclipse RCP (called the Eclipse Rich Client Platform) is a library which gives rise to the creation of rich graphical interfaces. To implement the system a so-called fat client architecture (called thick client) has been used. The program is executed directly on the user's computer called the client, and the data is stored on the server side.

#### *Experiment 1. Advanced database retrieval – queries*

In this experiment, we assume the role of the banking analyst who analyzes the context of transactions to detect money laundering operations.

Suppose that an analyst is interested in transactions having the most frequent accounts, that is, accounts which participate in the largest number of transactions during a specified period. For example, assume that the concerns are related to transactions in January 2011, involving a top 10 the most frequent accounts in the database. This task can be written in the form of the following SQL query:

```

SELECT *
FROM "Transaction"
WHERE ("Date" BETWEEN '2011-01-01' AND '2011-01-31')
AND ("SrcAccl" IN
(
SELECT "AccountId"
FROM
(
SELECT "SrcAccl" AS "AccountId" FROM "Transaction"
UNION ALL
SELECT "DstAccl" AS "AccountId" FROM "Transaction"
)
GROUP BY "AccountId"
ORDER BY COUNT(*) DESC
LIMIT 10
)
OR "DstAccl" IN
(
SELECT "Accl"
FROM
(
SELECT "SrcAccl" AS "AccountId" FROM "Transaction"
UNION ALL
SELECT "DstAccl" AS "AccountId" FROM "Transaction"
)
GROUP BY "AccountId"
ORDER BY COUNT(*) DESC
LIMIT 10
)
)

```

Following the query definition, the system shows the characteristics of the resulting graph (Fig. 6), where the number of accounts in the graph is 140 and the number of

transactions 306. The analyst can visualize the transactions chains because the volume seems feasible to display. This experiment also illustrates an advantage over traditional reports. Data having 140 accounts and 306 transactions could be displayed on one page. However, this visualization does not show all relevant data, such as account numbers, amounts and dates of transactions; but the analyst can easily identify the main streams of cash flow. If necessary, by using the graph zooming, or by clicking on particular objects, he can very quickly get more information. The resulting graph is the subject of further research.

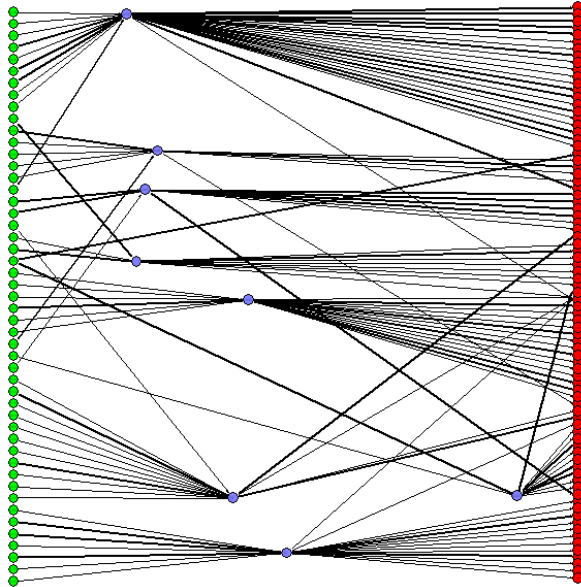


Fig.6. Graph showing the selected transactions by the query

This experiment shows how the analyst can apply his previous experience in searching money laundering transactions and the possibilities of visual exploration of cash flow chains. The program interface is in fact very simple, but just to define a SQL query may require some skills and experience in using this language. The solution to this question could be collaboration of the analyst with an SQL programmer who may predefine the specified queries or templates allowing the formulation of queries by QBE. However, the great advantage is the richness of language features that allow us to define any transaction filtering operation.

*Experiment 2. Genetic optimization of graph of transactions*

In this experiment, we are going to assess the quality of an evolutionary algorithm used to plot the graph [5]. The goal is to answer the question whether the parameters - the population size and number of mutations ( $N$  and  $M$ ) - significantly affect the result obtained. In other words, we are interested whether with limited computing power it is more profitable to run a lot more shorter evolutions or fewer but longer ones.

Computing of the number of intersections in the graph has the complexity  $O(m^2)$ , where  $m$  is the number of edges

in the graph [3], [13]. Thus, the computational complexity of the algorithm is  $O(m^2 * N * M * C')$ , where the parameter  $C'$  is dependent on the parameter  $C$ . From the definition of the parameter  $C$ , the estimation of  $C'$  is very difficult, since it is not known how many times one can carry out the innermost loop of the algorithm.

The algorithm has been tested on the previous database with 140 accounts and 306 transactions. In the experiment, the parameter  $C = 100$ , mutation probability of the source and destination nodes are 50%, and 90% of intermediate vertices. TABLE II shows the number of intersections obtained in each of the nine trials.

TABLE II. RESULTS OF GRAPH OPTIMIZATION

Number of individuals	Number of mutation trials	Number of intersections
1	1	277
	2	426
	3	258
2	1	229
	2	304
	3	373
3	1	506
	2	353
	3	334

The best result is the 229 intersections, and this is less than half the size of the worst. The standard deviation of the results is 87.0. The experimental data showed that the number of iterations ( $C$ ) and population size ( $N$ ) affected the value of the evaluation function. Fig. 7 shows the decreasing value of the objective function for individual evolution. After conducting many experiments we can conclude that it is better to run several times short mutations of many individuals rather than focus on a few individuals and increase the number of iterations.

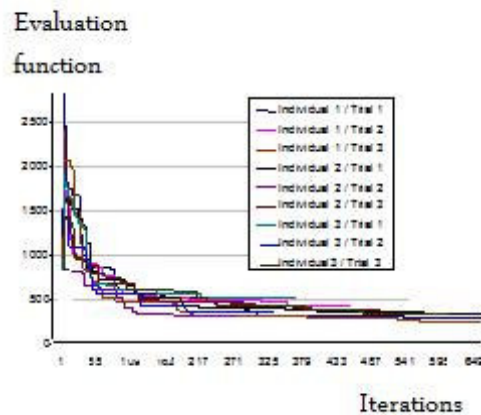


Fig.7. Curves of evaluation functions

#### IV. CONCLUSIONS

This study presents the visual exploration of cash flow chains, which has not previously been addressed in existing systems. The greatest difficulty in designing and implementing the system was to find a way to reduce the size of the transaction chains graph. This process has been divided into two stages. First, the heuristics have been applied to deploy accounts by type, then grouping accounts with the same neighbor in the graph, and placing at the bottom of the graph accounts that are related with only one other account. In order to reduce the visual complexity of the multiple edges, the edges were aggregated into a single edge (displayed in bold).

The graph visualization was optimized by an evolutionary algorithm whose task was to find an organization of vertices and edges with minimal number of intersections. Experiments have shown that the number of intersections in the graph has been reduced significantly, which improved its readability.

It is noteworthy that the system offers other interesting features such as logging user operations performed on the graph. The register of user actions can be used to work further on machine learning that would try to imitate the work of the analyst.

The proposed solution can be easily adapted into existing systems of transactions analysis and contribute to the development of new methods for exploration of complex data structures.

#### REFERENCES

- [1] Actimize, <http://www.actimize.com>, date of access: 2010-08-23.
- [2] V. Bryant, "Aspects of combinatorics. A Wide-ranging Introduction", Cambridge Univ. Press, 1993. ]
- [3] T.H. Cormen, C.E. Leiserson, R. Rivest, C. Stein, "Introduction to Algorithms", MIT Press, 2009.
- [4] D. S. Demetis, Artificial non-intelligence and anti-money laundering, London School of Economics, London, 2009.
- [5] D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley, 1989.
- [6] E. Iwonin, City Handlowy, Implementation of AML directive into Polish legal system - City Handlowy experience, in Advanced Information Technologies for Management AITM 2009 Wrocław University of Economics, Research Papers no. 85, Wrocław 2009.
- [7] J. Korczak, W. Marchelski, B. Oleszkiewicz, A New Technogical Approach to Money Laundering Discovery using Analytical SQL server, in Advanced Information Technologies for Management – AITM 2008, J. Korczak, H. Dudycz and M. Dyczkowski (eds.) Research Papers no 35, Wrocław University of Economics, 2008, pp. 80-104.
- [8] J. Korczak, B. Oleszkiewicz, Modelling of Data Warehouse Dimensions for AML Systems, in Advanced Information Technologies for Management – AITM 2009, J. Korczak, H. Dudycz and M. Dyczkowski (eds.) Research Papers no 85, Wrocław University of Economics, 2009, pp. 146-159.
- [9] W. Łuszczuk, Wizualna eksploracja łańcuchów transakcji bankowych, Praca magisterska (MSc thesis), Uniwersytet Ekonomiczny, Wrocław, 2010.
- [10] Norkom, <http://www.norkom.com>, date of access: 2010-08-23
- [11] Ocean Technology, <http://www.oceantechnology.com.vn/front/?page=solution&sid=27>, date of access: 2010-08-23
- [12] TETA S.A., <http://www.teta.com.pl/71578.php>, date of access: 2010-08-23.
- [13] R.J. Wilson, "Introduction to Graph Theory", Addison-Wesley, 1996.

# Logical Inference in Query Answering Systems Based on Domain Ontologies

Juliusz L. Kulikowski

Nalecz Institute of Biocybernetics  
and Biomedical Engineering PAS,  
4, Ks. Trojdena Str,  
02-109 Warsaw, Poland

Email:

juliusz.kulikowski@ibib.waw.pl

**Abstract**—This paper describes a proposal of using ontological models as a basis of Query Answering Systems design. It is assumed that the models are presented in the form of relations described on some classes of items (ontological concepts) specified by taxonomic trees. There are analyzed the sufficient and necessary conditions for getting the replies to the queries as solution of relational equations based on the data provided by ontological databases. Simple examples illustrate basic concepts of practical realization of the Query Answering Systems based on domain ontologies.

**Index Terms**—query answering systems, domain ontologies, relations, relational equations,

## I. INTRODUCTION

DOMAIN ontology is a tool of formal representation of knowledge concerning an application area that in a decision problem should be taken into consideration. The application area knowledge can in general in many different forms be represented. Despite the traditional, descriptive form, rather unsuitable to computer implementation purposes, various types of knowledge can be presented by propositional logic terms [1] graphs (including their specific forms like trees [2], Petri nets [3], PERT networks [4], Bayesian networks [5], etc.), Horn clauses [6], semantic networks [7], Minsky frames [8], etc., less or more useful in a given decision problem. Generally speaking, a domain knowledge consists of verified data concerning the domain, its general nature, components, structure, properties of the components and satisfied by them relations. The above-mentioned formal tools to presentation of various aspects of the domain knowledge can be used. Otherwise speaking, they make possible construction of formal models of selected aspects of the application domain under consideration. The model should specify the concepts used to the application domain characterization and the existing among them physical, logical, organizational, etc. relationships. This leads to a domain ontology described by a quadruple [9]:

$$O = [C, R, A, Top] \quad (1)$$

where:  $C$  – a non-empty set of concepts,  
 $R$  – a set of relations among the concepts,  
 $A$  – a set of axioms,  
 $Top$  – a highest-level concept in  $C$ .

Among the relations a taxonomy of concepts headed by  $Top$  is mandatory, other relations are established according to the application problem needs.

This is illustrated by the following example.

Example 1.

A problem consists in calculation of a voltage drop on an electric resistor. This is well known that the solution is given by the Ohm law:

$$U = I \cdot r \quad (2)$$

where  $U$  denotes the voltage drop [V],  $I$  – current intensity [A],  $r$  – resistor resistance [ $\Omega$ ]. However, the problem can also be described by a domain ontology formalism. The domain ontology is then given by (1) where:

$C = \{\text{electric current flow, physical entities, measurable parameters, resistor, electric current, electric tension, electric resistance } r, \text{ current intensity } I, \text{ voltage drop } U\}$ ;

$R = \{\text{a taxonomy } T \text{ of concepts (see below), relationships between the concepts, formula (2)}\}$ ;

$A = \{\text{axioms concerning real functions}\}$ ;

$Top = \text{electric current flow}$ .

The taxonomy  $T$  of concepts can be presented in the form of a tree (Fig.1.):

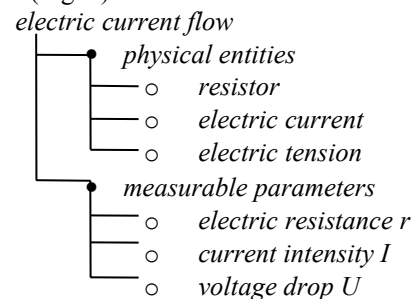


Fig. 1. A taxonomic tree of concepts constituting a domain ontology *electric current flow*.

The domain ontology should also contain a relation not following directly from the taxonomic tree, assigning *measurable parameters* to the *physical entities*. The relation  $D$  (described by) is given by a set of ordered pairs:

$$D = \{[\text{resistor, electric resistance } r], [\text{electric current, current intensity } I], [\text{electric tension, voltage drop } U]\}.$$

So-presented domain ontology contains full information necessary to solve the above-formulated task. At a first glance it may see that the ontological description of the given application domain is too much complicated; this is because in practice, in a simplified reality description most of the ontology components by default are assumed. Nevertheless, solutions of a large class of simple physical, engineering, economical, management etc. tasks consciously or unconsciously on the corresponding domain ontologies are based. On the other hand, domain ontologies presented by (1) in many other cases are insufficient. For example, description of an enterprise (a domain) should consist of components concerning its organizational structure, personal staff, administrative procedures, services, etc. The relationships between the staff members and the administrative procedures, the services and the organizational sections etc. play a substantial role in the enterprise functioning description, however, they do not follow directly from the domain ontology presented in the above-given form. This problem has also been remarked in [9] where an attempt to construct a taxonomy for the *blood* concept led to the necessity of *blood* characterization as a *fluid* and as a biological *tissue*. Each aspect of a composite real object like an enterprise, blood circulatory system, large educational project, etc. constitutes a specific sub-domain which by its proper ontology should be described. A formal description of such object should thus rather by several sub-domain ontologies than by a single, general domain ontology be presented. It also seems clear that a multi-aspect description of a composed object would be difficult when a single, unified taxonomic tree of concepts is used. E.g., in a domain representing a certain city, an entity *City Hall* of the corresponding taxonomic tree, as having several aspects, should be included into the sub-trees *Architectural monuments*, *Municipal offices*, *City communication sites*, etc. However, this destroys the structure of a single taxonomy tree where each concept by only one node should be represented. On the other hand, if the original concept *City Hall* is split into three separate concepts, say: *City Hall<sup>1</sup>*, *City Hall<sup>2</sup>*, *City Hall<sup>3</sup>* then it arises a problem, how in the taxonomic tree some pairs, triples, etc. of nodes should be to the same physical object assigned. Entering into consideration a concept of *multi-model domain ontologies* removes the above-mentioned difficulty. In such case, the domain ontology should also contain a set  $Q$  of higher-level relations (*super-relations*) among the relations constituting the ontological models. The concept of *ontological models* defined as sub-domain ontologies used to formal description of various aspects of a higher-level application domain has been introduced in [11]. The more general, multi-aspect domain ontology definition takes thus the form:

$$O = \{OM_1, OM_2, \dots, OM_i; Q; A\} \quad (3)$$

where:

$OM_i$  – ontological models of the sub- domains;

$Q$  – a family of *super-relations* over the models;

$A$  – an extended algebra of relations and super-relations.

This concept is convergent with a represented by other Authors [12], [13] general tendency to base domain ontologies on strong mathematical backgrounds. In the below-presented case mathematical set theory and extended relations algebra is used as a basic tool for domain ontologies formal description. Domain ontology given in the form (3) provides a basic information that makes possible reasoning about objects belonging to the given domain and relations among them not directly specified by the ontological models.

However, also this concept has some drawbacks caused by the dependencies between the component ontological models leading to a problem of effective logical reasoning based on the statements drawn from several different ontological models. The problem has been partially analyzed in a context of using domain ontologies to computer-aided image understanding [14]. In the present paper a problem of logical inference in computer query answering systems (QAS) based on domain ontologies is more largely presented. In particular, the concepts of *relational equations* and of *reversed relations* using to logical inference improvement is below presented. Basic concepts of the extended algebra of relations and super-relations have been presented in the papers [14]-[16] and here only roughly are described. The paper is organized as follows: In Sec. II selected basic concepts used in the paper are shortly presented. Sec. III presents the idea of the relationships among the ontological models description by super-relations. Sec IV contains a proposal of using reversed relations to improve the effectiveness of reasoning based on the statements drawn from the domain ontology. Conclusions are given in Sec. V.

## II. BASIC CONCEPTS

### A. Ontologies

Generally speaking, ontology is a philosophical concept while domain ontology in computer science is a notion related to a formal description of a selected, less or more complicated, part of reality. Domain ontology is also not a mathematical notion, but it in mathematical terms can be described. However, a certain duality of terms used in domain ontologies occurs and needs to be explained. In particular, concepts in ontological sense are equivalent to classes or to sets in a mathematical sense. A taxonomy of concepts  $T$  in ontology is represented by a rooted tree in the graph theory sense while the highest-level concept  $Top$  is the root of the tree. The leafs (lowest-level nodes) of a tree are in ontological sense interpreted as instances of the closest higher-level ontological concepts, usually representing some basic, individual objects of the described reality. Relations in the mathematical sense (see below) may denote not only simple set theory relations but also any algebraic or functional, discrete or continuous, deterministic or fuzzy relations. Entities described by ontological models are objects if considered in the application domain context while from ontological models point of view are instances satisfying the corresponding relations. The family  $Q$  of super-relations



tions (see below) in (3) in ontological sense can be interpreted as a domain knowledge structure.

### B. Relations and super-relations

For description of a relation  $r$  between the elements of a finite family of sets  $S_1, S_2, \dots, S_K$  they should be linearly ordered in a logical sense and then the family can be denoted by  $[S_k]$ ,  $k = 1, 2, \dots, K$  or, shortly, by  $[S_k]_1^K$ . However, the logical order is not obviously identical to the physical one assuming that the a permutations transforming the logical order into the physical one (and vice versa) are defined. For the sake of simplicity, below, any logical order of a linearly ordered set will be assumed to be identical to the one of its presentation in the text. This is substantial for correct understanding of the operations of the extended algebra of relations.

A Cartesian product of a family of sets  $[S_k]_1^K$  will be denoted by  $C$ . If  $|S_k|$  denotes a cardinal number of elements of  $S_k$  then  $|C| = |S_1| \cdot |S_2| \cdot \dots \cdot |S_K|$ . The relation is called *finite* if  $|C|$  is finite. Any subset

$$\rho \subseteq C \quad (4)$$

is called a *relation* described on  $[S_k]_1^K$ . The number  $K$  is called a *length* of the relation while the maximum among the cardinal numbers  $|S_1|, |S_2|, \dots, |S_K|$  is called an *extent* of the relation. The relation is called an *empty relation* if it is an empty set; in such case it is denoted by  $\theta$ . The relation  $\rho$  is called a *trivial relation* if  $\rho \equiv C$ . For a given relation  $\rho$  any subset  $\rho', \rho' \subseteq \rho$ , is called a *sub-relation* of  $\rho$  while  $\rho$  is called an *over-relation* of  $\rho'$ . For a given relation  $\rho$  described on  $[S_k]_1^K$  any  $K$ -tuple  $s = [s_1, s_2, \dots, s_K]$  belonging to  $\rho$  is called an *instance* of the relation. For a given sub-family of  $n$  sets  $[S_p, S_q, \dots, S_t] \subseteq [S_k]_1^K$  (preserving its original logical order), and their Cartesian product denoted by  $C^*$ , the corresponding elements of the instances  $s$  of  $\rho$  determine linearly ordered  $n$ -tuples ( $n \leq K$ ) belonging to  $C^*$ . The set  $\rho^*$  constitutes a relation called a *partial relation* of  $\rho$  created by its projection on  $[S_p, S_q, \dots, S_t]$ ; in such case it is said that  $\rho$  is an *extension* of  $\rho^*$ .

In computer realization, due to the necessity of numerical data quantization, all relations are approximated by the finite ones. In a finite relation of finite length a coefficient:

$$d = \frac{|\rho|}{|C|} \quad (5)$$

is called a *density* of the relation.

Due to the fact that relations are described as subsets of Cartesian products the general set algebraic notions to the relations can be applied. In particular, if a countable linearly ordered family of sets  $F$  is taken into consideration and  $2^F$  denotes a class of all its sub-families (including a null

sub-family  $\emptyset$  and  $2^F$  itself) then it can be taken into consideration a family  $\Omega$  of all possible relations described on the linearly ordered sub-families of  $2^F$ . In  $\Omega$  the ordinary set algebra can be interpreted as an *extended algebra of relations* described on the sub-families of  $2^F$  [15]. The so-defined *extended algebra of relations* is a Boolean algebra with  $\theta$  as its null-element and  $\Omega$  as its unity element. The result of an algebraic operation (sum, intersection, difference) performed on any two relations  $\rho', \rho''$  described, respectively, on the families of sets  $F', F''$  is by a definition a relation described on  $F' \cup F''$  (this is not so in a “traditional” algebra of relations defined on the families of all relations described on a fixed family of sets). The extended algebra of relations admits algebraic operations on finite algebraic compositions of other relations.

In a family  $[S_k]_1^K$  of sets on which a relation is described some sets may be defined as relations described on lower-level sub-families of sets. This leads to the concept of *super-relations* as relations whose instances contain variables being instances of some lower-level relations. The structure of a super-relation may be presented by a multi-level bracket-expression, like e.g.:

$$\rho \subseteq S_1 \times [S_2 \times [S_3 \times S_4]] \times S_5 \times S_6$$

representing a three-level super-relation. It should be remarked that the above-given super-relation is not identical to  $\rho^* \subseteq S_1 \times S_2 \times S_3 \times S_4 \times S_5 \times S_6$ , because a Cartesian product of sets is neither a symmetrical nor associative operation. Dates [year, month, day] or addresses [country, city, street, house, flat] are typical examples of compact variables used in super-relations. In algebraic operations performed on super-relations the contents of the pairs of corresponding (the same level) brackets should be considered as a compact (single) variables. Therefore, the extent of the above-presented relation is equal 4 (not 6).

### C. Semantic interpretation

The relations described on the concepts of a domain ontology can usually be semantically interpreted. For example, if a domain ontology *Teaching plan* contains the concepts *Teacher, Subject, Group, Day, Time, Room* then the relation  $\rho'$  described on [Teacher, Subject, Group] can be semantically interpreted as “teacher’s obligations” while  $\rho''$  described on [Group, Subject, Day, Time, Room] as “groups’ time-schedule”, etc. Similarly, if there is given an instance of  $\rho'$ :

$$s = [Smith, physics I, A3]$$

where :  $Smith \in Teacher, physics I \in Subject, A3 \in Group$ , then a semantic interpretation of  $s$  may be:

“Smith is a teacher of physics I in the group A3”.

The semantic interpretation of the relation instance can thus be given by an affirmative statement presented in any of stylistic forms admissible in a given natural language. Any given ontological model  $OM$  specifying its concepts

and based on them relations defines thus a *semantic area*  $\Sigma$  as a set of semantic interpretations that can be assigned to the instances of any formally admissible instances of the relations. However, the semantic area may in general contain not only relations' instances corresponding to some real situations but also the ones that only theoretically are admissible, that are suggested, supposed, suspected, etc. In this sense, the semantic area does not describe the reality but rather a space of conceptual models in which a description of the actual reality is possible.

Let  $s = [s_1, s_2, \dots, s_k]$  be an instance of the relation  $\rho$  described on  $[S_k]_1^K$ . We can replace a selected value  $s_k$  by  $x$  as a symbol of unknown variable and put:

$$[s_1, s_2, \dots, s_{k-1}, x, s_{k+1}, \dots, s_k] \in \rho \quad (6)$$

This expression is *true* only for certain  $x \in S_k$ . Therefore, (6) is a *relational equation* and any  $x$  making it *true* is its solution. The values  $s_1, s_2, \dots, s_{k-1}, s_{k+1}, \dots, s_k$  can be considered as fixed *equation parameters*. The relational equation (6) can semantically be interpreted as a question: "What is (are) an  $x$  such that for given  $s_1, s_2, \dots, s_{k-1}, s_{k+1}, \dots, s_k$  the relation  $\rho$  is satisfied?". Similar questions can be formulated with respect to any other variable or subsets of variables.

Example 2

On the basis of the (above-mentioned) relation:

$$\rho' \subseteq \text{Teacher} \times \text{Subject} \times \text{Group}$$

and its instance  $s = [\text{Smith}, \text{physics I}, \text{A3}]$  the following relational equations can be formulated:

- i.  $[x, \text{physics I}, \text{A3}] \in \rho'$ ,
- ii.  $[\text{Smith}, y, \text{A3}] \in \rho'$ ,
- iii.  $[\text{Smith}, \text{physics I}, z] \in \rho'$ .

In the equations two parameters are fixed (e.g. *physics I* and *A3* in eq. i). Similarly, some multi-variable equations, like e.g.:

$$\text{iv. } [x, \text{physics I}, z] \in \rho',$$

(containing a single parameter *physics I*) can be formulated. The following queries correspond to the above-given relational equations:

- i. "Who is the teacher of physics I in the group A3?";
- ii. "What is the subject Mr Smith teaches the group A3?";
- iii. "What is the group Mr Smith teaches physics I?";
- iv. "What group and by whom is taught physics I?";

(the queries, if necessary, can be stylistically reformulated). Therefore, a given relation  $\rho$  generates a set of *schemes of*

*queries* that potentially on the basis of domain knowledge contained in the relation can be replied. A relational equation may represent a given query if the following conditions are satisfied:

- a) Each item which in the query is asked for in the relational equation is represented by a variable;
- b) each parameter of the query occurs and takes the same value as a parameter of the relational equation;
- c) semantic interpretation of the relational equation is in accordance with this suggested by the query.

The conditions *a)*, *b)* can be formally proven. However, the condition *c)* is not easy to be proven if the semantic accordance is not by default assumed and no relations' entries suggest their semantic meaning. Moreover, some queries can not be replied on the basis of a given relation if it does not admit formulation of a relational equation satisfying the above-given conditions. E.g., a query:

"What day Mr Smith teaches the group A3 physics I?"

can not be replied on the basis of  $\rho'$  which does not contain a variable *Day* being an item of the query. However, such queries can be replied if the ontological model contains additional relations satisfying the necessary conditions. For example, suppose that from the above-mentioned relation  $\rho'$  the following instances can be drawn:

$$\begin{aligned} s' &= [\text{A3 Mathematics II}, \text{Tuesday}, 11^{00}-11^{45}, 120], \\ s'' &= [\text{A3, Mathematics II}, \text{Wednesday}, 12^{00}-12^{45}, 122], \\ s''' &= [\text{A3, Physics I}, \text{Friday}, 11^{00}-11^{45}, 122]. \end{aligned}$$

Then, we can take into consideration an extended intersection of the relations  $\tau = \rho' \sqcap \rho''$  and on the basis of the above-given instances the following instances will be obtained:

$$t = [\text{Smith}, \text{physics I}, \text{A3}, \text{Friday}, 11^{00}-11^{45}, 122].$$

One can try to express the above-given query in the form of a set of relational equations:

$$[\text{Smith}, \text{physics I}, \text{A3}, x, *, *] \in \tau$$

where  $*$  denotes any value of the parameters *Time* and *Room* which in the question have not been specified. Therefore, the relation  $\tau$  can be reduced by taking into consideration a partial relation  $\tau^*$  defined as a projection of  $\tau$  on  $[\text{Teacher}, \text{Subject}, \text{Group}, \text{Day}]$ . This leads to the relational equation

$$[\text{Smith}, \text{physics I}, \text{A3}, x] \in \tau^*$$

whose solution is *Friday*.



III. LOGICAL INFERENCE BASED ON A DOMAIN ONTOLOGY

It has been shown that a domain ontology provides domain knowledge presented in the form of concepts and based on them relations and also it determines the set of admissible structures of queries about the given application domain that can be logically answered. In the case of a single ontological model the set of admissible questions is the larger the larger is the number of relations and are their lengths.

The *sufficient* conditions for answering by a QAS a query concerning the given application domain are:

- 1) Existence of a corresponding domain ontology;
- 2) Expression of the query in the terms semantically equivalent to some concepts of the given ontology;
- 3) Presentation of the query by a semantically equivalent form of a relational equation based on a non-empty relation belonging to an ontological model and satisfying the conditions *a)*, *b)*, *c)* (formulated in Sec. II).

Let us remark that if  $\rho$  is a relation in the given ontological model then any its sub-relation as well as any partial relation can also be considered as a relation of this ontological model. Hence, the sufficient condition 3) means that a relational equation can be based on a relation directly belonging to the ontological model or on any its sub-relation or partial relation. And still, the situations in which the above-described sufficient conditions are fully satisfied are rather exceptional. In a more general case, a relation that directly can be used to presentation of a given query in the form of a relational equation can not be found. In such case it is necessary to look for the suitable *algebraic combinations* of the relations satisfying the below-formulated *necessary* conditions:

- a) the algebraic combination of relations contains as its entries the variables and parameters of the query;
- b) the algebraic combination of relations is not empty by definition;
- c) the algebraic combination of relations admits semantic interpretation being in accordance with this suggested by the query.

Like in Sec. II, the conditions a) , b) can be formally proven by the below-described procedure while c) is assumed to be satisfied in the below-considered cases.

Let  $\rho^1, \rho^2, \dots, \rho^M$  be all relations that in a given ontology have been described. Each relation  $\rho^m$ ,  $m = 1, 2, \dots, M$ , is described on a sub-family of sets (ontological concepts)  $S^m$ .

The sum

$$S = S^1 \cup S^2 \cup \dots \cup S^M \tag{7}$$

contains all concepts on which the relations are based.

A general structure of the domain ontology then can be described by a *hyper-graph* [17]  $H$  whose nodes are given by  $S$  and the subfamilies  $S^m$ ,  $m = 1, 2, \dots, M$ , represent its hy-

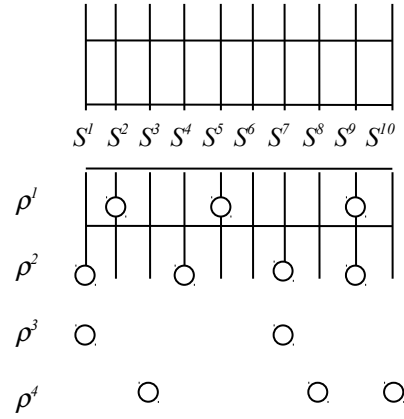


Fig. 2. A hyper-graph consisting of 10 nodes and 4 hyper-edges.

per-edges. Two hyper-edges are *adjacent* if their intersection  $S^u \cap S^v$  is non-empty.

Example of a hyper-graph is shown in Fig. 2. It consists of 10 nodes ( $S^1, \dots, S^{10}$ ) and 4 hyper-edges ( $\rho^1, \dots, \rho^4$ ) denoted by circles in the corresponding rows. In the context of ontology's structure interpretation, each hyper-edge represents a relation described on the sets (ontological concepts) represented by the nodes. So, for example,  $\rho^1$  is a relation described on the sets  $S^2, S^5$  and  $S^9$ . The pairs of hyper-edges  $(\rho^1, \rho^2)$  and  $(\rho^2, \rho^3)$  are adjacent while  $(\rho^1, \rho^3)$  is not. Similarly,  $\rho^4$  is not adjacent to any other hyper-edge. A subset of hyper-edges such that any pair of them can be connected by a sequence of pair-wise adjacent hyper-edges of the subset is called a *weak component* of the hyper-graph. The hyper-graph in Fig. 2 consists thus of two weak components:  $(\rho^1, \rho^2, \rho^3)$  and  $(\rho^4)$ .

Let us denote by  $Q$  a family of concepts that are used as unknown data or fixed parameters of a query ordered to the QAS. For proving the necessary conditions a)-b) for replying the query on the basis of a domain ontology whose structure is described by the hyper-graph  $H$  it is necessary:

1. to select in  $H$  the minimum sub-family of nodes such that  $Q \subseteq \{S^i\}^*$ ;
2. to find in  $H$  all minimal subsets  $R^h$  of hyper-edges such that the sum  $I^h$  of nodes belonging to  $R^h$  satisfies the condition:

$$Q \subseteq I^h \subseteq \{S^i\}^*; \tag{8}$$

3. taking into account that each hyper-edge in  $R^h$  represents a relation of the domain ontology described by  $H$ , construct an extended intersection of all represented by  $R^h$  relations;
4. if  $Q \subset I^h$  then replace the intersections by partial relations defined as projections of the intersections on  $Q$ , denote such relations by  $r^\alpha$ ;

5. from any relation  $r^\alpha$  select its instances whose components corresponding to the query parameters take values equal to those of the query, collect all such instances as a sub-relation of  $r^\alpha$ ,  $\chi^\alpha \subseteq r^\alpha$ ;

6. each  $\chi^\alpha$  being not an empty relation can be used as a basis for a relational equation representing the given query;

7. solution of the relational equation based on  $\chi^\alpha$  is given as its projection on the set (node of  $H$ ) belonging to  $Q$  and denoting the unknown data of the query.

The above-described procedure may provide more than one solution of the problem presented by the query. This may happen not only because  $\chi^\alpha$  may contain several instances satisfying the condition 7 but also because in the step 2 more than one subset  $R^h$  can be found. In such case elimination of formal solutions not satisfying the necessary semantic condition c) is possible and recommended.

### Example 3

Let us assume that a query has a form:

“What is the value of  $S^2$  assuming that the values of  $S^4$  and  $S^7$  are given?”

Let the corresponding domain ontology is represented by the hyper-graph  $H$  shown in Fig. 2. For replying the question the set  $Q = \{S^2, S^4, S^7\}$  will be taken into consideration. No hyper-edge of  $H$  directly covers  $Q$ . However, it is covered by the sub-set of hyper-edges  $R = \{\rho^1, \rho^2\}$  for which we obtain

$$\Gamma = \{S^1, S^2, S^4, S^5, S^7, S^9\} \supset Q.$$

It will be thus taken into consideration an extended intersection of the relations:

$$r = \rho^1 \sqcap \rho^2$$

described on the family of sets  $\Gamma$ . However, this relation is redundant and for replying the query it is sufficient to take into account its projection  $\chi$  on the sub-family  $Q$ . Then the steps 6 and 7 should be performed by taking into account the instances of  $\chi$ .

In the above-described procedure of query answering the following basic steps can be discriminated:

- i. reformulation of the query for its formal processing;
- ii. proving formal possibility to reply the query on the basis of the given domain ontology (steps 1, 2);
- iii. finding formal solutions of the query problem (steps 3 – 7).

Therefore, a block-scheme of a QAS based on a domain ontology can be presented as shown in Fig. 3.

## IV. CONCLUSIONS

A practical role of QAS in numerous application areas is permanently increasing. However, it arises the problem of application domain knowledge representation in a form suitable to computer processing. The proposal of using for this purpose ontological models based on the extended algebra of relations seems to be one of several possible ones. It seems to be relatively easy to be realized by computer system assuming that the ontological models for the application domains being of interest are designed and available. However, this approach needs practical solution of several technical problems among which proving semantic accordance between the queries expressed in natural language with the ontological models and effective searching of rela-

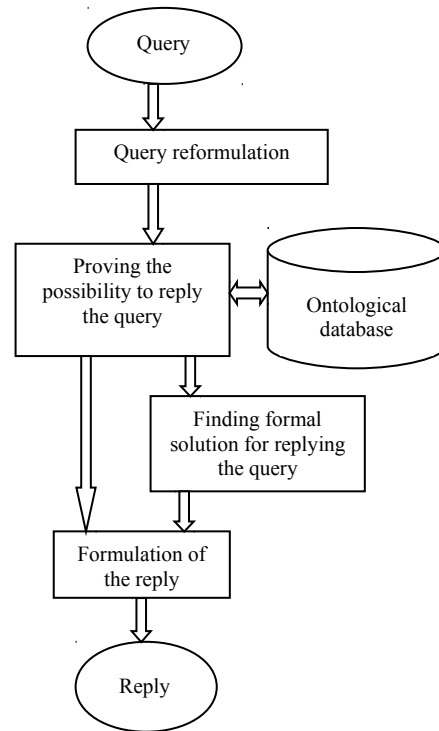


Fig. 3. Block-scheme of a Query-Answering System based on domain ontology.

tions' instances in ontological databases should be mentioned as the most important ones.

## REFERENCES

- [1] S. Russel, P. Norvig. *Artificial Intelligence. A Modern Approach*. Prentice Hall, 2003.
- [2] R. J. Wilson. "Introduction to Graph Theory". Longman Group Ltd., London, 1979.
- [3] J. L. Peterson. "Petri Net Theory and the Modeling of Systems". Prentice-Hall, Inc., Englewood Cliffs, 1981.
- [4] D. T. Phillips, A. Garcia-Diaz. "Fundamentals of Network Analysis". Prentice-Hall, Inc., Englewood Cliffs, 1981.
- [5] J. Pearl. "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference". Morgan Kaufmann, San Mateo, 1988.
- [6] R. Kowalski. "Logic for Problem Solving". North-Holland, New York, 1979.
- [7] R. C. Schank. "Conceptual Information Processing". North Holland Publishing Co., Amsterdam, 1975.
- [8] M. A. Minsky. "Frameworks for Representing Knowledge in the Psychology of Computer Vision". Mc Graw Hill, New York, 1975.
- [9] M. Fernandez-Lopez, A. Gomez-Perez, "Overview and Analysis of Methodologies of Building Ontologies". *The Knowledge Eng. Rev.*, vol. 17, no 2, 2002, pp. 129-156.
- [10] O. Bodenreier, A. Burgun. "Biomedical Ontologies". In: H. Chen, S. S. Fuller, C. Friedman, W. Hersh (Eds.), *Medical Informatics. Knowledge Management and Data Mining in Biomedicine*. ISIS 8, Springer, USA, 2005, Chapt. 8, pp. 211-236.
- [11] J. L. Kulikowski. "The Role of Ontological Models in Pattern Recognition". In: M. Kurzynski *et al.* (Eds.), *Computer Recognition Systems. Proc. of the 4<sup>th</sup> Int. Conf. on Computer Recognition Systems CORES'05*. AiSC, Springer, Berlin, 2005, pp. 43-52.
- [12] A. Abdoullayev. "Reality, Universal Ontology, and Knowledge Systems. Toward the Intelligent World". IGI Publishing, Hershey, 2008.
- [13] A. Zilli, E. Damiani, P. Ceravolo, A. Corallo, G. Elia (Eds.). "Semantic Knowledge Management. An Ontology-Based Framework". IGI Global, Information Science Reference, Hershey, 2009.
- [14] J. L. Kulikowski, "Ontological Models as Tools for Image Content Understanding". In: *Computer Vision and Graphics. Int. Conf.*

- ICCVG 2010, Warszawa, Proc. Part 1*, LNCS 6374, Springer, Berlin, 2010, pp.43-58.
- [15] J. L. Kulikowski. "Relational approach to structural analysis of images". *Machine Graphics and Vision*, vol. 1, nos ½, 1992, pp. 299-309.
- [16] J. L. Kulikowski "Description of irregular composite objects by hyper-relations". In: K. Wojciechowski, B. Smolka, H. Palus, R. S. Kozera, W. Skarbek, L. Noakes (Eds.), *Computer Vision and Graphics. Int. Conference ICCVG 2004, Warsaw. CIV 32*, Springer, Dordrecht, 2006, pp. 141-146.
- [17] C. Berge. "Graphs and Hypergraphs". North-Holland, Amsterdam, 1973.



# Competitive and self-contained gene set analysis methods applied for class prediction

Henryk Maciejewski

Institute of Computer Engineering, Control and Robotics,  
Wrocław University of Technology,  
ul. Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland  
Email: Henryk.Maciejewski@pwr.wroc.pl

**Abstract**—This paper compares two methodologically different approaches to gene set analysis applied for selection of features for sample classification based on microarray studies. We analyze competitive and self-contained methods in terms of predictive performance of features generated from most differentially expressed gene sets (pathways) identified with these approaches. We also observe stability of features returned. We use the features to train several classifiers (e.g., SVM, random forest, nearest shrunken centroids, etc.) We generally observe smaller classification errors and better stability of features produced by the self-contained algorithm. This comparative study is based on the leukemia data set published in [3].

## I. INTRODUCTION

**B**UILDING diagnostic or prognostic classifiers based on profiles of gene expression from microarray or similar massive throughput experiments seems one of the most challenging tasks in bioinformatics. The problem of class prediction can be regarded as ill-formulated as the number of samples (e.g., patients) in a typical microarray study does not exceed a few hundred with the number of features (gene expression values) recorded for a sample usually exceeding 20 thousand. Many different approaches to class prediction based on massive throughput data were proposed (e.g., [7], [10], [13], [14]). One of the most challenging problems is related to feature selection based on high dimensional data. Standard approaches start with identification of sets of differentially expressed genes to be used as features for class prediction. These methods focus on features with individual strong predictive power, however they treat the features as unrelated and they do not take into account potential (biological) relationships among features. This explains why most feature selection methods produce very unstable features, i.e. small changes in training data result in different feature sets, [25], [15], [16]. This further explains why classifiers built from microarray studies are very sensitive to the selection of parameters (such as the number of features, etc.), and generally demonstrate unstable estimates of prediction error.

In our previous work [17], we proposed an enhanced procedure of feature selection based on domain knowledge about potential relationships among features (genes). Such knowledge of groups of functionally related genes is available in databases e.g., KEGG, Gene Ontology or Biocarta, and is

now being actively developed. The method proposed in [17] derives features for class prediction from the most strongly activated pathways. In [17] we compare this approach to the standard method and empirically show that although individually weaker, the new features seem more stable and bring improved performance.

In [17] the global test algorithm, developed in [11], was applied to identify activated pathways to be used as features for class prediction. Recently different approaches to gene set analysis were proposed, e.g., [8], [23], [11], [4], [25]. They can be broadly categorized as *competitive* or *self-contained*, and they fundamentally differ methodologically, [12]. It is not clear whether these two approaches produce similar feature sets in terms of their predictive performance and in terms of stability. The purpose of this work is to investigate this issue. We compare predictive performance of features generated with a selected competitive method (Gene set analysis (GSA) algorithm) and a selected self-contained method (global test). We also analyze whether the feature sets differ in terms of stability.

The organization of the paper is as follows. First, competitive and self-contained methods of gene set analysis are described in detail. Then an algorithm of sample classification based on activation of gene sets is presented. The algorithm is later used to evaluate predictive performance and stability of feature sets returned by the two gene set analysis methods. Finally, results of a comparative study are elaborated based on a real microarray assay.

## II. GENE SET ANALYSIS METHODS

Many different approaches to gene set analysis have been recently proposed. An overview of the most important methods is available in [25], and the statistical issues related to these different methods were analyzed by Goeman and Buehlmann in [12]. The earliest developed and probably simplest methods compare the list of genes in the set of interest with the list of differentially expressed genes. An example of such methods is the over-representation analysis (ORA) proposed in [6], which compares these two lists of genes by means of contingency tables. The chi-square test is used in order to verify the null hypothesis that the differentially expressed genes are not over-represented in the gene set of interest. Rejection of the null hypothesis indicates that the gene set is differentially expressed

This work was sponsored by the grant MNiSzW N516 510239

(or *activated*). An extension of the ORA method is the Gene Set Enrichment Analysis (GSEA), proposed in [20], [23], which does not require that genes are potentially arbitrarily declared as differentially expressed by using a fixed threshold. The method ranks the genes by some measure of differential expression and then tests the null hypothesis that the members of the gene set of interest are uniformly distributed along the ranking list. The null is tested against the alternative that the gene set appears at the top or bottom of the ranking list, ie. can be regarded as activated. GSEA uses a modified Kolmogorov-Smirnov statistic to test the null hypothesis. Another method, based on GSEA is the Get Set Analysis (GSA) algorithm developed in [8]. It uses a maxmean statistic in place of the Kolmogorov-Smirnov test which leads to slightly better power.

In the methodological paper [12], these and similar approaches were named as *competitive* methods. These methods test whether a gene set is differentially expressed (or associated with the target) by comparing expression of genes in the set with expression of genes not in the set. A fundamentally different approach is realized by *self-contained* methods [12], which directly analyze association of genes in the set of interest with the target and do not take the remaining genes into account. Examples of self-contained methods are the Global Test [11], Global Ancova, [18], PLAGE, [24] or SAM-GS proposed in [4].

In this work we use gene set analysis methods to identify pathways which will be used to generate features for classification of samples. We identify the most differentially expressed (or activated) pathways and then use genes in these pathways as features for class prediction. In this study, we use one self-contained approach (global test) and one competitive method (GSA algorithm) and experimentally compare these methods in terms of (a) predictive power of features returned, and (b) stability of features in the presence of small modifications of data. The gene set analysis methods used in the study are now explained in detail.

#### A. Competitive methods – GSA algorithm

The competitive methods compare differential expression of genes in the gene set of interest  $G$  with expression of genes not in  $G$ . More specifically, they aim to verify the null hypothesis:

$H_0$  : The genes in  $G$  are at most as often differentially expressed as the genes not in  $G$ .

The GSEA method and its more powerful version GSA test the specific null hypothesis that the genes in the set  $G$  are uniformly distributed over the list of all genes ranked by some measure of differential expression.

In order to test the hypothesis, the GSA algorithm uses the maxmean statistic [25]:

$$M = \max \left\{ \left| \frac{\sum_{i=1}^m I(t_i > 0) t_i}{m} \right|, \left| \frac{\sum_{i=1}^m I(t_i < 0) t_i}{m} \right| \right\} \quad (1)$$

where  $m$  is the number of genes in  $G$  and  $t_i$  is the measure of differential expression of the  $i$ -th gene in  $G$ .

Significance of the  $M$  statistic is evaluated using the permutation test (permutation involves both genes and class labels). In the algorithm proposed in the next section we will use the permutation based p-values to select the top differentially expressed pathways whose member genes will be used as features for class prediction.

#### B. Self-contained methods – global test

The global test method aims to verify the null hypothesis of no association of the set  $G$  with the target, namely:

$H_0$  : No genes in a set  $G$  are associated with the target (ie. no genes in  $G$  are differentially expressed).

In order to test the hypothesis, global test uses generalized linear models to express the relationship between expression of genes in the set  $G$  and the target, such that

$$g(E(Y_i|\beta)) = \alpha + \sum_{j=1}^m x_{ij}\beta_j \quad (2)$$

where  $g$  is link function in generalized linear models (e.g., the logit function for binary target),  $x_i$  denotes vector of expression of  $m$  genes in the gene set  $G$  for sample  $i$ , with class label  $Y_i$ , and  $\beta_j$  is the coefficient for gene  $j$ .

The assumption that no genes in  $G$  are associated with the target is equivalent to testing the null hypothesis:

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_m = 0 \quad (3)$$

Global test assumes that the coefficients  $\beta_1 \dots \beta_m$  are iid with mean 0 which simplifies the hypothesis and makes the test feasible given small number of samples relative to the number of genes in  $G$ . In the algorithm presented in the next section we will use pathways with the smallest global test p-value as features for class prediction.

### III. ALGORITHM OF CLASS PREDICTION

In [17] we proposed an algorithm for class prediction based on activation of pathways and compared it with the commonly used approach where top most differentially expressed unrelated genes are used as features. Here we will use a slightly modified version of this algorithm, with two different methods of feature selection: GSA-based and global test (GT)-based. The estimates of the expected prediction error (EPE) returned by the algorithm will be used as a measure of quality of features produced by these competing gene set analysis methods.

Using the same notation as in [17], we denote results of a microarray study as a matrix  $X_{n,p}$  with  $p$  features (gene expressions) measured for  $n$  samples, with class designation for a sample  $i$  given in  $Y_i$ ,  $i = 1, 2, \dots, n$  (which represent e.g., tumor and control samples). Also let PWDB be the set of  $d$  subsets (denoted  $PW_i$ ,  $i = 1, 2, \dots, d$ ); these represent a priori domain knowledge of groups of related features (e.g., genes in a signaling pathway or genes with a common GO term, etc). The purpose of the algorithm proposed in [17] is

to (a) build the sample classifier given  $X$ ,  $Y$  and PWDB, and (b) estimate the expected prediction error for new samples. Since the number of samples  $n$  is small relative to the number of features  $p$  (the  $p \gg n$  problem), the  $EPE$  has to be estimated by data reuse techniques. We use *internal* cross validation (CV) where the data are repeatedly split into training and test partitions, with the  $EPE$  calculated as the average misclassification rate over all the test partitions. *Internal* cross validation places the feature selection step within subsequent iterations of cross validation, which is mandatory to obtain a reliable measure of classifier performance, as argued e.g. in [19]. (Note that the commonly used and computationally cheaper *external* cross validation realizes the feature selection step once, prior to the CV loop, and based on the complete training data.)

It should be noted that by using internal cross validation we can simultaneously observe *stability* of features generated under slight modifications of the training data. Namely, with the leave-one-out (LOO) internal cross validation scheme the data used for feature selection differs by one sample in consecutive iterations of CV. Hence by observing how the features change during CV steps we will be able to compare the GSA and GT-based procedures in terms of stability of features generated. This is the main justification of our choice to use the LOO cross validation loop in the algorithm proposed. It should be noted that LOO cross validation (realizing low bias but high variance of the estimate of prediction error) is often used in similar studies, e.g., [9], [21], [22].

It should be noted that poor stability of features produced by standard methods (ie. by selecting most differentially expressed unrelated genes) may account for unstable behavior of classifiers built from microarray data [16].

The class prediction algorithm can be summarized in the following steps.

- 1) Leave out sample  $i$ ,  $i = 1, 2, \dots, n$  for model testing, ie., remove row  $i$  from  $X$  and element  $i$  from vector  $Y$  and denote the remaining matrix and vector as  $X^i$  and  $Y^i$ .
- 2) Using the training data  $(X^i, Y^i)$  calculate the p-value with the GT or GSA for each of the PWs in PWDB. Order the PWs by increasing p-value:  $PW_{(1)}, PW_{(2)}, \dots, PW_{(d)}$ .
- 3) Remove columns from  $X^i$  related to features not present in  $PW_{(1)} \cup \dots \cup PW_{(k)}$ , denote this matrix as  $X_{tr}^i$ .
- 4) Using the training data  $(X_{tr}^i, Y^i)$  fit a predictive model  $f$  and classify the sample  $Y_i$  as  $\hat{Y}_i = f(Y_i)$ .
- 5) In the list of counters  $c(PW_j), j = 1, \dots, d$ , corresponding to the  $d$  elements of PWDB, increment the counters  $c(PW_{(j)}), j = 1, \dots, k$ , which correspond to the PWs selected in the current step.
- 6) Repeat steps 1 through 5 for  $i = 1, 2, \dots, n$ .
- 7) Calculate the expected misclassification rate as  $EPE = \frac{1}{n} \sum_{i=1}^n I(\hat{Y}_i \neq Y_i)$ .

In the following section, we will compare performance of several classifiers based on GSA or GT features in terms of the  $EPE$ . For the purposes of this numerical example, the following classifiers were used:

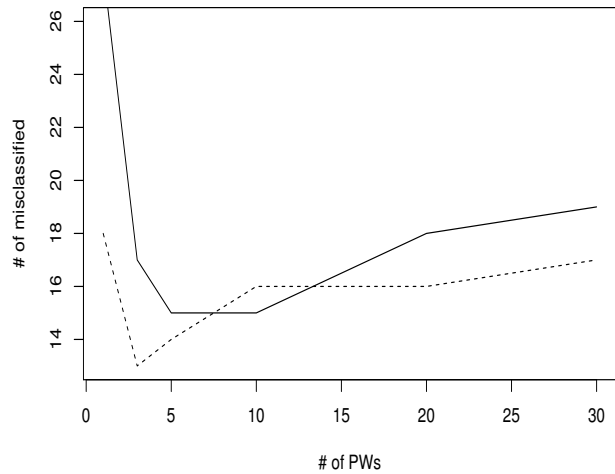


Fig. 1. Number of misclassifications as a function of the number of PWs selected. SVM classifier. Solid line - GSA method, dashed line - global test.

- support vector machine (SVM),
- logistic regression with L2 (Ridge) penalty,
- nearest neighbors,
- nearest shrunken centroid algorithm,
- random forests.

The cost parameter of the SVM classifier as well as the lambda (shrinkage) parameter of the logistic regression were tuned using a simple grid search.

We will also compare the methods in terms of stability of features selected for varying number of gene sets used as features:  $k \in \{1, 3, 5, 10, 20, 30\}$ . Note that the counters in step 5 of the algorithm are maintained to facilitate stability analysis.

#### IV. COMPARATIVE STUDY

The numerical study is based on a subset of the acute leukemia microarray data, published by S. Chiaretti [3]. The dataset includes  $n = 79$  samples with  $p = 12625$  gene expressions; 37 samples represent patients with leukemia and 42 samples represent the control group (the samples are labeled in the original data as ‘BCR/ABL’ and ‘NEG’, respectively). In this example we use the KEGG signaling pathway databases as the collection of gene sets PWDB. The task is to classify patients as leukemia or control based on a profile of most activated pathways.

The overall performance of the different classifiers for varying number of most activated pathways used as features is summarized in Figs. 1 through 5. In the figures the  $EPE$  obtained for the global test and GSA feature selection is compared (note that the  $EPE$  is shown as the the number of misclassified items in 79 iterations of cross validation rather than the ratio).

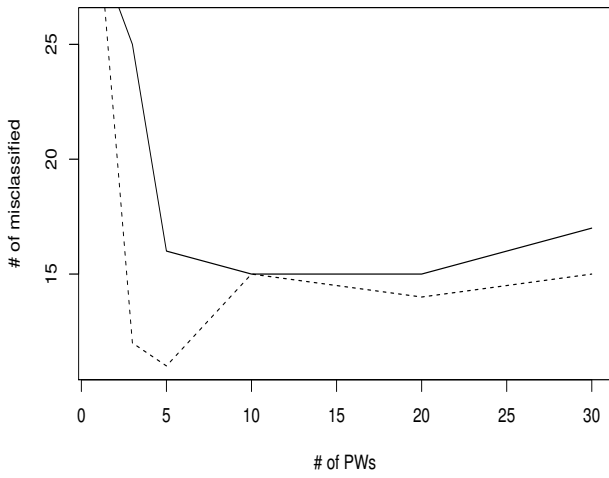


Fig. 2. Number of misclassifications as a function of the number of PWs selected. Logistic regression. Solid line - GSA method, dashed line - global test.

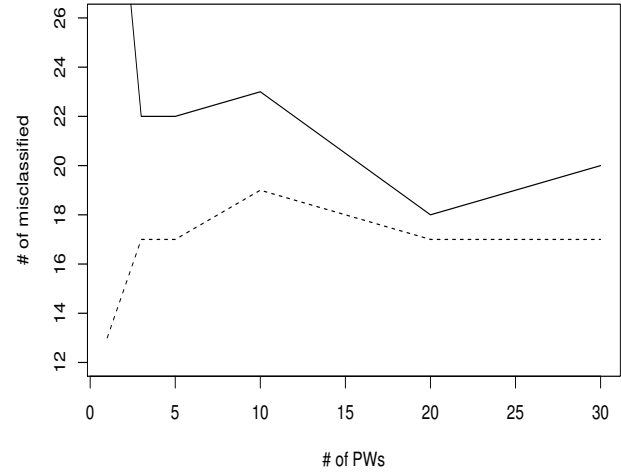


Fig. 4. Number of misclassifications as a function of the number of PWs selected. Nearest shrunken centroid algorithm. Solid line - GSA method, dashed line - global test.

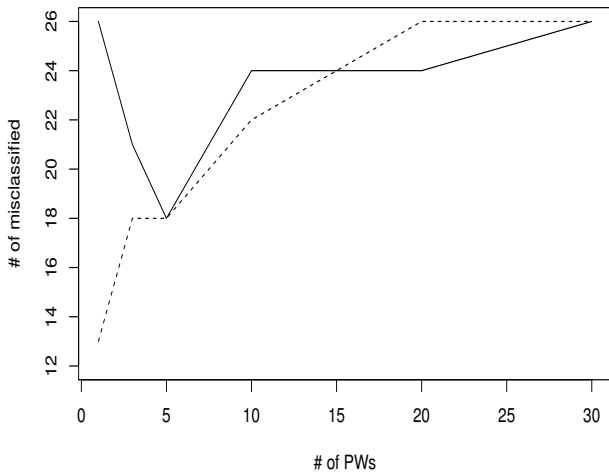


Fig. 3. Number of misclassifications as a function of the number of PWs selected. KNN classifier. Solid line - GSA method, dashed line - global test.

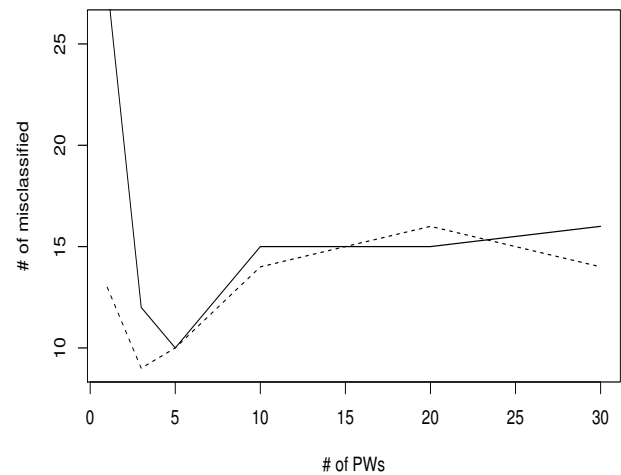


Fig. 5. Number of misclassifications as a function of the number of PWs selected. Random forest algorithm. Solid line - GSA method, dashed line - global test.

We observe that the overall best result was realized by the random forest classifier with  $k = 3$  top PWs selected by the GT algorithm. The winning classifier realized 9 misclassifications, i.e.  $EPE = 11\%$ . This result is better than the best result obtained with the standard feature selection method where the top ranking unrelated genes are selected as features [17]. We also observe that for all the five classifiers the smallest number of misclassifications along  $k \in \{1, 3, 5, 10, 20, 30\}$  is always realized for features selected by the global test (dashed line



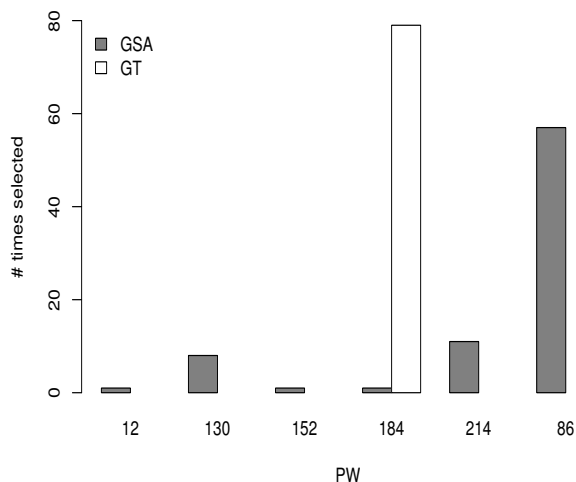


Fig. 6. Features selected in consecutive iterations of CV with corresponding frequency, 1 PW.

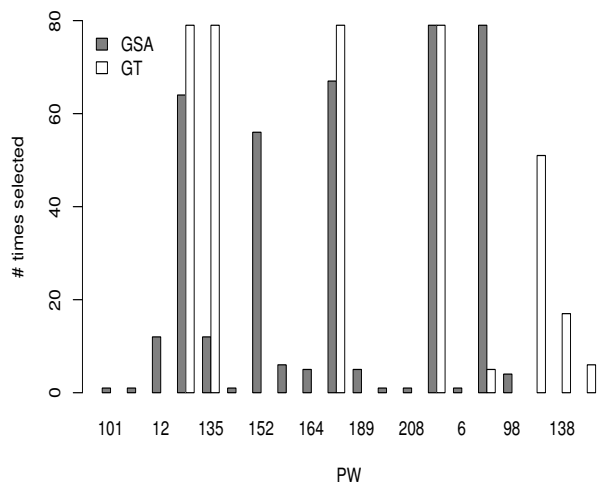


Fig. 8. Features selected in consecutive iterations of CV with corresponding frequency, 5 PWs.

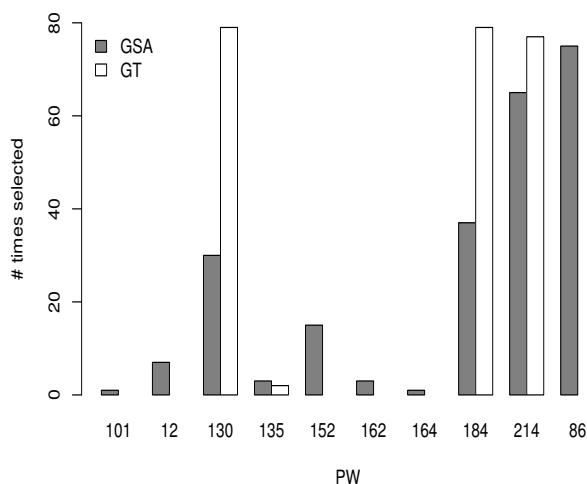


Fig. 7. Features selected in consecutive iterations of CV with corresponding frequency, 3 PWs.

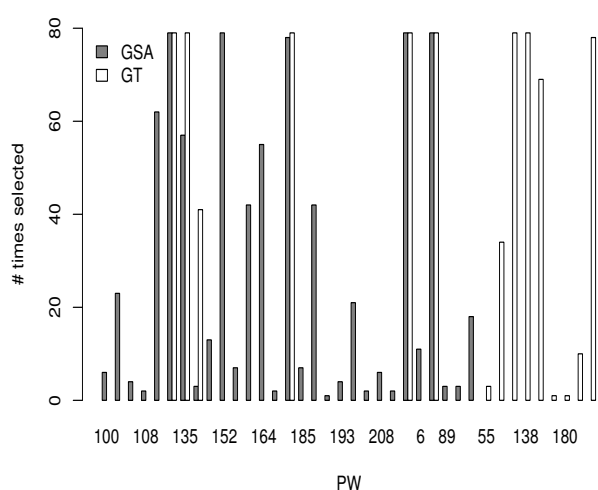


Fig. 9. Features selected in consecutive iterations of CV with corresponding frequency, 10 PWs.

in Figs. 1–5), rather than the GSA method (solid line). The smallest number of misclassifications observed for consecutive classifiers are:

- SVM: 13,
- logistic regression: 11,
- nearest neighbors: 13,
- nearest shrunken centroid algorithm: 13,
- random forests: 9.

It can be observed that with growing number of features (for  $k > 3$ ), performance of models generally deteriorates,

however this effect is strong only for the nearest neighbors classifier (Fig. 3), as the other models internally realize feature selection and therefore are more immune to overfitting.

Another important characteristic of the competing methods is stability of features observed when data changes slightly. Analysis of stability of features is doable using the table of counters  $c(PW_j)$ ,  $j = 1, \dots, d$  maintained in step 5 of the algorithm. The counters record how often (in all 79 iterations of cross validation) a given pathway was selected as a feature. High values of the counters indicate that the feature selection

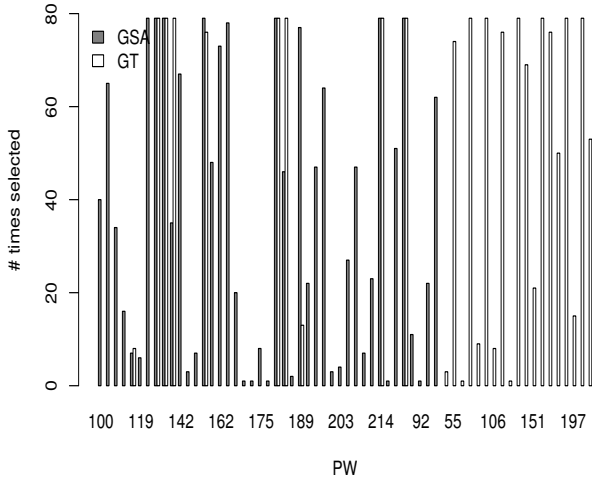


Fig. 10. Features selected in consecutive iterations of CV with corresponding frequency, 20 PWs.

method tends to produce the same features even if data is (slightly) different. On the other hand, if a change of just one sample in the training data leads to many different features, this should be regarded as a drawback of feature selection used.

To analyze stability of features, we first present values of the counters  $c$  for varying value of  $k$  (ie. the number of pathways selected as features) – Figs. 6–10. Fig. 6 shows that the global test kept selecting the same PW (PW with index = 184) over all iterations of CV, while GSA selected 6 different features in 79 iterations, with the feature with index = 86 selected most frequently. (It should be noted that pathways in Figs. 6–10 are denoted by indices to the table of KEGG pathways, as given e.g. in the Bioconductor hgu95av2.db package. For instance, the pathway index=184 corresponds to the KEGG pathway ID 05130.)

For  $k = 3$  (Fig. 7) global test always selects 2 features (184 and 130) and 77 out to 79 times also feature 214. The GSA selects 10 different features (with the winner PW=86 selected 75 times). Similar observations hold for Figs. 8–10.

Although in the method proposed we do not check if  $p$ -values of the selected PWs are significant (as we rely on feature selection capabilities of the classifier used next), we observe in this study that all the top  $k$  PWs selected in step 3 are significant under multiple testing adjustment for  $k$  up to 20 (Holm-adjusted  $p$ -values below 0.05). It should be also noted that since the number of items in PWDB is much smaller than the number of genes (e.g., KEGG database includes ca. 200 pathways, as compared with ca.  $10^4$  genes on a typical microarray), multiple testing correction in the proposed method will be much less conservative than in standard gene selection procedures.

TABLE I  
NUMBER OF DIFFERENT PWs SELECTED AS FEATURES ( $NF$ ) AND MEAN FREQUENCY OF PW SELECTION IN THE CV PROCEDURE AS A FUNCTION OF THE NUMBER OF TOP PWs ( $k$  IN STEP 3 OF THE ALGORITHM)

No of top PWs ( $k$ )	GSA		GT	
	NF	mean freq (%)	NF	mean freq (%)
1	6	16.7	1	100.0
3	10	30.0	4	75.0
5	17	29.4	8	62.5
10	29	34.5	15	66.7
20	43	46.5	29	69.0
30	55	54.5	37	81.1

Analysis of stability can be summarized by using two measures calculated from the tables of counters  $c$  (see step 5 of algorithm):

$$NF = \sum_{i=1}^d I(c(PW_i) > 0) \quad (4)$$

which gives the overall number of different features selected in  $n = 79$  rounds of cross validation, and

$$mean\ freq = \frac{1}{n\ NF} \sum_{i=1}^d c(PW_i) = \frac{k}{NF} \quad (5)$$

which shows mean frequency of selection of features in the set of  $NF$  different features in  $n$  rounds of cross validation. It should be noted that for fixed value of  $k$  the second measure does not provide any more information than  $NF$ , however this measure is convenient to highlight the difference between the methods. Results for  $k \in \{1, 3, 5, 10, 20, 30\}$  are given in Tab. 1, with the mean frequency expressed as percentage. As already observed in Fig. 6, global test always selected one feature ( $NF = 1$ , hence the frequency of its selection is 100%, ie. 79 times in 79 rounds of CV); the GSA selects 6 features, each with mean frequency equal ca. 17% (which translates into 13 times a feature is selected in 79 rounds of CV). For growing  $k = 1, 3, 5$  we observe decreasing mean frequency, which suggests that a growing number of weaker features start getting included, which leads to less consistent feature selection when data changes. It is interesting to notice that for  $k > 5$  the mean frequency again increases, however explanation of this effect requires further investigation.

The final conclusion from stability analysis is clear: the sets of features selected by global test seem more stable as compared to GSA-based features. This characteristic of the global test may account for better predictive performance of these features.

## V. CONCLUSIONS

In this work two methods of gene set analysis were empirically compared in terms of predictive performance of classifiers built using most activated pathways as features. The methods realize different approaches to pathway analysis: global test is a self-contained algorithm and the GSA is a competitive method. The comparative study brings several

interesting observations. First, the self-contained method outperforms the competitive approach in terms of classification error. Second, features selected by the self-contained method appear more stable if data is modified. This is a remarkable characteristic of this feature selection procedure, as due to high dimensionality and small number of samples used for microarray class prediction, standard methods of feature selection demonstrate poor stability. Finally, the methods do select different features (pathways) for prediction (although some overlapping is observed). Based on these results a number of interesting questions and directions for further research can be raised. First, it seems interesting to investigate characteristics of the features returned by the self-contained and competitive methods: whether the activation of pathways is due to weaker effect observed consistently over a large number of member genes, or on the contrary - the method favors stronger local effect in the set of member genes. Next, a hybrid method is worth considering which would combine the strong points of these two approaches. Further research is also necessary into how features can be generated in a more sophisticated way out of the set of most activated pathways. Also, the proposed method requires more comprehensive validation based on further microarray datasets as well as simulated data. Although the purpose of this work was to compare different approaches to gene set analysis in terms of quality of feature selection, another interesting direction for further research involves comparison of these prior biological knowledge based methods with regularized learning techniques such as ridge regression, lasso or elastic net.

#### REFERENCES

- [1] A.J. Adewale, I. Dinu, J.D. Potter, Q. Liu and Y. Yasui, "Pathway analysis of microarray data via regression," *J. Comput. Biol.*, vol. 15, 2008, pp. 269–277.
- [2] D.B. Allison, X. Cui, G.P. Page, M. Sabripour, "Microarray data analysis: from disarray to consolidation and consensus," *Nature Reviews Genetics*, 7, 2006, pp. 55–65.
- [3] S. Chiaretti, et al., "Gene expression profile of adult T-cell acute lymphocytic leukemia identifies distinct subsets of patients with different response to therapy and survival," *Blood*, vol. 103, 2004, pp. 2771–2778.
- [4] I. Dinu, et al., "Improving gene set analysis of microarray data by SAM-GS," *BMC Bioinformatics*, 8, 242, 2007.
- [5] I. Dinu, et al., "Gene-set analysis and reduction," *Briefings in Bioinformatics*, vol. 10, 2008, pp. 24–34.
- [6] S. Draghici, P. Khatri, R.P. Martins, G.C. Ostermeier, S.A. Krawetz, "Global functional profiling of gene expression," *Genomics*, vol. 81, 2003, pp. 98–104.
- [7] S. Dudoit, J. Fridlyand, P. Speed, "Comparison of discriminant methods for classification of tumors using gene expression data," *Journal of American Statistical Association*, vol. 192, 2005, pp. 77–87.
- [8] B. Efron, R. Tibshirani, "On testing the significance of sets of genes," *Ann. Appl. Stat.*, vol. 1, 2007, pp. 107–129.
- [9] K. Fajarewicz, M. Wiench, "Selecting differentially expressed genes for colon tumor classification," *Int. J. Appl. Math. Comput. Sci.*, vol. 13, 2003, pp. 327–335.
- [10] A.M. Glas, "Converting a breast cancer microarray signature into a high-throughput diagnostic test," *BMC Genomics*, 7:278, 2006.
- [11] J.J. Goeman, S.A. van de Geer, F. de Kort, H.C. van Houwelingen, "A global test for groups of genes: testing association with a clinical outcome," *Bioinformatics*, vol. 20, 2004, pp. 93–99.
- [12] J.J. Goeman, P. Buehlmann, "Analyzing gene expression data in terms on gene sets: methodological issues," *Bioinformatics*, vol. 23, 2007, pp. 980–987.
- [13] J. Khan, et al., "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks," *Nature Med.*, vol. 7, 2001, pp. 673–679.
- [14] Y.H. Lin, "Multiple gene expression classifiers from different array platforms predict poor prognosis of colorectal cancer," *Clin. Cancer Res.*, vol. 13, 2007, pp. 498–507.
- [15] H. Maciejewski, "Quality of feature selection based on microarray gene expression data," in Int. Conference on Computational Science 2008, LNCS, vol. 5103, pp. 140–147.
- [16] H. Maciejewski, P. Twaróg, "Model instability in microarray gene expression class prediction studies," in Eurocast 2009, LNCS, vol. 5717, pp. 745–752.
- [17] H. Maciejewski, "Class prediction in microarray studies based on activation of pathways," in Hybrid Artificial Intelligence Systems 2011, LNAI, vol. 6678, pp. 321–328.
- [18] U. Mansmann, R. Meister, "Testing differential gene expression in functional groups: Goeman's global test versus an ANCOVA approach," *Methods of Information in Medicine*, vol. 44, 2005, pp. 449–453.
- [19] F. Markowetz, R. Spang, "Molecular diagnosis. Classification, Model Selection and Performance Evaluation," *Methods Inf. Med.*, vol. 44, 2005, pp. 438–443.
- [20] V.K. Mootha, et al., "PGC-1 alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes," *Nature Genetics*, vol. 34, 2003, pp. 267–273.
- [21] C.H. Ooi, P. Tan, "Genetic algorithms applied to multi-class prediction for the analysis of gene expression data," *Bioinformatics* vol. 19, 2003, pp. 37–44.
- [22] S. Peng, et al., "Molecular classification of cancer types from microarray data using the combination of genetic algorithms and support vector machines," *FEBS Letters* vol. 555, 2003, pp. 358–362.
- [23] A. Subramanian, et al., "Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles," *Proc. Natl. Acad. Sci. USA*, vol. 102, 2005, pp. 15545–15550.
- [24] J. Tomfohr, J. Lu, T.B. Kepler, "Pathway level analysis of gene expression using singular value decomposition," *BMC Bioinformatics*, 6, 225, 2005.
- [25] M.C. Wu, X. Lin, "Prior biological knowledge-based approaches for the analysis of genome-wide expression profiling using gene sets and pathways," *Statistical Methods in Medical Research*, vol. 18, 2009, pp. 577–593.



# Knowledge patterns for conversion of sentences in natural language into RDF graph language

Rostislav Miarka

University of Ostrava, Faculty of Science,  
 Department of Informatics and Computers,  
 30. dubna 22, 701 03 Ostrava, Czech Republic  
 Email: rostislav.miarka@osu.cz

Martin Žáček

University of Ostrava, Faculty of Science,  
 Department of Informatics and Computers,  
 30. dubna 22, 701 03 Ostrava, Czech Republic  
 Email: martin.zacek@osu.cz

**Abstract**— This paper describes the knowledge patterns for the conversion of sentences in natural language into RDF graph language. While creating knowledge base in RDF graph language from sentences expressed in natural language, one must convert words from sentences to nodes and arcs in RDF graphs. For this conversion, it is important to know which members of a sentence represent particular words. In this paper, knowledge patterns are proposed as a tool for conversion of sentences. In order to capture knowledge patterns one can use extended RDF graph language. For the representation of knowledge patterns, further extension of this language is proposed. The paper contains four examples of knowledge patterns and their use.

## I. WORD ORDER IN NATURAL LANGUAGE

**I**N NATURAL language, people use sentences to express various statements, questions, orders etc. Each natural language is defined by its vocabulary and its grammar. Individual words are marked as vocabulary; they are words one can use in a given language. Typically, words are divided into word classes (nouns, pronouns, verbs, adverbs etc.), which determine their meaning. The grammar of a language determines the construction of sentences, which means the way of ordering particular words in sentences. Constituents of sentences (members) are basic building blocks of constructions of sentences. The basic members are subject and predicate. Other members, which extend information included in a sentence, are object, attribute and adverbial complement. The word classes of particular words are not important. It is important which member these words represent. A word-order in language, which determines the order of members in sentence, relates to the creation of sentences. Word order abides by some rules and one can identify two basic types of a word order – fixed word order and free word order. Fixed word order defines relatively strict rules of order of members in sentence. This word order is typical for Germanic languages such as English. On the other hand, free word order is greatly flexible, rules of ordering members are not so strict and they can be modified to the context of the sentence. Free word order is typical for languages which enable declension and inflexion. Among

The research described here has been financially supported by University of Ostrava grant SGS23/PfF/2011. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

these languages are Slavonic languages such as the Czech language [6], [8].

The essential members are subject and predicate, which form bare sentences. For the marking of these members the letter S is used for subject and the letter V for predicate (verb). Except of these essential members, other, elaborative members can be used: object, attribute and adverbial complement. To mark an object the letter O is used. According to the combination of subject, verb and object we can differentiate six basic word orders in sentence: SVO (subject verb object), SOV (subject object verb), VSO (verb subject object), VOS (verb object subject), OSV (object subject verb) and OVS (object verb subject). English language uses word order SVO [6], [8].

## II. CONCEPTS AND THEIR RELATIONS

The typical approach for the creation of formal ontologies is presently the concept-oriented approach. A concept is a set of objects sharing some particular features. One of the methods for searching concepts is formal concept analysis [7], which has a formal context as its input and a conceptual graph as its output. Problems of the sentence creation can be depicted as a formal context R, which is represented by a table called the data matrix, where columns are particular properties and rows are objects (types of sentences – declarative sentence, question etc.). If the table cell contains “X”, an object on the appropriate row has the property in the appropriate columns. If the cell contains “O”, objects can have the appropriate property. The data matrix for English grammar is displayed in table I; next tables describe objects and properties.

Figure 1 shows a conceptual graph based on the data matrix in table I. Each node represents one concept and it has a label consisting of two rows. Objects in this concept are in the first row, properties in this concept are in the second row. The top concept of this graph contains all objects, which share the property f (verb) – each sentence must contain a verb. The bottom concept of this graph contains all properties and no object at all, because there are no types of sentences that contain all the properties from the data matrix. Each concept, except the top concept, has its superconcepts; they are concepts connected with particular concept and are positioned above it. Each concept, except the bottom concept, has its subconcepts; they are concepts

TABLE I.  
FORMAL CONTEXT R

R	a	b	c	d	e	f	g	h	i	j
1		X				X	X	X	X	X
2		X				X	X	O	O	O
3		X		X		X	X	O	O	O
4	X	X		X		X	X	X	X	X
5	X	X		X		X	X	O	O	O
6		X		X		X	X	O	O	O
7					X	X	X	O	O	O
8				X		X		O	O	O
9				X		X	X	O	O	O
10		X				X	X	X	X	X
11		X	X			X	X	X	X	X
12		X	X	X		X	X	X	O	O

TABLE II.  
OBJECTS AND ATTRIBUTES IN FORMAL CONTEXT R

	objects
1	Basic construction of sentence
2	Affirmative declarative sentence
3	Negative declarative sentence
4	General interrogative sentence
5	Factual interrogative sentence
6	Declaratory interrogative sentence
7	Interrogative sentence for subject
8	Imperative – 2 <sup>nd</sup> person
9	Imperative 1 <sup>st</sup> and 3 <sup>rd</sup> person
10	Adverbial complements
11	Adverb of frequency
12	Adverb of frequency with complex predicate

	attributes
a	Wh – Interrogative pronoun or interrogative verb
b	S – Subject
c	AF – Adverb of frequency
d	AV – Auxiliary verb
e	S (Wh) – Interrogative pronoun
f	V – Verb
g	O – Object
h	M – Adverbial complement of Manner
i	P – Adverbial complement of Place
j	T – Adverbial complement of Time

connected with particular concept and are positioned below it. For example, the concept containing objects 1, 4, 10, 11, 12 and properties b, f, g, h has two superconcepts (first with objects 1, 2, 3, 4, 5, 6, 10, 11, 12 and with properties b, f; second with objects 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12 and with properties f, g) and two subconcepts (first with objects 1, 4, 10, 11 and with properties b, f, g, h, i, j; second with objects 11, 12 and properties b, c, f, g, h). Further, concepts containing optional properties (h, i, j) are connected with its superconcepts and subconcepts by a dashed line. In the conceptual graph shown on figure 1, we can see how the concepts are connected.

With the help of knowledge patterns the representation of natural language sentences in extended RDF graph language [2] will be further described. This representation is useful while building an ontology or knowledge base. The representation of some types of sentences does not make sense, e.g. an imperative or interrogative sentence. In the rest of this paper the representation of affirmative declarative sentences and negative declarative sentences will be described. A basic construction of a sentence is the special case of affirmative declarative sentences.

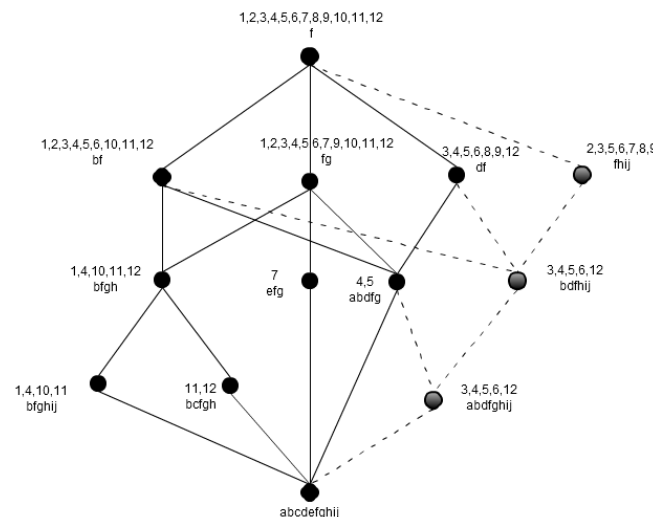


Fig. 1. Conceptual graph of formal context R

### III. KNOWLEDGE PATTERNS

The term 'knowledge pattern' was first used in [1]. While building ontologies or knowledge bases, one can see that some structures of modelled knowledge are the same. These same structures of knowledge can be captured as knowledge patterns. Knowledge patterns are general structures (patterns) of knowledge, which are not a part of the target knowledge base. They can be included into a target knowledge base by renaming their non-logical symbols. This renaming is called morphism. The morphism is an important part of using knowledge patterns.

Presently, there is no direction for capturing knowledge patterns. We propose to model knowledge patterns in RDF graph models [4], [5]. This model is simple to understand, even for amateur users. The RDF graph model is a set of RDF triples. The RDF triple consists of subject, predicate and object. Subject and object are nodes of the graph; predicate is a directed-arc from subject to object. Each more complex statement must be decomposed into individual RDF triples. In this paper, the idea of knowledge patterns for conversion of sentences in English language into RDF graph language will be introduced. For modelling of knowledge patterns extended RDF graph model introduced in [2] will be used.

RDF(S), enriched with possibility of quantification and reasoning (analogous to associative networks [9]), is an

accessible tool on conceptual level even for users that do not know OWL or other formal languages based on logic which are suitable for formalization of knowledge patterns. A transcription from RDF to OWL is then quite simple.

#### A. Representation of knowledge patterns in RDF graph language

Solid lines are used to display nodes and arcs in classical RDF graphs. To distinguish the knowledge patterns from the classical statements in RDF graph we will use dashed lines. Figure 2 shows a classical RDF triple (above) and RDF triple representing knowledge pattern (below).

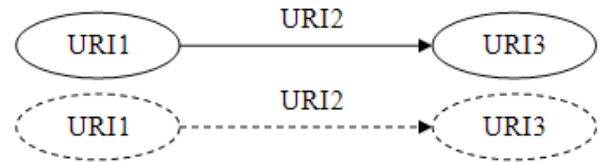


Fig. 2. RDF triples in classical RDF graph and as knowledge pattern

Apart from knowledge patterns, morphisms will be introduced for some sentences. Morphisms for mentioned knowledge patterns will be displayed as classical RDF triples. Subject of triples for morphisms will represent the term from a modelled domain, the predicate will be “isa” (Is-

TABLE III.  
SHORTCUTS FOR FULL URIs USED IN RDF GRAPHS

Shortcut	URI
subject	<a href="http://en.wikipedia.org/wiki/Subject_(grammar)">http://en.wikipedia.org/wiki/Subject_(grammar)</a>
predicate	<a href="http://en.wikipedia.org/wiki/Predicate_(grammar)">http://en.wikipedia.org/wiki/Predicate_(grammar)</a>
object	<a href="http://en.wikipedia.org/wiki/Object_(grammar)">http://en.wikipedia.org/wiki/Object_(grammar)</a>
isa	<a href="http://en.wikipedia.org/wiki/Is-a">http://en.wikipedia.org/wiki/Is-a</a>
David	<a href="http://en.wiktionary.org/wiki/David">http://en.wiktionary.org/wiki/David</a>
likes	<a href="http://en.wiktionary.org/wiki/like">http://en.wiktionary.org/wiki/like</a>
chocolate	<a href="http://en.wiktionary.org/wiki/chocolate">http://en.wiktionary.org/wiki/chocolate</a>
rdf:Bag	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag">http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag</a>
rdf:type	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#type">http://www.w3.org/1999/02/22-rdf-syntax-ns#type</a>
rdf:statement	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement">http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement</a>
rdf:subject	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#subject">http://www.w3.org/1999/02/22-rdf-syntax-ns#subject</a>
rdf:predicate	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#predicate">http://www.w3.org/1999/02/22-rdf-syntax-ns#predicate</a>
rdf:object	<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#object">http://www.w3.org/1999/02/22-rdf-syntax-ns#object</a>
complement	<a href="http://en.wikipedia.org/wiki/Adverbial_complement">http://en.wikipedia.org/wiki/Adverbial_complement</a>
extendedBy	<a href="http://en.wiktionary.org/wiki/extend">http://en.wiktionary.org/wiki/extend</a>
manner	<a href="http://en.wiktionary.org/wiki/manner">http://en.wiktionary.org/wiki/manner</a>
place	<a href="http://en.wiktionary.org/wiki/place">http://en.wiktionary.org/wiki/place</a>
time	<a href="http://en.wiktionary.org/wiki/time">http://en.wiktionary.org/wiki/time</a>
play	<a href="http://en.wiktionary.org/wiki/play">http://en.wiktionary.org/wiki/play</a>
the piano	<a href="http://en.wiktionary.org/wiki/piano">http://en.wiktionary.org/wiki/piano</a>
loudly	<a href="http://en.wiktionary.org/wiki/loudly">http://en.wiktionary.org/wiki/loudly</a>
at home	<a href="http://en.wiktionary.org/wiki/home">http://en.wiktionary.org/wiki/home</a>
daily	<a href="http://en.wiktionary.org/wiki/daily">http://en.wiktionary.org/wiki/daily</a>

A relation – relation of specialization) and the object will symbolize the general term from the knowledge pattern.

URI references are used to identify nodes and arcs in RDF graph. RDF graphs with full URI references would be too confusing to the users, therefore shortcuts usage for each node and arc are essential. These shortcuts are shown in table III.

1) *Affirmative declarative sentence*

The first described type of sentence is an affirmative declarative sentence. This sentence can have two forms. The first form is a sentence without an adverbial complement, second is one which includes it. In the first case, the sentence is in the form subject – predicate – object. This sentence forms RDF triple. The knowledge pattern for this type of sentence is shown in figure 3. This knowledge pattern will be marked as KPS1.

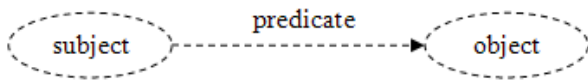


Fig. 3. KPS1

Particular members play the same role, in the RDF triple as in the sentence. Subject plays the role of subject etc. For instance, we convert the sentence “David likes chocolate.” into RDF graph language. The morphism for particular members is shown in the next figure.

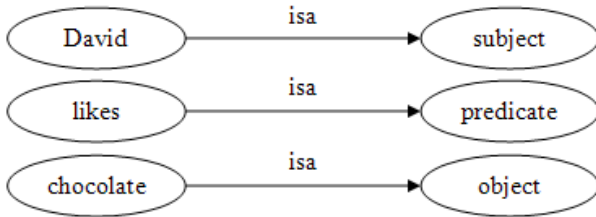


Fig. 4. KPS1 – morphism

While using the knowledge pattern KPS1 in a target knowledge base, the labels of nodes and arcs are renamed using morphism in figure 4. The resulting RDF graph (in this case, it is one RDF triple) is shown in figure 5.

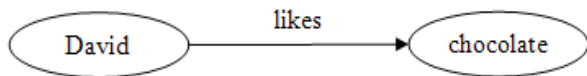


Fig. 5. KPS1 – result

The second form of affirmative declarative sentence contains an adverbial complement, which extends the meaning of the verb in this sentence. It can be an adverbial complement of manner, place or time. The sentence can contain one, two or all three types of adverbial complement. In terms of RDF triples, it means that adverbial complements extend the whole RDF triple containing the appropriate verb. In order to represent the whole RDF triple the RDF data model offers the predefined resource `rdf:statement` and

to represent parts of this statement it offers the predefined properties `rdf:subject`, `rdf:predicate` and `rdf:object` (full URIs for predefined resource and properties are in table III). When capturing an affirmative declarative sentence with adverbial complements as the knowledge pattern, the RDF container `rdf:Bag` is used (full URI is in table III). The knowledge pattern of an affirmative declarative sentence with adverbial complements is shown in figure 6 and will be marked as KPS2.

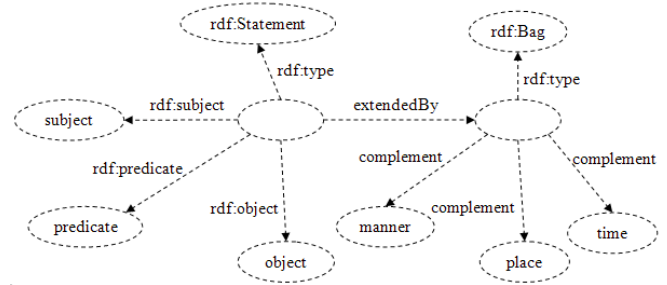


Fig. 6. KPS2

Furthermore, it is important to note that 'subject' and `rdf:subject` in KPS2 are not the same thing. The node with the 'subject' label represents the member of the sentence subject, while the arrow with label `rdf:subject` represents the predefined property from the RDF data model. Similarly, it stands both for the pair predicate – `rdf:predicate` and for the pair object – `rdf:object`.

While using pattern KPS2, the morphism contains only renaming the following symbols: subject, predicate, object, manner, place and time. The remaining symbols do not change. Let us consider the sentence “David plays the piano loudly at home daily.”. The morphism for this sentence is shown on figure 7.

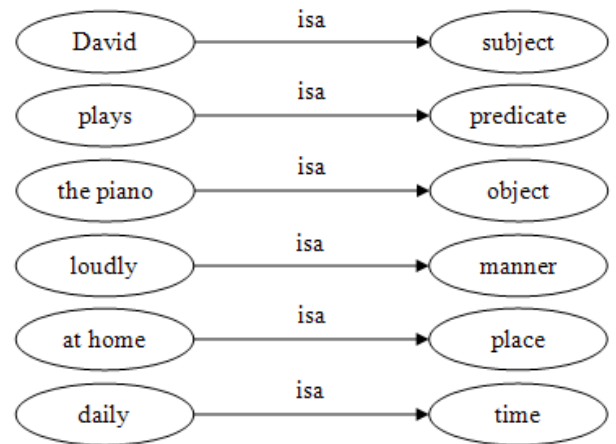


Fig. 7. KPS2 – morphism

After applying this morphism to the knowledge pattern, the symbols are renamed and the target knowledge base will contain the resulting RDF graph (figure 8).



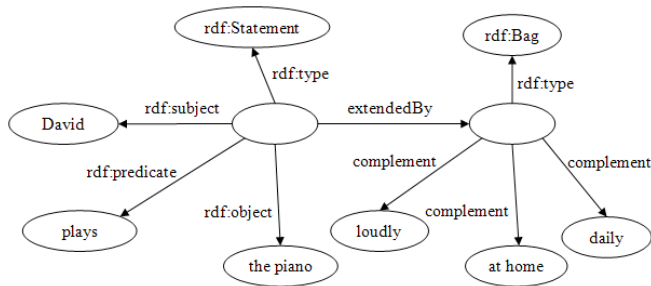


Fig. 8. KPS2 – result

2) *Negative declarative sentence*

The second described type of sentence is a negative declarative sentence. As well as an affirmative declarative sentence, this type of sentence can have two forms; the first form contains only subject, predicate and object, the second form contains an adverbial complement as well.

Extended RDF graph language [2] allows user to express negation of a statement. Negation is expressed by the help of special symbol called falsum (notation  $\otimes$ ), which is false in all interpretations. Falsum is always bound to predicate.

The knowledge pattern for a negative declarative sentence without adverbial complements (KPS3) is shown in figure 9. The pattern is very similar to pattern KPS1, the only difference is that the arrow representing predicate is marked by falsum.

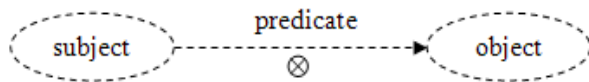


Fig. 9. KPS3

For simplicity, let us consider the knowledge base contains negation of a sentence, stated as an example in KPS1, i. e. the sentence “David does not like chocolate.”. The morphism for this sentence (figure 10) is very similar to the morphism for affirmative sentences. To denote that the predicate is in its negative form, the falsum is inside the node representing the predicate.

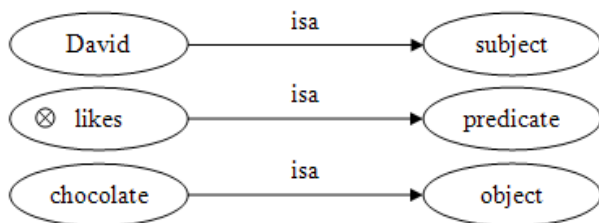


Fig. 10. KPS3 – morphism

After renaming the symbols, the resulting RDF graph contains one RDF triple (figure 11).



Fig. 11. KPS3 – result

The second form of negative declarative sentences contains adverbial complements. It can be an adverbial complement of manner, place, time or more adverbial complements together. The knowledge pattern for this type of sentence will be marked as KPS4 and is very similar to KPS2. The only difference is, again, in using falsum for marking the predicate in the negative form. Falsum is inside the node representing the predicate. KPS4 is shown on figure 12.

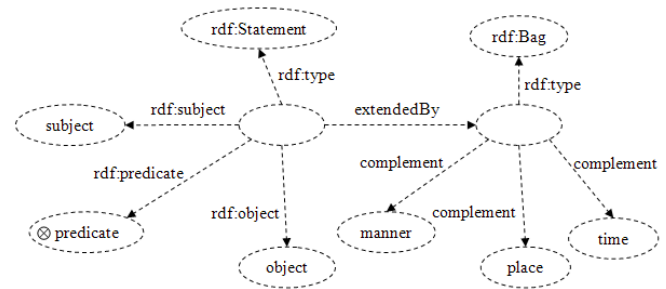


Fig. 12. KPS4

While using this knowledge pattern in a concrete knowledge base, let us consider, again for simplicity, the negation of sentence shown while using of KPS2, i. e. sentence “David does not play the piano loudly at home daily.”. The morphism for this sentence is shown in figure 13.

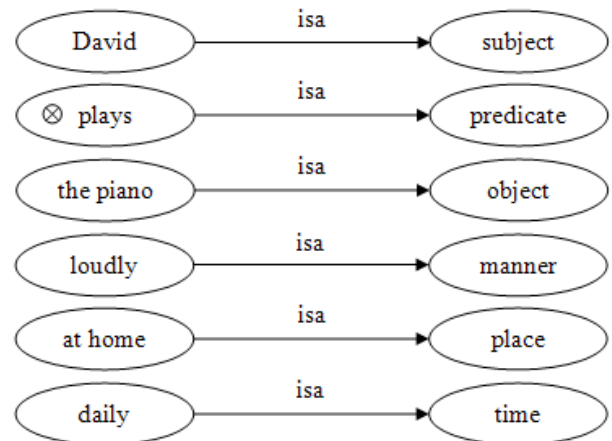


Fig. 13. KPS4 – morphism

The resulting RDF graph (figure 14) was created by renaming the symbols listed in the morphism.

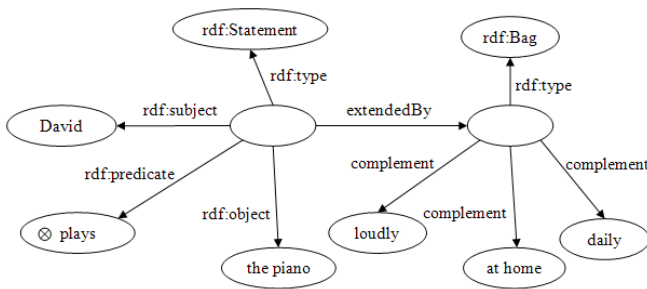


Fig. 14. KPS4 – result

#### IV. CONCLUSION AND FUTURE WORK

While creating a knowledge base from sentences expressed in natural language, it is important to determine members of particular words in a sentence. The member of a sentence determines the position of a word in the RDF graph. In this paper, it was proposed that the help of knowledge patterns would be a way of converting natural language sentences into RDF graph language. Knowledge patterns are general structures of knowledge, which are not part of the target knowledge base or ontology. An essential part of the use of the knowledge patterns is the specification of renaming non-logical symbols. This part is called morphism. This paper introduced a way of capturing knowledge patterns in extended RDF graph language. A dashed line was used for the representation of knowledge patterns in RDF graphs (in

contrast to solid line in classical RDF graphs). Several examples of knowledge patterns and their use are part of this paper.

Future work will be focused on discovering other knowledge patterns and the representation of these knowledge patterns in the RDF/XML languages.

#### REFERENCES

- [1] P. Clark, J. Thompson and B. Porter, "Knowledge patterns," in *Handbook on Ontologies*, S. Staab and R. Studer, Eds. Berlin: Springer-Verlag, 2004, ISBN 3-540-40834-7, pp. 191–207.
- [2] A. Lukasová, M. Vajgl and M. Žáček, "Reasoning in RDF graphic formal system with quantifier," *Proceedings of the International Multiconference on Computer Science and Information Technology*, 2010, ISBN 978-83-60810-22-4, pp. 67–72.
- [3] S. Staab, M. Erdmann, A. Maedche and S. Decker "Ontologies in RDF(S)," [online] <http://www.ida.liu.se/ext/etai/received/semaweb/010/paper.pdf>.
- [4] W3C: Resource Description Framework (RDF): Concepts and Abstract Syntax, <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>
- [5] W3C: RDF Primer, <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>
- [6] M. S. Dryer, "Order of Subject, Object, and Verb", <http://linguistics.buffalo.edu/people/faculty/dryer/dryer/DryerWalsSOVNoMap.pdf>
- [7] B. Ganter and R. Wille, "Formal Concept Analysis: Mathematical Foundations," Berlin: Springer-Verlag, 2004, ISBN 3-540-62771-5.
- [8] D. Crystal, "The Cambridge Encyclopedia of Language (2nd edition ed.)", Cambridge: Cambridge University Press, 1997, ISBN 0-521-55967-7.
- [9] A. Lukasová, "Reprezentace znalostí v asociativních sítích," *Znalosti 2001*, ISBN 80-245-0190-2, pp. 245-250. (in Czech)

# Graph Mining Approach to Suspicious Transaction Detection

Krzysztof Michalak, Jerzy Korczak

Institute of Business Informatics

Wroclaw University of Economics, Wroclaw, Poland

Email: {krzysztof.michalak, jerzy.korczak}@ue.wroc.pl

**Abstract**—Suspicious transaction detection is used to report banking transactions that may be connected with criminal activities. Obviously, perpetrators of criminal acts strive to make the transactions as innocent-looking as possible. Because activities such as money laundering may involve complex organizational schemes, machine learning techniques based on individual transactions analysis may perform poorly when applied to suspicious transaction detection.

In this paper, we propose a new machine learning method for mining transaction graphs. The method proposed in this paper builds a model of subgraphs that may contain suspicious transactions. The model used in our method is parametrized using fuzzy numbers which represent parameters of transactions and of the transaction subgraphs to be detected. Because money laundering may involve transferring money through a variable number of accounts the model representing transaction subgraphs is also parametrized with respect to some structural features. In contrast to some other graph mining methods in which graph isomorphisms are used to match data to the model, in our method we perform a fuzzy matching of graph structures.

## I. INTRODUCTION

**F**INANCIAL institutions such as banks are legally obliged to monitor activities of their customers and to report events that may indicate involvement in a criminal act. In the case of bank transactions monitoring, one of the goals is to detect money laundering activities i.e. activities aimed at concealing the origin of illegally-obtained money.

There are several difficulties in money laundering detection. People involved in money laundering obviously try to conceal the real purpose of money transfers used in this process. Therefore, one can expect that individual transactions will not clearly stand out from amongst other bank transfers. The probability of a fraud depends not only on parameters of individual bank transfer but also on relations with other transfers and the entities that send them.

The volume and value of transactions reported as suspicious are very high. For example, the value of transactions reported to the anti-money laundering watchdog by Russian financial institutions in the first nine months of 2010 was 120 trillion roubles (2.44 trillion pounds, 3.8 trillion dollars) [9]. 5.6 million filings were made by banks, insurance companies and financial service companies in this period.

In the area of fraud detection it is typical that events to be detected are in considerable minority compared to the overall amount of data. For example, in 2010 there were

593,819 fraudulent credit card transactions detected in Australia worth in total \$145,854,208 [7]. In the same year over 1.4 billion credit card transactions were made for a total amount of more than 195 billion dollars (values calculated using statistics from [8]). The same sources report 241,063 fraudulent credit card transactions totalling \$85,215,615 in 2006 and a total number of over 1.1 billion transactions worth in total over 155 billion dollars. In the above-cited cases fraudulent transactions constitute only 0.02 – 0.04% of all transactions which means that data manifest extreme class imbalance. This in itself creates a significant challenge for many machine learning methods.

Typically, the money laundering process is divided into three stages: placement, layering and integration [5]. Each of the stages involves various schemes of money transferring between bank accounts. These schemes can be identified as subgraphs in the transaction graph. Figure 1 shows two simple examples of subgraphs which represent transferring money via a number of intermediaries.

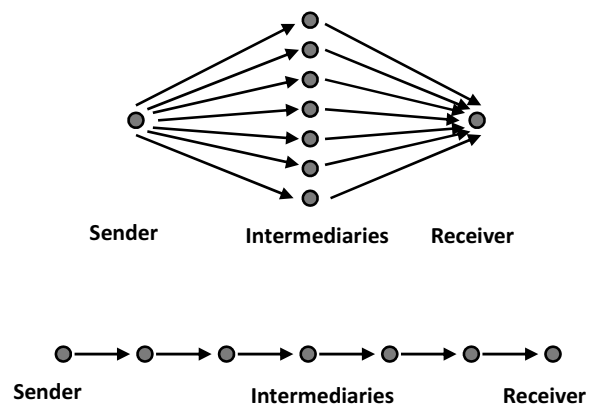


Fig. 1. Examples of subgraphs which may indicate money-laundering activities

In the first case a larger amount is split to smaller transfers in order to decrease the probability that individual transactions will be reported as suspicious. The second case serves the purpose of obscuring the connection between the sender and the receiver. The entire money laundering operation may involve many such schemes, so identification of a suspicious subgraph may help in uncovering much larger network of ille-

gal transactions. Money laundering operations are intertwined with many other transactions, including legal ones. Figure 2 shows a small subgraph of transactions (in which vertices are shaded in gray) that may raise suspicion because funds are transferred from one account to another via many independent transaction chains. It is clear that accounts participating in what may turn out to be illegal transaction structuring may as well be engaged in other, probably harmless activities.

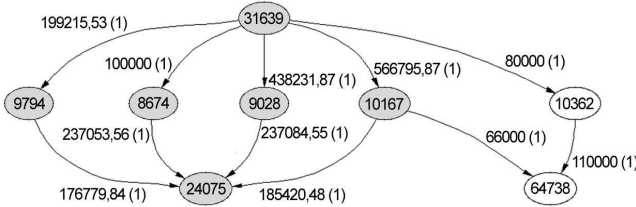


Fig. 2. Suspicious transaction graph connected with other, possibly legal activities. Vertex labels are account identifiers, edge labels contain number of transactions between the two accounts and the total amount transferred.

## II. PROPOSED METHOD

Money laundering is hard to detect because the occurrences are very infrequent in the graph of all transactions and individual bank transfers involved in a money laundering scheme are structured so that they appear as legitimate. These obstacles are especially hard to overcome if the detection process is based only on features of individual transactions. Therefore, graph mining methods seem to be interesting, because they can detect complex dependencies between transactions. It is also possible to take into account properties and relations of entities involved in sending and receiving the transfers.

We present a method for graph structure learning using a model which can be trained on a previously annotated graph of transactions and then can be matched against a graph of unannotated transactions in order to select a number of transactions for a more thorough checking by human expert. This is in agreement with the mode of operation of financial watchdog institutions which employ experts to scrutinize suspicious transactions. Because of a huge number of transactions the work of an expert can be made significantly more efficient if a computer system is able to suggest a limited number of transactions to be checked for indications of possible money laundering. The results of the expert's work may in turn be used to train the system again.

The work presented in this paper is a part of a larger research project carried out at our Institute aimed at developing various money laundering detection methods. One of the research topics in this project is money laundering detection in data warehouses [3], [4]. Suspicious transactions discovered in data warehouses can be used as training data for the method described in this paper.

As described in the introduction, organizational schemes involved in money laundering may vary to a great extent with respect to the number of transactions used to conceal the nature of the activity. Therefore, we propose a model which can

adapt to training data with respect not only to transaction and account parameters but also with respect to the graph structure.

Proposed model represents subgraphs similar in structure to the graph presented in Figure 3.

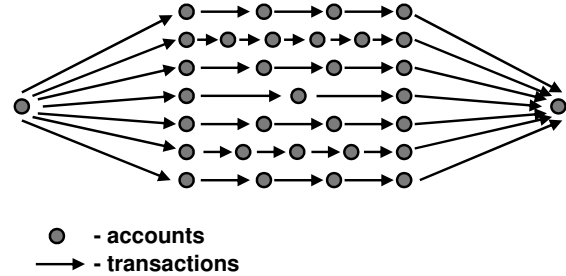


Fig. 3. General structure of a detected subgraph

In order to model such subgraphs, hierarchical three-level patterns are generated. Some of the parameters of the model are polygonal fuzzy numbers [1]. They are denoted using the hat ( $\hat{\cdot}$ ) symbol. In the proposed method we use simplified polygonal fuzzy numbers which use only 6 real numbers  $x_1$ ,  $x_2$ ,  $x_3$ ,  $x_4$ ,  $m_2$  and  $m_3$ . Membership function  $\mu_{\hat{x}}$  of such fuzzy number  $\hat{x}$  is presented in Figure 4 and is calculated as follows:

$$\mu_{\hat{x}}(x) = \begin{cases} 0 & \text{for } x \leq x_1, \\ m_2 \cdot \frac{x-x_1}{x_2-x_1} & \text{for } x \in (x_1, x_2], \\ m_2 + (m_3 - m_2) \cdot \frac{x-x_2}{x_3-x_2} & \text{for } x \in (x_2, x_3], \\ m_3 \cdot \frac{x_4-x}{x_4-x_3} & \text{for } x \in (x_3, x_4), \\ 0 & \text{for } x > x_4. \end{cases} \quad (1)$$

where:

$\hat{x} = \langle x_1, x_2, x_3, x_4, m_2, m_3 \rangle$  - polygonal fuzzy number.

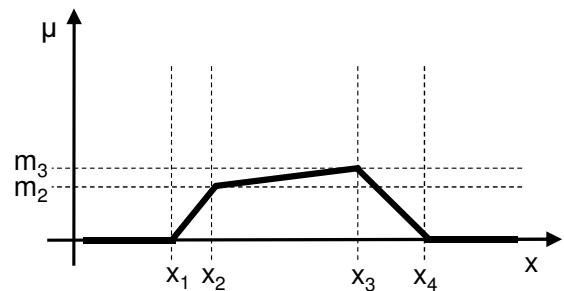


Fig. 4. Polygonal fuzzy number with components  $x_1$ ,  $x_2$ ,  $x_3$ ,  $x_4$ ,  $m_2$  and  $m_3$

The lowest level of the model is a *TR* pattern that describes a single transaction:

$$TR = \langle \hat{a}, r(\cdot), s(\cdot) \rangle, \quad (2)$$

where:

$\hat{a}$ —polygonal fuzzy number representing transaction amount,

$s(\cdot)$ —function assigning weights to classes to which entities sending transfers belong,

$r(\cdot)$ —function assigning weights to classes to which entities receiving transfers belong.

The middle level is a *SER* pattern that describes transaction chains in which transactions are connected in series:

$$SER = \langle \hat{m}, \hat{\delta} \rangle, \quad (3)$$

where:

$\hat{m}$ —polygonal fuzzy number representing the number of transactions in a chain,

$\hat{\delta}$ —polygonal fuzzy number representing the ratio of amount transferred in the last transaction to the amount transferred in the first transaction.

The top level is a *PAR* pattern that describes parallel transaction chains (which are described by the *SER* pattern)

$$PAR = \langle \hat{n}, \hat{\Delta}, \tau \rangle, \quad (4)$$

where:

$\hat{n}$ —polygonal fuzzy number representing the number of transactions in a chain,

$\hat{\Delta}$ —polygonal fuzzy number representing the ratio of the sum of amounts received by the receiving account to the sum of amounts sent from the sending account.

$\tau$ —acceptance threshold used for deciding which transaction subgraphs match the pattern.

A complete pattern contains one set of parameters for each of the three levels.

Using such patterns, transaction subgraphs are evaluated in the following manner. First, weights are assigned to individual transfers using parameters of the *TR* pattern. A transfer  $T$  of amount  $a$  sent by an entity belonging to a class  $c_s$  to an entity belonging to a class  $c_r$  is assigned a weight  $w_{TR}$  which is calculated as:

$$w_{TR}(T) = \mu_{\hat{a}}(a) \cdot s(c_s) \cdot r(c_r), \quad (5)$$

where:

$TR = \langle \hat{a}, s(\cdot), r(\cdot) \rangle$  - pattern to which the transaction is matched,

$\mu_{\hat{a}}(\cdot)$ —membership function of the fuzzy number  $\hat{a}$ .

Transaction chains are evaluated using parameters of the *SER* pattern. A transaction chain  $L$  of length  $m$  in which the ratio of amount transferred in the last transaction to the amount transferred in the first transaction equals  $\delta$  is assigned a weight  $w_{SER}$  which is calculated as:

$$w_{SER}(L) = \frac{\sum_{T \in L} w_{TR}(T)}{m} \cdot \mu_{\hat{m}}(m) \cdot \mu_{\hat{\delta}}(\delta), \quad (6)$$

where:

$SER = \langle \hat{n}, \hat{\delta} \rangle$  - pattern to which the transaction chain is matched,

$TR$ —pattern used to match individual transactions,

$T$ —transaction which belongs to the transaction chain  $L$ ,

$\mu_{\hat{m}}(\cdot)$ —membership function of the fuzzy number  $\hat{m}$ ,

$\mu_{\hat{\delta}}(\cdot)$ —membership function of the fuzzy number  $\hat{\delta}$ .

Subgraphs consisting of parallel paths are evaluated using parameters of the *PAR* pattern. A subgraph  $P$  containing  $n$  parallel paths in which the ratio of the sum of amounts received by the receiving account to the sum of amounts sent from the sending account equals  $\Delta$  is assigned a weight  $w_{PAR}$  which is calculated as:

$$w_{PAR}(P) = \frac{\sum_{L \in P} w_{SER}(L)}{n} \cdot \mu_{\hat{n}}(n) \cdot \mu_{\hat{\Delta}}(\Delta), \quad (7)$$

where:

$PAR = \langle \hat{n}, \hat{\Delta}, \tau \rangle$  - pattern to which the transaction subgraph is matched,

$SER$  - pattern used to match transaction chains,

$L$  - transaction chain which belongs to the subgraph  $P$ ,

$\mu_{\hat{n}}(\cdot)$  - membership function of the fuzzy number  $\hat{n}$ ,

$\mu_{\hat{\Delta}}(\cdot)$  - membership function of the fuzzy number  $\hat{\Delta}$ .

The entire *PAT* pattern includes all the parameters required to match graph elements at each of three levels:

$$PAT = \langle TR, SER, PAR \rangle, \quad (8)$$

$$PAT = \langle \hat{a}, r(\cdot), s(\cdot), \hat{m}, \hat{\delta}, \hat{n}, \hat{\Delta}, \tau \rangle. \quad (9)$$

Fuzzy parameters  $\hat{a}$ ,  $\hat{m}$ ,  $\hat{\delta}$ ,  $\hat{n}$  and  $\hat{\Delta}$  can be described by 6 real numbers each. Functions  $r(\cdot)$  and  $s(\cdot)$  have discrete domains and thus are adequately represented by discrete sets of weights. The entire *PAT* pattern is thus described by  $31 + 2k$  real numbers, where  $k$  is the number of entity classes.

Model parameters ( $31 + 2k$  real numbers describing a *PAT* pattern) have to be adjusted based on a training data set containing transactions annotated by an expert. For optimization of parameters of *PAT* patterns we propose to use a genetic algorithm with the following properties.

**Specimen**—a set of  $31 + 2k$  real numbers interpreted as *PAT* pattern parameters,

**Mutation**—each of the  $31 + 2k$  real numbers in each of the specimens is mutated with equal probability  $P_{mut}$ .

Mutation of the numbers that represent fuzzy number components  $x_1$ ,  $x_2$ ,  $x_3$  and  $x_4$  is controlled by additional parameters  $\Delta_x$ ,  $m_x$  and  $M_x$  defined separately for each fuzzy parameter of the model (i.e.  $\hat{a}$ ,  $\hat{m}$ ,  $\hat{\delta}$ ,  $\hat{n}$  and  $\hat{\Delta}$ ). First, a random value  $d$  is drawn with uniform probability from the range  $[-\frac{\Delta_x}{2}, \frac{\Delta_x}{2}]$ . Then, value of the component  $x_i$  is modified by adding  $d$ . As this modification may disrupt the order of fuzzy number components  $x_1$ ,  $x_2$ ,  $x_3$  and  $x_4$  and may also lead to violation of constraints  $m_x \leq x_1$  and  $x_4 \leq M_x$  a check (and possibly also a correction) must be performed. This correction is performed as follows:

- if  $x_1 < m_x$  then  $x_1 \leftarrow m_x$
- for  $i = 1, 2, 3$  : if  $x_{i+1} \leq x_i$  then  $x_{i+1} \leftarrow x_i + 0.001$
- if  $x_4 > M_x$  then  $x_4 \leftarrow M_x$
- for  $i = 3, 2, 1$  : if  $x_i \geq x_{i+1}$  then  $x_i \leftarrow x_{i+1} - 0.001$

Mutation of the numbers that represent fuzzy number parameters  $m_2$  and  $m_3$ , weights assigned to entity classes and the value of acceptance threshold  $\tau$  is performed by adding a value drawn with uniform probability from the range  $[-0.005, 0.005]$  and ensuring that the result is in the range  $[0, 1]$ .

**Selection**—a standard roulette-wheel selection procedure [6] is used.

**Crossover**—a standard single-point crossover operator [2] is used. Probability of a crossover being performed on any two specimens is controlled by the parameter  $P_{cross}$ .

**Evaluation function**—the evaluation function for a given specimen  $S$  is calculated in the following way:

- from the specimen  $S$  a pattern  $PAT(S) = \langle TR(S), SER(S), PAR(S) \rangle$  is constructed using  $31 + 2k$  real numbers as parameters,
- a set  $\mathcal{P}$  is constructed containing those subgraphs  $G$  that match the pattern  $PAT(S)$  and have a weight  $w_{PAR(S)}(G) > \tau$ ,
- for each subgraph  $G \in \mathcal{P}$  a total number of transactions  $t_n(G)$  in this subgraph and a sum of weights of transactions  $t_w(G)$  in this subgraph are calculated. Weights of transactions are based on annotations made by the expert. For example, transactions annotated as "illegal" may have a weight 1.0, transactions annotated as "legal" a weight 0.0 and transactions annotated as "not classified" a weight 0.1.
- evaluation of the specimen is calculated as:

$$F(S) = \frac{\sum_{G \in \mathcal{P}} w_{PAR(S)}(G) \cdot t_w(G)}{\sum_{G \in \mathcal{P}} t_n(G)}. \quad (10)$$

The specimen (or specimens) achieving the highest values of evaluation function  $F$  may be used to identify suspicious transactions in previously unseen data.

### III. EXPERIMENTS

For experiments training and testing data sets containing transactions annotated by an expert are required. Because such data sets are hard to obtain due to confidentiality of banking data, the experiments performed so far were based on artificially-generated data. In order to build a data set containing transactions similar to those encountered in real-life we used a model which represents transactions in a "mini-economy" during a period of one year. In this model, three classes of economic entities are defined: companies, individual persons and offices (tax offices and social security offices). Economic entity classes are characterized by probability distributions which are used to generate parameters for instances belonging to each class.

Companies are characterized by the following probability distributions:

- distribution of the number of employees:  $N(m_E, \sigma_E)$ ,
- distribution of salary:  $N(m_S, \sigma_S)$ ,
- distribution of the number of goods sold per year:  $N(m_G, \sigma_G)$ ,

- distribution of prices of goods:  $N(m_P, \sigma_P)$ .

A predefined number of companies  $n_c$  is generated. For each company a number of employees  $n_E$  is drawn from the Gaussian distribution  $N(m_E, \sigma_E)$ . Then,  $n_E$  persons are added to the model. For each of the 12 months in a year a salary  $a_s$  is drawn from the Gaussian distribution  $N(m_S, \sigma_S)$  and a transaction representing the payment (with the amount  $a_s$ ) is generated.

Next, buying of goods is simulated. For each company a number of goods sold during the simulated year  $n_G$  is drawn from the Gaussian distribution  $N(m_G, \sigma_G)$ . For each good a price  $a_p$  is drawn from the Gaussian distribution  $N(m_P, \sigma_P)$ . A buyer is selected at random from all employees of all companies and a new transaction (with the amount  $a_p$  - a payment for the good) is added.

Generation of offices is controlled by the parameters  $n_T$  - the number of tax offices and  $n_F$ —the number of social security offices. Offices are also characterized by two probability distributions:

- distribution of tax rate:  $N(m_T, \sigma_T)$ ,
- distribution of social security fee rate:  $N(m_F, \sigma_F)$ .

One tax office and one social security office are assigned at random to each company. The sum of payments  $C_p$  received by the company for goods sold in each month is calculated and a tax rate  $\alpha_T$  is drawn from the Gaussian distribution  $N(m_T, \sigma_T)$ . Tax amount  $a_T$  is calculated as  $a_T = C_p \cdot \alpha_T / 100$  and a transaction sent by the company to the tax office account is generated. Social security fee  $a_F$  is calculated as  $a_F = C_s \cdot \alpha_F / 100$  based on the sum of salaries in each month  $C_s$  and a social security fee rate  $\alpha_F$  which is drawn from the Gaussian distribution  $N(m_F, \sigma_F)$ .

The steps described above produce a set of transactions representing the usual activities observed in economy. To this set of transactions  $n_{ML}$  money laundering schemes are added. Each of these schemes consists of a sender, a number  $n_B$  of intermediaries and a receiver. Transfers are sent from the sender to one intermediary and then to the receiver. Each parallel path goes through one intermediary only, so  $n_B$  parallel paths are created. The generation of the money laundering schemes is characterized by the following probability distributions:

- distribution of the amount sent from the sender to one intermediary:  $N(m_Q, \sigma_Q)$ ,
- distribution of the number of intermediaries (equal to the number of parallel paths):  $N(m_B, \sigma_B)$ ,
- distribution of the fraction of the amount received by the intermediary that is forwarded to the receiver:  $N(m_\Delta, \sigma_\Delta)$ .

Generated transactions are annotated in the following manner:

- legal—tax and social security fee transactions,
- illegal—transactions belonging to the generated money laundering schemes,
- unknown—all the remaining transactions (salaries and payments for goods).

Using data generation method described above, we have generated four data sets:  $SMALL_A$ ,  $SMALL_B$ ,  $LARGE_A$ ,  $LARGE_B$ . All data sets were generated using the same parameters for offices:  $n_T = 5$ ,  $n_F = 5$ ,  $m_T = 15$ ,  $\sigma_T = 1.5$ ,  $m_F = 20$ ,  $\sigma_F = 2.0$ . Also, parameters of money laundering were the same:  $m_Q = 5000$ ,  $\sigma_Q = 1000$ ,  $m_B = 40$ ,  $\sigma_B = 10$ ,  $m_\Delta = 1.0$ ,  $\sigma_\Delta = 0.1$ . These data sets contain three classes of companies that can be briefly characterized as large (L), medium (M) and small (S). The  $SMALL$  and  $LARGE$  data sets differ in the number of companies in each class. Parameters controlling generation of companies for data sets  $SMALL_A$  and  $SMALL_B$  and for data sets  $LARGE_A$  and  $LARGE_B$  are summarized in Tables I and II respectively.

TABLE I

PARAMETERS CONTROLLING GENERATION OF COMPANIES FOR DATA SETS  $SMALL_A$  AND  $SMALL_B$

Parameter	Company class		
	large	medium	small
$n_C$	2	4	25
$m_E$	5 000	500	50
$\sigma_E$	1 000	100	20
$m_S$	6 000	5 000	4 000
$\sigma_S$	1 500	1 200	1 000
$m_G$	100 000	1 000	100
$\sigma_G$	30 000	300	30
$m_P$	50	500	500
$\sigma_P$	10	100	100

TABLE II

PARAMETERS CONTROLLING GENERATION OF COMPANIES FOR DATA SETS  $LARGE_A$  AND  $LARGE_B$

Parameter	Company class		
	large	medium	small
$n_C$	2	8	100
$m_E$	5 000	500	50
$\sigma_E$	1 000	100	20
$m_S$	6 000	5 000	4 000
$\sigma_S$	1 500	1 200	1 000
$m_G$	1 000 000	10 000	1 000
$\sigma_G$	300 000	3 000	300
$m_P$	50	500	500
$\sigma_P$	10	100	100

The number of accounts and transactions of each type in each of the data sets is summarized in Table III.

In the experiments, one of data sets in a  $LARGE/SMALL$  pair was used for training and the other one for testing. A population of  $N_{pop} = 20$  specimens was trained for  $N_{gen} = 20$  generations of genetic algorithm. Crossover and mutation probabilities were set to  $P_{cross} = 0.1$  and  $P_{mut} = 0.01$  respectively. Parameters controlling mutation of  $x_i$  components of fuzzy numbers are summarized in Table IV.

Specimen evaluation requires that transaction annotations are converted to numerical weights. In the experiments a weight 1.0 was assigned to “illegal” transactions,

TABLE III

THE NUMBER OF ACCOUNTS AND TRANSACTIONS OF EACH TYPE IN EACH OF THE DATA SETS

Object type	Number of objects			
	$SMALL_A$	$SMALL_B$	$LARGE_A$	$LARGE_B$
Accounts	11 238	11 401	19 261	21 270
companies	31	31	110	110
offices	10	10	10	10
personal	11 197	11 360	19 261	21 150
Transactions	294 972	383 463	2 854 965	2 625 671
legal	744	744	2 640	2 640
unknown	289 336	377 207	2 848 435	2 619 049
illegal	4 892	5 512	3 890	3 982
annot. ratio	0.0191	0.0166	0.0023	0.0025

TABLE IV

PARAMETERS CONTROLLING MUTATION OF  $x_i$  COMPONENTS OF FUZZY NUMBERS

Fuzzy number	Parameter		
	$\Delta_x$	$m_x$	$M_x$
$\hat{a}$	200	range not limited	
$\hat{m}$	$x_i$ not mutated, fixed at 0, 1, 2 and 3, only $m_2$ and $m_3$ are mutated		
$\hat{\delta}$	0.1	0.5	1.5
$\hat{n}$	2	3	100
$\hat{\Delta}$	0.1	0.5	1.5

a weight 0.0 to transactions annotated as “legal” and a weight 0.1 to transactions annotated as “not classified.”

Tests were performed in 10 independent iterations for each pair of  $SMALL$  and  $LARGE$  data sets. In each iteration, after the training has been completed, the best specimen (with the highest value of the evaluation function) was selected from the population and it was used for selecting suspicious transaction subgraphs from the testing data set. Only one, the best specimen, was used, because we wanted to limit the number of transactions marked as suspicious. In the real-life environment, transactions marked as suspicious are reviewed by human expert which obviously imposes limitations on the number of transactions that can be processed. The number of transactions that were actually legal, illegal and of unknown status (according to annotations) was used to measure the quality of the detection. Results are summarized in Tables V-VIII.

During the experiments execution time of test iterations was recorded. Measured values are summarized in Table IX.

A meaningful comparison of execution times is only possible for tests performed on the same machine. Therefore, only three tests that were performed on the same computer are included in the table. For comparison, the number of accounts and the number of transactions in each data set are also presented in the table.

TABLE V

THE NUMBER OF "LEGAL", "NOT CLASSIFIED" AND "ILLEGAL" TRANSACTIONS MARKED AS SUSPICIOUS IN THE EXPERIMENT WITH TRAINING DATA SET *SMALL<sub>A</sub>*, TEST DATA SET *SMALL<sub>B</sub>* AND EACH ITERATION PERFORMED INDEPENDENTLY. PERCENTAGE OF "NOT CLASSIFIED" TRANSACTIONS AMONG TRANSACTIONS MARKED AS SUSPICIOUS: 15.98%

Iteration	Number of transactions		
	<i>legal</i>	<i>unknown</i>	<i>illegal</i>
1	0	0	26
2	0	9	9
3	0	8	8
4	0	0	30
5	0	0	26
6	0	0	26
7	0	0	46
8	0	11	11
9	0	7	7
10	0	0	30
TOTAL	0	35	219

TABLE VI

THE NUMBER OF "LEGAL", "NOT CLASSIFIED" AND "ILLEGAL" TRANSACTIONS MARKED AS SUSPICIOUS IN THE EXPERIMENT WITH TRAINING DATA SET *SMALL<sub>B</sub>*, TEST DATA SET *SMALL<sub>A</sub>* AND EACH ITERATION PERFORMED INDEPENDENTLY. PERCENTAGE OF "NOT CLASSIFIED" TRANSACTIONS AMONG TRANSACTIONS MARKED AS SUSPICIOUS: 13.88%

Iteration	Number of transactions		
	<i>legal</i>	<i>unknown</i>	<i>illegal</i>
1	0	10	10
2	0	0	34
3	0	10	10
4	0	0	30
5	0	0	32
6	0	0	36
7	0	0	34
8	0	10	11
9	0	9	9
10	0	0	36
TOTAL	0	39	242

#### IV. CONCLUSION

In this paper we presented a graph mining method intended for detection of suspicious transactions. Contrary to data mining methods based solely on transaction features the method proposed in this paper takes into consideration those dependencies between individual transfers that may be indicative of illegal activities. We expect this feature to be a significant advantage, because single transactions are often tailored by criminals in order to be as innocent-looking as possible.

Results of the experiments suggest that the method proposed in the paper has several advantageous properties:

- in the experiments the proposed method managed to avoid marking as suspicious of any transactions that were annotated as "legal". Although it is not clear at this point to what extent this quality will persist in the case of other data sets, this is a desirable behaviour for a suspicious transaction detection method;

TABLE VII

THE NUMBER OF "LEGAL", "NOT CLASSIFIED" AND "ILLEGAL" TRANSACTIONS MARKED AS SUSPICIOUS IN THE EXPERIMENT WITH TRAINING DATA SET *LARGE<sub>A</sub>*, TEST DATA SET *LARGE<sub>B</sub>* AND EACH ITERATION PERFORMED INDEPENDENTLY. PERCENTAGE OF "NOT CLASSIFIED" TRANSACTIONS AMONG TRANSACTIONS MARKED AS SUSPICIOUS: 27.05%

Iteration	Number of transactions		
	<i>legal</i>	<i>unknown</i>	<i>illegal</i>
1	0	10	10
2	0	11	11
3	0	0	22
4	0	0	22
5	0	9	9
6	0	0	24
7	0	0	22
8	0	0	22
9	0	12	12
10	0	24	24
TOTAL	0	66	178

TABLE VIII

THE NUMBER OF "LEGAL", "NOT CLASSIFIED" AND "ILLEGAL" TRANSACTIONS MARKED AS SUSPICIOUS IN THE EXPERIMENT WITH TRAINING DATA SET *LARGE<sub>B</sub>*, TEST DATA SET *LARGE<sub>A</sub>* AND EACH ITERATION PERFORMED INDEPENDENTLY. PERCENTAGE OF "NOT CLASSIFIED" TRANSACTIONS AMONG TRANSACTIONS MARKED AS SUSPICIOUS: 21.05%

Iteration	Number of transactions		
	<i>legal</i>	<i>unknown</i>	<i>illegal</i>
1	0	0	26
2	0	9	9
3	0	0	28
4	0	9	9
5	0	0	28
6	0	10	10
7	0	0	24
8	0	0	26
9	0	10	10
10	0	10	10
TOTAL	0	48	180

- more than 2/3 of transactions marked as suspicious were actually involved in money laundering schemes;
- as shown in Table IX computation time doubled with the similar—twofold increase in the number of accounts. The difference in transaction number between the same two data sets was about 10 times. This is beneficial, because the volume of transactions in historic record increases more rapidly than the number of bank customers;

The fact that the proposed method did not mark any legal transactions as suspicious is promising. It is easy to understand that reporting transactions which are considered legal by human experts most probably means raising a false alarm. A much harder question is, how large percentage of transactions that are unannotated by human expert (or annotated as "not classified") should be marked as suspicious by the algorithm. In the case of artificial data used in this paper it is obvious that all "not classified" transactions were,



TABLE IX

EXECUTION TIME (IN SECONDS) OF TEST ITERATIONS. ALL TESTS PRESENTED IN THE TABLE WERE PERFORMED ON THE SAME MACHINE.

Training data set	<i>SMALL<sub>B</sub></i>	<i>LARGE<sub>A</sub></i>	<i>LARGE<sub>B</sub></i>
accounts	11 401	19 261	21 270
transactions	383 463	2 854 965	2 625 671
Test data set	<i>SMALL<sub>A</sub></i>	<i>LARGE<sub>B</sub></i>	<i>LARGE<sub>A</sub></i>
accounts	11 238	21 270	19 261
transactions	294 972	2 625 671	2 854 965
Iteration 1 time	1 108	2 110	2 204
Iteration 2 time	1 191	2 088	2 119
Iteration 3 time	1 197	2 027	2 067
Iteration 4 time	1 126	2 146	2 098
Iteration 5 time	1 162	2 109	2 173
Iteration 6 time	1 156	2 116	2 165
Iteration 7 time	1 189	2 060	2 451
Iteration 8 time	1 166	2 130	2 141
Iteration 9 time	1 185	2 128	2 409
Iteration 10 time	1 172	2 075	2 096
Average time	1 165	2 099	2 192

in fact, legal. Therefore, the percentage of "not classified" transactions among transactions marked as suspicious can be interpreted as a false positive ratio. In real-life scenario, however, some of the "not classified" transactions contained in the training data set will actually be illegal (i.e. involved in a money laundering schemes) because it is never possible to identify all illegal activities. Nevertheless, statistically, most of the "not classified" transactions are legal. It is thus hard to decide at this point, to what extent the learning model should be trained to avoid reporting transactions similar to "not classified" examples from the training set.

Further work may concern improving the precision of the detection (in order to avoid too many legal transactions being submitted to human experts for evaluation) but also improving the completeness of the results (ensuring that as many illegal transactions as possible are marked as suspicious). Also, improvement in computational speed may be important, especially because computation time may be a factor limiting the possibility of searching for more complex subgraph patterns.

In order to achieve the above mentioned goals further work may be conducted in some of the following directions:

- training a population of positive and negative patterns. Currently, specimens represent only such patterns that identify suspicious subgraphs. Learning patterns that represent legal transaction subgraphs may help in reducing the false positive ratio;
- using decision rules designed by experts to improve detection quality. Apart from knowledge extracted from annotated graphs, rules, for example excluding tax transfers from suspicion, may help in reducing the search space that must be processed;
- interacting with human expert, for example using the number of transactions confirmed by the expert to be

illegal as a criterion in model training. Another possibility is to allow the expert who uses the system to adjust some parameters by hand. For example, parameters controlling fuzzy number mutation ( $\Delta_x$ ,  $m_x$  and  $M_x$ ) may be tuned in order to ensure that the model obtained during training satisfies certain constraints;

- using Genetic Multi-Objective Optimization (GMOO) in order to balance conflicting requirements (maximizing the number of detected frauds, but at the same time limiting the number of transactions suggested for review by human expert);
- implementing computationally-intensive parts of the algorithm (for example, evaluation function calculation) using CUDA architecture in order to take advantage of massive parallelism available on modern GPUs;
- allowing more complex subgraph patterns to be used. One of the possible approaches can be to implement evolving graph structure (not only evolutionary optimization of subgraph model parameters);
- alternative methods for searching for matching subgraphs. Currently all matching subgraphs are evaluated at each level of the hierarchical pattern. Using non-deterministic methods may be hard to avoid, especially if more complex subgraph patterns will be used.

Another important aspect of further research is to obtain real-life data from financial institutions and test the proposed method on such data.

#### REFERENCES

- [1] Th. Fetz, J. Jager, D. Koll, G. Krenn, H. Lessmann, M. Oberguggenberger and R. Starkm "Fuzzy Models in Geotechnical Engineering and Construction Management" *Computer-Aided Civil and Infrastructure Engineering*, vol. 14, no. 2, pp. 93–106, 1999.
- [2] O. Hasancebi and F. Erbatır, "Evaluation of crossover techniques in genetic algorithm based optimum structural design" *Computers & Structures*, vol. 78, no. 1-3, pp. 435–448, 2000.
- [3] J. Korczak, W. Marchelski and B. Oleszkiewicz, "A New Technological Approach to Money Laundering Discovery using Analytical SQL server," in: *Advanced Information Technologies for Management AITM 2008*, J. Korczak, H. Dudycz and M. Dyczkowski (eds.) *Research Papers no. 35*, pp. 80-104, Wrocław University of Economics, 2008.
- [4] J. Korczak, B. Oleszkiewicz, "Modelling of Data Warehouse Dimensions for AML Systems," in: *Advanced Information Technologies for Management AITM 2009*, J. Korczak, H. Dudycz and M. Dyczkowski (eds.) *Research Papers no. 85*, pp. 146-159, Wrocław University of Economics, 2009.
- [5] E. M. Truman and P. Reuter, "Chasing Dirty Money: The Fight Against Anti-Money Laundering", Peterson Institute for International Economics, 2004
- [6] J. Zhong, X. Hu, J. Zhang and M. Gu, "Comparison of Performance between Different Selection Strategies on Simple Genetic Algorithms," in: *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce Vol-2 (CIMCA-IAWTIC'06) - Volume 02*, pp. 1115–1121, IEEE Computer Society, 2005.
- [7] "Debit and Credit Card fraud requires vigilance" <http://www.moneyreview.com.au/debit-and-credit-card-fraud-requires-vigilance/>
- [8] "Reserve Bank of Australia - Payments Data" <http://www.rba.gov.au/payments-system/resources/statistics/index.html>
- [9] "Russia reports \$3.8 trillion in suspect transfers: report," <http://www.reuters.com/article/2010/12/06/us-russia-economy-money-idUSTRE6B516Z20101206>



# Growing Hierarchical Self-Organizing Map for searching documents using visual content

Paweł B. Myszkowski  
Applied Informatics Institute,  
Wrocław University of Technology  
Wyb. Wyspiańskiego 27, 51-370 Wrocław, POLAND  
email: pawel.myszkowski@pwr.wroc.pl

Bartłomiej Buczek  
email: bartlomiej.buczek@op.pl

**Abstract**—This paper presents document search model based on its visual content. There is used hierarchical clustering algorithm - GHSOM. Description of proposed model is given as learning and searching phase. Also some experiments are described on benchmark image sets (e.g. ICPR, MIRFlickr) and created document set. Paper presents some experiments connected with document measures and their influence on searching results. Also in this paper some first results are given and directions of further research are given.

## I. INTRODUCTION

THE document search task is connected to large dataset of documents which can be used in many applications e.g. in libraries to find a relevant document similar to one given in a query. Such task can be solved as classification task of data mining domain in knowledge discovery database process. However, classification needs labeling of all documents and this is nearly impossible because of time/cost constraints connected to labeling by human. Thus our work concerns on clustering task in unsupervised learning mode, where structure of data is not given or we barely know anything about it. Also, we decided to use a special type of clustering – hierarchical clustering to build a hierarchy of documents. It gives us a very useful advantage in navigation of document search space to find similar documents navigating between included images that can be connected and further in searching for similar documents basing of its visual aspects. However, our task is specific, where the document is analyzed only in the visual context. So basically our model bases only on visual content of document, not text. The visual content of document, in this stage of research, means only included images, figures, tables and schemas. Our work is alternative to semantic text based document analyses and indeed our approach results would be linked to such method in the complex system.

There are many methods that bases on hierarchical clustering of data for previously artificially created hierarchy by human. There are also some methods, like Growing Hierarchical Self-Organizing Map (*GHSOM*) which creates hierarchy from scratch.

Description of methods and architectures based on similar images search or image category search, clustering or clusterisation can be found in work [13]. Another survey [5] presents various approaches to document layout description, document/images features selection, classification and application. Models are hidden markov model, neural network, k-Nearest Neighbor or rule based approaches.

This work presents our first research results for quality of documents search using *GHSOM*. The 2nd section shows ideas of organizing data as a map and *GHSOM* as extension of *SOM*. Section 3rd presents proposed approach, architecture and connected processes. Some experiments are included in section 4th, it also shows results of our first research. Last section concludes and describes directions of further research.

## II. GROWING HIERARCHICAL SELF-ORGANIZING MAP

Organizing data was firstly proposed in 1982 by Kohonen to explain the spatial organization of the brain's functions. The data is presented as neural network with the aid of adaptation of weights vectors becomes organized [10]. The Self-Organizing Map (*SOM*) is a computational, unsupervised tool to the visualization and analysis of high-dimensional data. There are plenty applications where *SOM* is used, especially in text mining [10].

A useful way to organize data presents Mäkelä [3]. His method can be described as Hierarchical Self-Organizing Map (*HSOM*) which relays on series of *SOMs* which are placed in layers where the number of *SOMs* in each structure layer depends on number of *SOMs* in previous layers (the upper one). In this model number of structure layers in *HSOM* and dimension of maps are defined *a priori*.

In the learning process (which always starts from top layers to the bottom layer) units weight vectors in *SOMs* of each layer are adapted. The adapt of weight in next layer is only possible when given layer is finished. Final effect of this process is hierarchical clustering of data set. This approach has some advantages:

- smaller number of connections between input and units in *HSOM* layers [3],

This work is partially financed from the Ministry of Science and Higher Education Republic of Poland resources in 2010-2013 years as a research SYNAT project (System Nauki i Techniki) in INFINITI-PASSIM.

- much shorter processing time which comes from the point above and from hierarchical structure of learning process [3].

In *HSOM* there are some necessities:

- definition of maps sizes and number of layers which depends on data set,
- choosing of learning parameters for each layer in *HSOM*.

There is another *SOM* extension that reduces some above disadvantages. Growing Self-Organizing Map (*GSOM*) is another variant of *SOM* [1] to solve the map size issue. The model consists of number of units which in learning process grows. The learning process begins with minimal number of units and grows (by adding new units) on the boundary based on a heuristic. To control the growth of the *GSOM* there is special value called Spread Factor (*SF*) [1] - a factor that controls the size of growth. At the beginning of learning process all units are boundary units which means that each unit has the freedom to grow in its own direction. If the unit is chosen to grow it grows in all its free neighboring positions.

However in *HSOM* still exists a problem of *a priori* given architecture. Another approach, Growing Hierarchical Self-Organizing Map (*GHSOM*) solves it.

The architecture of *GHSOM* allows to grow in both a hierarchical and in a horizontal direction [2]. This provides a convenient structure of clustering for large data sets which can be simple navigate through all layers from the top to the bottom.

The learning process begins with a virtual map which consists of one vector initialized as average of all input data [15]. Also for this unit, error (distance) between its weight vector and all data is calculated - this error is global stop parameter in *GHSOM* learning process. Then next layer map (usually initialized by 2x2 [8]) is created below the virtual layer (this newly created *SOM* indeed is a child for unit in virtual layer). From now on, each map grows in size to represent a collection of data at a specified level of detail. The main difference between growing in *GHSOM* from *GSOM* is that the whole row or column of units is added to currently learning *SOM* layer. The algorithm can be shortly described as:

1. For present *SOM* start learning process and finish it after  $\lambda$ -iterations.

2. For each unit count error (distance) between its weight vector  $m_i$  and input patterns  $x(t)$  mapped onto this unit in the *SOM* learning process (this error is called quantization error  $qe$ ).

3. If the sum of quantization errors is greater or equal than certain fraction of quantization error of parent unit:

$$\sum qe_i \geq \tau_1 * qe_{parent_{unit}}$$

3a. If yes - select unit with biggest error and find neighbor to this unit which is most dissimilar. Go to step 4.

3b. else - stop.

4. Between these two units put row/columnn.

5. Reset learning parameter and neighbor function for next *SOM* learning process.  
Go to step 1.

When the growth process of layer is finished the units of this map are examined for hierarchical expansion [11]. Basically, units with large quantization error will add a new *SOM* for the next layer in the *GHSOM* structure according to  $\tau_1$  value. More specifically, when quantization error of unit in examined map is greater than fraction  $\tau_2$  of global stop parameter, then a new *SOM* (usually initialized by 2x2) is added to structure. The learning process and unit insertion now continue with the newly established *SOMs*. The difference in learning process of new layer is that fraction of data set is used for training corresponding to units mapped into parent unit [2].

The whole learning process of the *GHSOM* is terminated when no more units are required for further expansion. This learning process does not necessarily lead to a balanced hierarchy (the hierarchy with equal depth in each branch) [15]. To summarize, the growth process of the *GHSOM* is guided by two parameters  $\tau_1$  and  $\tau_2$ . The parameter  $\tau_1$  controls the growth process in layer (certain fraction in algorithm for expanding in layer for *GHSOM*) and the parameter  $\tau_2$  controls hierarchical growth of *GHSOM*.

The clustering of images is needed because, when today's search engine is asked about images usually as an answer comes images which were named by the asked phrase. It would be better if the answer come on the base of what image contains. The content of images can be described by a vector of features - vector of numbers. *GHSOM*, in simply way, bases on vectors of real numbers so images can be successfully used as a data set. In this work the idea was to create a *GHSOM* structure which will hierarchically organize images in groups of similar images. The visual aspects of such clustering (visualization of images in cluster) can be analyzed by human or by some quality measures and ratings.

### III. PROPOSED *GHSOM* BASED MODEL

Model base on *GHSOM* for searching similar documents in database includes mainly 3 modules: document preprocessing (for visual element extraction and description), *GH-SOM* structure and document of indexed documents. The schema of proposed model is given on Fig.1.

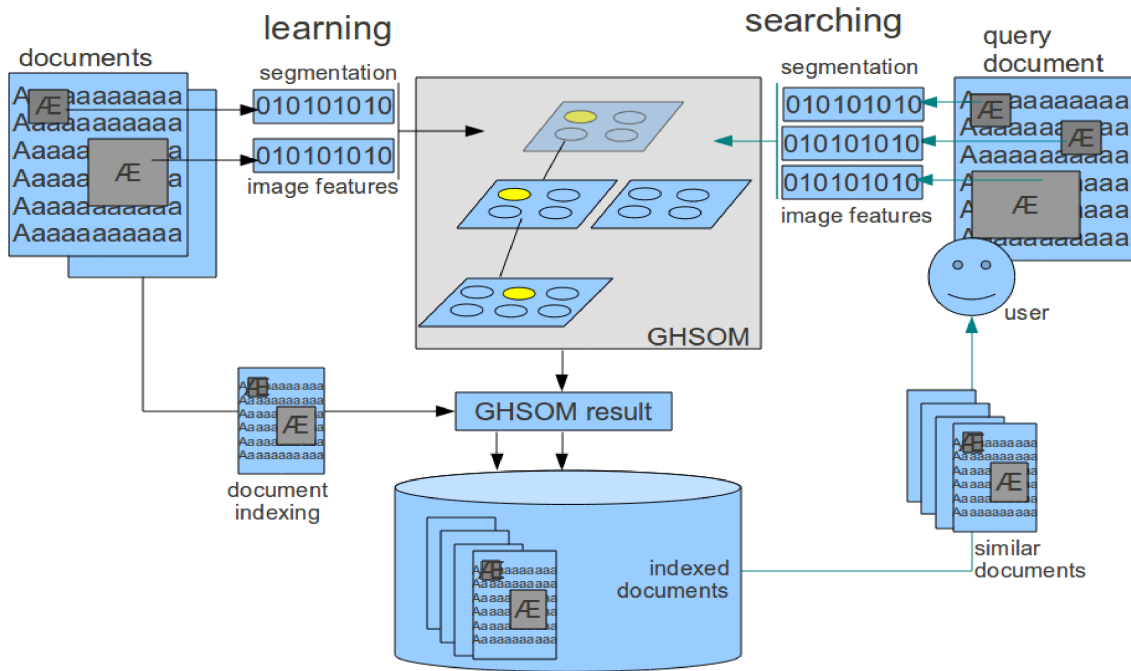


Fig. 1 Proposed document searching model

### A. Learning phase

The first stage of such process is preprocessing. Each document is analyzed by getting its visual content: figures, tables, schemas and others. Each visual element is divided in 25 segments (5x5 grid); for each segment image features are calculated (e.g. feature connected to image color, structure or shapes). Each visual element is represented as a sequence of vectors (=segments) and set of visual elements gives the whole representation of document.

Next stage of learning phase uses the hierarchical clustering in *GHSOM* model to organize space of set of documents. Each visual element is presented to *GHSOM* to build hierarchical cluster that would be used in search phase.

In the next step *GHSOM* learning procedure runs and gives hierarchical clustering structure for document indexing. The document is indexed and then registered into base of all indexed documents as vector of output *nodes* from *GHSOM* structure. The length of vector equals to the total number of *node* units, and each node unit has the own identifier and place in output document vector  $\Phi(\text{document}_i)$ . If its activated value is not zero, e.g. if in *i*-th document the 2nd node has been activated once and 6th has been activated twice output document vector is:

$$\Phi(\text{document}_i) := \langle 0,0,1,0,0,0,2,0 \rangle.$$

Such representation is intuitive and allows to use document similarity or metrics in simple way (very crucial aspect in searching phase). On the end of learning proces each already indexed document is registered in database, that works the special role in searching phase.

### B. Searching phase

The already learned *GHSOM* can be used for searching similar documents in database of all indexed documents. We assume that the best way of searching is the document query

*document<sub>query</sub>*. The query document has to be converted by preprocessing procedure (finding visual elements, its segmentation and features calculations) and it allows to present to *GHSOM* structure the document representation. Already calculated the output of activated *GHSOM* units gives  $\Phi(\text{document}_{\text{query}})$  and it becomes the base of similar documents searching in database. According to used document measure (we experimented mainly with euclidean and cosine measures) it returns the sequence of *n* most similar documents in database.

### C. Model parameters discussion

The proposed model parametrization is not a complex problem. The more difficult issue is selection of visual images features. In this stage of research we used only color RGB based features, but we plan to extend this representation by adding extra features connected to shape and texture. The *GHSOM* algorithm uses only two basic values for steering learning process  $\tau_1$  and  $\tau_2$ . Another parameter is connected with database and searching process, where document measure must be given. We experimented with euclidean and canonical cosine measures of documents representation and it gives very promising results.

## IV. EXPERIMENTS AND FIRST RESULTS

We developed proposed model as programistic platform written in Java for experiments. Our experiments are divided into separate tasks: similar image search and similar document search. The motivation of the first task is connected with series of experiments to basic *GHSOM* model quality verification. Results of second types of experiments give information about effectiveness of the whole model.



### A. Similar image searching

The experiments of image hierarchical clustering using GHSOM were based on benchmark images dataset: ICPR<sup>1</sup>, IAPR TC12<sup>2</sup> and MIRFlickr<sup>3</sup>. These datasets consist of labeled images that make possible not only image clusterisation but also verification of image distribution in GHSOM hierarchical structure. Our research was concerned on:

- external measure of *GHSOM*: image query to model and verification according to connected with image and *GHSOM* unit keywords,
- internal measure of *GHSOM* as the set of image clusters – we used Davies Bouldin Index (*DBI*) and Dunn Index (*DI*)

Our experiments, detailed in description [4], show that *GHSOM* is a useful tool of image hierarchical clustering. Although we used only RGB color image features, results are very successful. The example of proper *GHSOM* clustering image distribution is presented in Fig. 2.



Fig. 2 Example of GHSOM images clustering – 'buildings' in benchmark ICPR benchmark image dataset

The *GHSOM* application to similar images searching task gives also some hints about further research. There are some badly created cluster and badly classified images (but the percent of them are respectively small). In the future, visual aspects of images in hierarchical clusters created by *GHSOM* can be researched. As well, in this work not only classical Euclidean distance was used (but also L1 and Tchybyszew's measure) but another experiments with different measures for distance should be applied.

<sup>1</sup>ICPR dataset: <http://www.cs.washington.edu/research/>

<sup>2</sup>IAPR TC12 dataset: <http://www.imageclef.org/photodata/>

<sup>3</sup>MIRFlickr dataset: <http://www.flickr.com/>

### B. Similar documents searching

The main task of GHSOM in proposed model is similar document searching. We created a dataset 1kPDF that consist of 1000 PDF documents. Each document is in polish language and has at least one visual element. Documents are not labeled but only grouped into 100 or 200 domain groups: sport, motorization, general, architecture and art. Each group is created by other person to make the whole document set more independent. Thus documents in various domains can be alike.

We developed on experiments to examine the search quality of model using given document measure. 25 selected randomly from various groups: 4 documents are connected with architecture, 8 motorization documents, 8 general, 4 sport and 1 concerns art. Search results for cosine document measure for these 25 documents are presented in Table I. The 'general' domain also has high effectiveness (33%). It means that these documents are more relevant in 10 answer documents.

The worst result is observed in art (5%) and sport (2,7%) documents. For art document query situation seems to be problematic in interpretation as there is chosen only one doc-

TABLE I.  
RESULTS OF FIRST EXPERIMENTS – COSINE DOCUMENT MEASURE

Q \ A	architect	motor.	general	sport	art
architect.	<b>25,00%</b>	33,75%	32,50%	6,25%	2,50%
motor.	31,87%	<b>26,87%</b>	35,00%	2,50%	3,75%
general	35,00%	25,64%	<b>33,12%</b>	1,25%	5,00%
sport	22,97%	27,02%	36,48%	<b>2,70%</b>	10,80%
art	45,00%	30,00%	15,00%	5,00%	<b>5,00%</b>

ument connected with art domain and result cannot be representative. Also, not clear situation occurs for sport, where only 5% of relevant answers were given. However, selected documents connected to sport are very specific – they are short and often consist of one image. Search result returns on first place similar documents (in sport domain) but some answered documents are connected to others domains.

The same set of 25 documents was used to examine Euclidean document measure (see Table II). Such measure seems to be more effective in search similar documents. The

TABLE II.  
RESULTS OF FIRST EXPERIMENTS – EUCLIDEAN DOCUMENT MEASURE

Q \ A	architect	motor.	general	sport	art
architect	<b>21,25%</b>	40,00%	18,75%	6,25%	13,75%
motor.	5,62%	<b>63,12%</b>	11,25%	7,50%	12,50%
general	9,37%	45,62%	<b>21,25%</b>	8,12%	15,62%
sport	0,00%	66,25%	7,50%	<b>11,20%</b>	15,00%
art	0,00%	35,00%	20,00%	10,00%	<b>35,00%</b>

best result was for 'motor' domain – 63%, the worst for sport group (11,2% valid, and 66% answer is motorcycle).

The example query for 'motor' document is presented on Table III, where 9 first most similar documents of 20 are relevant to questioned document, which makes answer very accurate.

TABLE III.  
EXAMPLE RESULTS OF 'MOTORYZACJA/620' DOCUMENT QUERY DOCUMENT  
USING EUCLIDEAN DOCUMENT MEASURE

<b>motoryzacja/656; 17.34935</b>	
<b>motoryzacja/628; 17.52141</b>	
<b>motoryzacja/1293; 17.7200</b>	...
<b>motoryzacja/613; 18.41195</b>	sztuka/1376; 20.09975124
<b>motoryzacja/1269; 18.4661</b>	arch-bud/529; 20.12461179
<b>motoryzacja/1290; 19.39071</b>	<b>motoryzacja/627; 20.17424</b>
<b>motoryzacja/1288; 19.51922</b>	<b>motoryzacja/1291; 20.1990</b>
<b>motoryzacja/645; 19.748417</b>	arch-bud/572; 20.2731349
<b>motoryzacja/665; 19.899748</b>	arch-bud/569; 20.29778313
arch-bud/552; 20.02498439	ogolne/1085; 20.3224014
ogolne/1115; 20.07485989	ogolne/1060; 20.3469899
...	ogolne/1058; 20.37154878

Our experiments show that model gives very promising results, however it difficult to say which measure is better. Also, results are better than it seems because groups of documents are alike and some documents can be labeled as member of two groups e.g. 'architecture' documents can be included to 'art' group too.

## V. SUMMARY AND FURTHER RESEARCH

This paper presents results of our first experiments with *GHSOM* based document searching model. Our approaches use hierarchical clustering of image content of collected documents to search for similar document. Paper presents results of experiments connected to different document measure (Euclidean and classic Cosine one). Thus there is no the best document measure and in conclusion is that there is strong need of extra experiments series. The important factor of future research is the image feature selection. In this stage only *rbg* based features were selected while there are useful features like texture and shape.

Presented series of experiments are initiative and extra experiments are planned for document collections of 10 000 documents (results given in paper bases of 1000 document collection). The larger collection may give the experiments more precision and generalization, which also allows to do various experiments.

## REFERENCES

- [1] Alahakoon, D., Halgamuge, S. K., Sirinivasan, B.: A Self Growing Cluster Development Approach to Data Mining. Proc. of IEEE Inter. Conf. on Systems, Man and Cybernetics (1998)
- [2] Bizzil, S., Harrison, R.F., Lerner, D. N.: The Growing Hierarchical Self-Organizing Map (GHSOM) for analysing multi-dimensional stream habitat datasets. 18th World IMACS / MODSIM Congress (2009)
- [3] Blackmore, J., Mikkulainen, R.: Incremental grid growing: Encoding high-dimensional structure into a two-dimensional feature map. Proc. of the IEEE Inter. Conf. on Neural Networks (1993).
- [4] Buczek B.M., Myszkowski P.B.: Growing Hierarchical Self-Organizing Map for Images Hierarchical Clustering, ICCCI proc. Conferences (in printing), LNCS 2011.
- [5] Nawei Chen, Dorothea Blostein.: A survey of document image classification: problem statement, classifier architecture and performance evaluation, IJDAR 10, pp.1–16, (2007)
- [6] Chih-Hsiang, C., Chung-Hong, L., Hsin-Chang, Y.: Automatic Image Annotation Using GHSOM. Fourth Inter. Conf. on Innovative Comp., Infor. and Control (2009)
- [7] Fritzsche, B.: Some Competitive Learning Methods. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.47.3885>
- [8] Halkidi, M., Batistakis, Y., Vazirgiannis, M.: Clustering validity check-ing methods: part II. ACM SIGMOD Record Vol. 31 (3) (2002)
- [9] Herbert, J. P., JingTao Yao: Growing Hierarchical Self-Organizing Maps for Web Mining. Proc. of the 2007 IEEE/WIC/ACM Inter. Confer. e on Web Intel. (2007)
- [10] Huiskes, M. J., Lew, M. S.: The MIR Flickr Retrieval Evaluation. ACM Inter. Conf. on Multimedia Inf. Retrieval (2008)
- [11] Kohonen, T.: Self-organizing maps, Springer-Verlag, Berlin (1995)
- [12] Rauber, A., Merkl, D., Dittenbach, M.: The GHSOM: Exploratory Analysis of High-Dimensional Data. IEEE Trans. on Neural Networks(2002)
- [13] Ritendra Datta, Dhiraj Joshi, Jia Li, James Z. Wang.: Image Retrieval: Ideas, Influences, and Trends of the New Age, ACM Computing Surveys, Vol. 40, No. 2, Article 5 (2008).
- [14] Raza A., Usman G., Aasim S.: Data Clustering and Its Applications. [http://members.tripod.com/asim\\_saeed/paper.htm](http://members.tripod.com/asim_saeed/paper.htm)
- [15] Vicente, D., Vellido, A.: Review of Hierarchical Models for Data Clustering and Visualization. [In] R.Giraldez et al. (Eds.) Tendencias de la Minería de Datos en España. España ola de Minería de Datos (2004)
- [16] Hierarchical clustering. [http://www.aiaccess.net/English/Glossaries/GlosMod/e\\_gm\\_hierarchical\\_clustering.htm](http://www.aiaccess.net/English/Glossaries/GlosMod/e_gm_hierarchical_clustering.htm)
- [17] Experiments with GHSOM. <http://www.ifs.tuwien.ac.at/~andi/ghsom/experiments.html>





# Automatic Image Annotation by Image Fragment Matching

Mariusz Paradowski  
 Institute of Informatics

Wroclaw University of Technology, Poland  
 Email: mariusz.paradowski@pwr.wroc.pl

Andrzej Śluzek  
 Khalifa University, Abu Dhabi  
 United Arab Emirates

**Abstract**—Automatic image annotation problem is one of the most important sub-problems of computer vision. It is strongly related to and goes beyond image recognition. The key goal of annotation is to assign a set of words from a given dictionary to a previously unseen image. In this paper, we address two key problems related to image annotation. The first one is low precision of generated answers, the second one is automatic localization of image fragments related to annotations. The proposed method utilizes image fragment matching to precisely localize near-duplicate visual content of images.

## I. INTRODUCTION

**A**UTOMATIC image annotation (AIA) is an extension of image recognition. The input for an AIA method is an image. The output is a set of words (also referred as classes), from a given dictionary, which describe the input image in a best possible way. However, in the AIA problem additional assumptions are made [1], comparing to the classic recognition problem [2]. The major assumption is the lack of relations between the words and their visual representation (feature vectors) within a single image. Each word within the image description is represented by all feature vectors, which can be generated from the visual content of the image. Instead of a one to one relations (like in a classic recognition), there are many to many relations.

Most of the available image annotation methods outputs a set of words, without any information on the localization of the detected words. Precise localization of image fragments related to detected words is a challenging task. The earliest approaches to AIA assumed image segmentation. Each segment has been related with one or more words, e.g. [3], [4]. Such approach fails in most cases, because image segmentation usually do not covers true boundaries of objects represented by the dictionary. More recent research shows that image segmentation used for the purpose of annotation do not have to recreate true object boundaries, e.g. [5]. Rectangular blocks [5], [8] or various setups of fixed regions [6] became very popular and effective. Obtained quality results are usually better than those with classic segmentation, but word localization is practically impossible. There are attempts to handle AIA with word localization without image segmentation, e.g. [7]. A very dense grid of rectangles is utilized. In general, all mentioned approaches give some rough, very imprecise information on the localization of word representations on the image.

Another problem is a low precision of practically all available image annotation approaches. Even the state-of-the-art methods (e.g. [8]) tested on relatively simple image databases fail to reach precision above 90%. There are many sources of this problem, however one of the most prominent is the mentioned lack of relations between words and image fragments. Construction of high quality recognition models is not possible, due to very high noise in the training data. Only a small fragment of positive training data is correct [7].

Recent research in AIA (e.g. [8]) shows that distance based methods, strongly related to image retrieval, give very promising results. Key-point based, near-duplicate retrieval is particularly interesting. Key-points such as *SIFT* [9] are used to model local photometric properties of images. These fragments are matched together to establish candidate relations between images. Global geometric [10], [11] or topological [12] constraints provide necessary information to verify the correctness of candidates.

The paper presents a new automatic image annotation method, strongly related to key-point based image retrieval. It solves mentioned problems for a specific sub-domain of images and words. Generated annotations reach high precision. The method it is able to precisely locate words on the image, without such information in the training set. The proposed approach has limited applicability, it is only able to annotate words with near-duplicate visual representation.

## II. PROPOSED APPROACH

The proposed method is called *Automatic Image Annotation by Matching* (AIAM). Let us formalize important concepts at the beginning. We follow with a general description and all the necessary details.

### A. Formal definitions

Let us give the formal definitions of important concepts. First, we focus on automatic image annotation. Later on, near-duplicate fragment detection is described.

a) *Automatic image annotation*: The dictionary  $\mathcal{W} = \{w_1, w_2, \dots, w_k\}$  is a set of words  $w_x : x = \{1, \dots, k\}$ , where  $k$  is the size of the dictionary. Words are strings, they do not have any related semantics. The training set  $\mathcal{I}$  consists of  $n$  pairs: images  $I_x$  and their annotations  $W_x \subseteq \mathcal{W}$ , where  $x = \{1, \dots, n\}$ :

$$\mathcal{I} = \{(I_1, W_1), (I_2, W_2), \dots, (I_n, W_n)\}. \quad (1)$$

Input image  $J$  is an image to be described in an automatic manner. Automatic image annotation method  $\mathcal{A}$  describes image  $J$  using words from the dictionary  $\mathcal{W}$ , based on the data from the training set  $\mathcal{I}$ . Output of the method is an annotation  $\mathcal{A}(J) = W_J \subseteq \mathcal{W}$  of the input image  $J$ .

b) *Near-duplicate image fragments*: Near-duplicate image fragment matching method  $\mathcal{M}$  accepts a pair of images  $(X, Y)$  as an input. If these two images contain near-duplicate image fragments, they are marked using outlines (convex hulls). Each pair of outlines represent a single near-duplicate image fragment. The output is a set of  $m$  outline pairs. Both the number of outline pairs ( $m$ ) and the outlines are detected fully automatically by the matching method  $\mathcal{M}(X, Y)$ :

$$\mathcal{M}(X, Y) = \{(f_X^1, f_Y^1), (f_X^2, f_Y^2), \dots, (f_X^m, f_Y^m)\}. \quad (2)$$

### B. Outline of the method

The proposed automatic image annotator utilizes near-duplicate fragment matching method as a key component. Matching of near-duplicates has to be performed with possible highest quality. Out of several available approaches to image fragment matching, a method proposed by the authors is chosen [13]. The method utilizes low level image features (*SIFT* [9]) and affine geometry. A six dimensional probability density function of available affine transformations is constructed. The density function is modelled using a non-parametric approach (sparse histogram built using hash-table), which allows simultaneous matching of unknown number of image fragments.

The largest advantage of this matching method is high precision. There are only a few *false positive* errors. *False negatives* are much frequent ones, but lack of detection is a problem of much less grade. Additionally, the applied method generates highly precise outlines of detected near-duplicates.

In the first step of the method input image  $J$  is matched with all images  $I_x : x = \{1, \dots, n\}$  from the training set  $\mathcal{I}$ . Output of a single matching is a set of outlines representing near-duplicates. Exemplary near-duplicates for a selected input image  $J$  are shown in Fig. 1.

Generated set of near-duplicates may not be directly used to propagate words from image annotations. There are no relations between image fragments and words. Presence of near-duplicates is only a prerequisite to word propagation. First, it has to be determined which words are to be propagated. Second, matching may be erroneous, thus it has to be verified.

1) *Propagation of words*: For each image  $I_x$  from the training set  $\mathcal{I}$  we have to automatically determine additional information required for word propagation. A set of images  $S_x^w \subset \mathcal{I}$  is called a *supporting set* for word  $w \in W_x$  and image  $I_x$ . Supporting set  $S_x^w$  contains images having exactly one common word ( $w$ ) with annotation  $W_x$  of image  $I_x$ . Formally, supporting set  $S_x^w$  for image  $I_x$  and word  $w \in W_x$  is defined as:



Fig. 1. Examples of matched near-duplicates for input image  $J$  and various images from the training set  $\mathcal{I}$ .

$$S_x^w = \bigcup_{\substack{1 \leq y \leq n \\ y \neq x}} \{I_y\} : (W_y \cap W_x = \{w\}). \quad (3)$$

Supporting set  $S_x^w$  for a given image  $I_x$  and word  $w \in W_x$  allows to:

- propagate word  $w$  from the annotation  $W_x$  with higher precision,
- relate word  $w$  with matched image fragments generated by  $\mathcal{M}(J, I_x)$ ,
- verify the correctness of image fragments matching  $\mathcal{M}(J, I_x)$ .

In case the fragment matching of two images ( $J$  and  $I_x$ ) is not empty ( $\mathcal{M}(J, I_x) \neq \emptyset$ ), propagation is performed for all words that annotate image  $I_x$ . Input image  $J$  is matched with all images from the supporting sets ( $\mathcal{M}(J, s) : s \in S_x^w$ ). If a word  $w \in W_x$  has supporting matches ( $\mathcal{M}(J, s) \neq \emptyset$ ), it is verified if matching outlines intersect each other. To get the most credible answer (we assume that matching precision is high, but recall may be low), the largest intersection is selected:

$$(f_J^1, f_{I_x}^1), (g_J^1, g_s^1) = \operatorname{argmax}_{\substack{(f_J, f_{I_x}) \in \mathcal{M}(J, I_x) \\ (g_J, g_s) \in \mathcal{M}(J, s)}} |f_J \cap g_J|. \quad (4)$$

Near-duplicate fragments generated by method  $\mathcal{M}$  are considered as valid, if and only if the intersection of outlines is large enough, i.e. it meets the following criterion (*first intersection test*):

$$\left( \frac{|f_J^1 \cap g_J^1|}{|f_J^1|} > t \right) \cap \left( \frac{|f_J^1 \cap g_J^1|}{|g_J^1|} > t \right), \quad (5)$$

where:  $t \in (0, 1)$  is the method parameter. This concept is presented in Fig. 2.

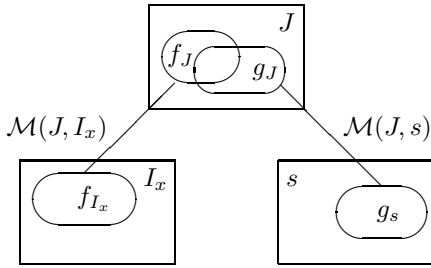


Fig. 2. Basic propagation of words with the usage of supporting sets. There are two different near-duplicate sets on image  $J$ , the first one comes from  $\mathcal{M}(J, I_x)$ , the second one from  $\mathcal{M}(J, s)$ . Outline intersections on image  $J$  have to match as close as possible.

Basic propagation of words with the proposed intersection tests eliminates a large number of *false positives*. However, some of them are still accepted as correct matches.

2) *Extended propagation of words*: Let us now propose another variant of word propagation. To get even higher precision of annotations, we introduce an extended propagation method. It is also based on the image fragment matching and the sets of supporting images  $S_x^w$ . It encapsulates the *first intersection test* used in basic propagation (eq. 5).

The extended propagation also utilizes a triple of images:  $J$ ,  $I_x$  and a supporting image  $s$ . As already mentioned, the basic propagation may not be sufficient to prevent *false positives*. Given the results of matching  $\mathcal{M}(I_x, s)$ , it is possible to define two additional tests of propagation correctness. They allow for even better rejection of incorrect matches. In case  $\mathcal{M}(I_x, s) \neq \emptyset$ , largest intersections of outlines within images  $I_x$  and  $s$  are found. The first one is done on image  $I_x$ :

$$(f_J^2, f_{I_x}^2), (h_{I_x}^2, h_s^2) = \operatorname{argmax}_{\substack{(f_J, f_{I_x}) \in \mathcal{M}(J, I_x) \\ (h_{I_x}, h_s) \in \mathcal{M}(I_x, s)}} |f_{I_x} \cap h_{I_x}|, \quad (6)$$

The second one is done on image  $s$  (from the supporting set  $S_x^w$ ):

$$(g_J^3, g_s^3), (h_{I_x}^3, h_s^3) = \operatorname{argmax}_{\substack{(g_J, g_s) \in \mathcal{M}(J, s) \\ (h_{I_x}, h_s) \in \mathcal{M}(I_x, s)}} |g_s \cap h_s|. \quad (7)$$

Having the largest intersections of outlines, we propose two additional tests to validate the correctness of  $\mathcal{M}(J, I_x)$  matching. For each test a threshold  $t \in (0, 1)$  is given (the same as in basic propagation, see eq. 5). The first test examines the intersection of outlines in image  $I_x$ :

$$\left( \frac{|f_{I_x}^2 \cap h_{I_x}^2|}{|f_{I_x}^2|} > t \right) \cap \left( \frac{|f_{I_x}^2 \cap h_{I_x}^2|}{|h_{I_x}^2|} > t \right), \quad (8)$$

The second test examines the intersection of outlines in image  $s$ :

$$\left( \frac{|g_s^3 \cap h_s^3|}{|g_s^3|} > t \right) \cap \left( \frac{|g_s^3 \cap h_s^3|}{|h_s^3|} > t \right). \quad (9)$$

The idea is presented in Fig. 3.

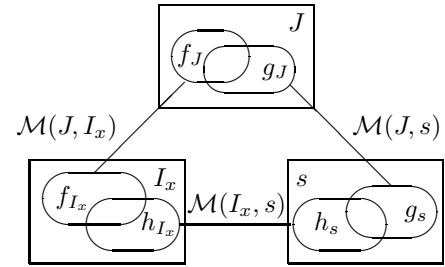


Fig. 3. Extended word propagation routine. Three outline intersection tests are performed. The first one is done on image  $J$  (matches  $\mathcal{M}(J, I_x)$  and  $\mathcal{M}(J, s)$ ), the second one is done on image  $I_x$  (matches  $\mathcal{M}(J, I_x)$  and  $\mathcal{M}(I_x, s)$ ) and the last one on image  $s$  (matches  $\mathcal{M}(J, s)$  and  $\mathcal{M}(I_x, s)$ ).

Experimental results presented in the later part of the paper shown that the extended propagation routine is fully sufficient. Less precise matching routines may require additional tests to reject false matches. In case such tests are necessary, they may be constructed using the same idea, as the above.

### C. Localization of annotated objects

One of our goals is to precisely localize fragments of images to which generated annotation should be related. It is possible due to the application of image fragment matching routine. Presented word propagation mechanism automatically relates words with some image fragments.

The input image  $J$  is automatically annotated with word  $w \in \mathcal{W}$ . The word could be propagated from any image  $I_x \in \mathcal{I}$  annotated by this word ( $w \in W_x$ ). To maximize recall we should propagate outlines from all matched images. However, such solution is not a precise and valid one.

Let us assume that three (or more) images from the training set contain a visual representation of word  $w$  on a near-duplicate background. Image fragment matching method  $\mathcal{M}$  works on a purely visual manner, and thus does not know anything about visual representation of word  $w$  or any neighboring background. The main near-duplicate match  $\mathcal{M}(J, I_x)$  and the supporting matches  $\mathcal{M}(J, s)$  and  $\mathcal{M}(I_x, s)$  are going to generate outlines containing both the proper image fragment

representing word  $w$  and the neighboring background. Intersection tests (see eqs. 5, 8 and 9) are also going to accept both the object and the background.

However, the intersection tests accept only near-duplicates of a similar relative fragment size (up to a threshold). If  $\mathcal{M}(J, I_x)$  returns both the objects of interest and the background, only a small subset (images containing both the object and the background) of supporting set will support this match. Thus it is worth rejecting near-duplicate matches  $\mathcal{M}(J, I_x)$  supported by small subsets of their supporting sets. We propose to select a single image  $I_x$ , for which the matched subset of the supporting set is the largest. In other words, to get the most credible match  $I_x$ , it is expected that the match is supported by as many other matches as possible.

#### D. The dictionary and the training set

The proposed automatic image annotation approach does have three noticeable flaws. They are related to the training set and the dictionary. We are going to describe them in detail.

First, the method annotates words that have near-duplicate visual representation. The proposed annotator utilizes highly precise, key-point based, image fragment matching method. The matching method is only able to detect near-duplicates, all other types of visual similarity are not taken into account. Words such as *sky*, *cloud*, etc. are not going to be detected (recall will be near zero or zero). The proposed method is applicable to words representing mainly *rigid bodies*, similar in shape and appearance. Surprisingly, most of automatic image annotation methods usually fail to correctly annotate such words. They focus on the mentioned holistic concepts, which can be easily generalized. This method does the opposite, it does not generalize. It annotates words with complex, but highly repeatable appearance.

Second, effective annotation of a word requires a sufficient representation in the training set. Propagation of a word requires an existence of the supporting set (see II-B1). The larger the supporting set is, the easier it is to propagate annotations. Used near-duplicate detection method has very high precision, but lower recall. It rarely makes *false positive* errors, but often causes *false negatives*. False negatives may be eliminated by providing a larger set of examples, e.g. captured under various lighting conditions, relative position to camera, etc. To propagate a word, it is sufficient to have only three correct matches. Larger word representations make finding such triples much easier.

Third, background repetition in the training set should be minimized. Near-duplicate detection is purely visual, i.e. it does not differentiate between objects of interest and background. If the same background goes together with object of interest in most images, it is also considered a valid part of image. Generated annotations are going to be correct. However, localization of these annotations on the image will not depict the true and expected boundaries. We consider this flaw the most prominent one, because it may require manual modification of the training set.

### III. EXPERIMENTAL VERIFICATION

The goal of experimental verification is to check the expected properties of the proposed method. According to the initial assumptions, the main expectation is a high precision of generated annotations. Additionally, annotated objects have to be precisely localized.

Presented experimental verification consists of four parts:

- presentation of exemplary results of the method,
- setup of method parameters,
- analysis of annotation quality,
- analysis of object localization quality.

The image set consists of 100 images, together with annotations and outlines for each annotated word<sup>1</sup>. Single images contain multiple objects of interest, thus the set is well suited for AIA. The dictionary is constructed from all available annotations. All performed tests are done using *leave one out* experimental protocol. Typical, state-of-the-art automatic image annotation methods fail to get high precision results.

#### A. Exemplary results

According to the definition of automatic image annotation, we have a database of annotated images. Annotations are related with entire images, there are no relations between words and image fragments. Exemplary elements from the training set are presented in Fig. 4.

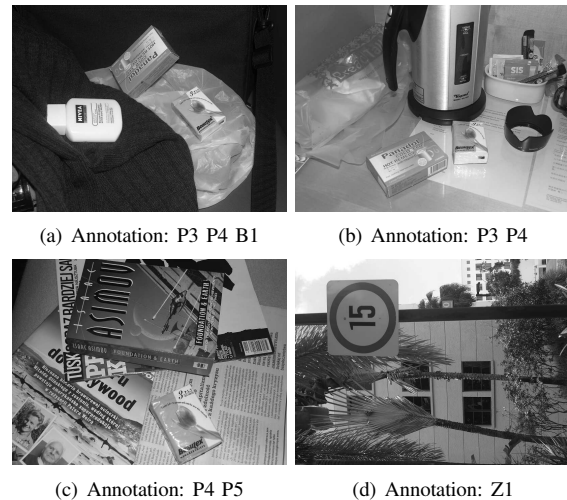


Fig. 4. Exemplary described images from the training set. According to the definition of automatic image annotation problem, annotations are related with entire images, instead of image fragments.

Fig. 5 presents exemplary results generated by the proposed method. The method generates image annotations and outlines of image fragments representing single words. Annotation outlines are generated by the near-duplicate image fragment detection method. They are a combination of at least two (usually much more) near-duplicate outlines. Of course, the more precise the near-duplicate outlines are, the better are the final results.

<sup>1</sup><http://www.ii.pwr.wroc.pl/~visible>

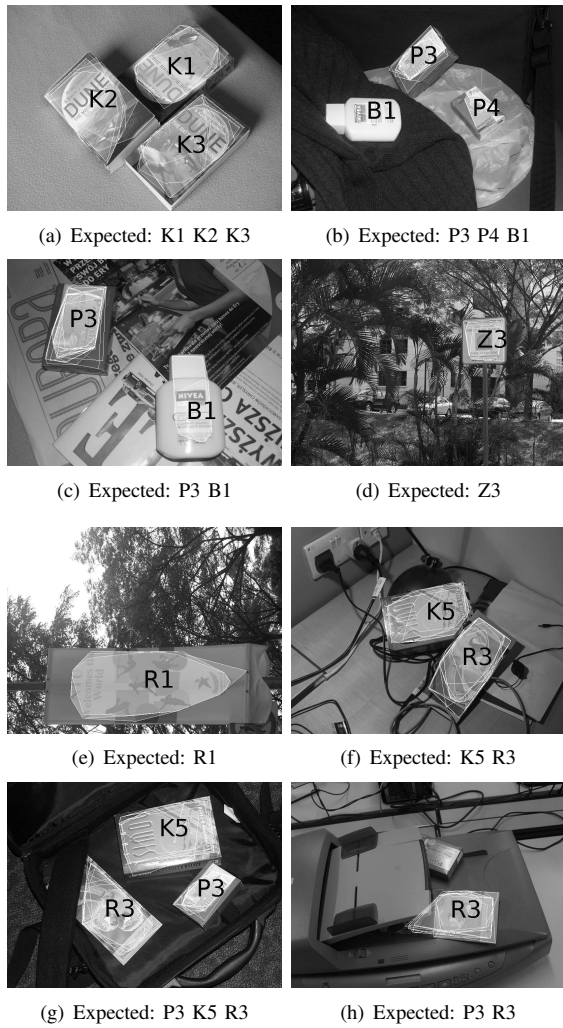


Fig. 5. Generated annotations together with localization of found words. Expected words are shown below the images. It should be noted, that the training set does not contain any information on words localization. It is determined by the proposed method fully automatically.

The automatic image annotator uses geometric approach to image fragment matching. The geometric approach tends to generate outlines smaller than the true boundaries of objects. Due to usage of key-points (*SIFT*) as the elementary visual data, it is very difficult to recreate affine transformations on object boundaries. Simply, there are not enough key-points on the boundaries (usually there are very few). Key-points outside the true boundaries may not be taken into consideration, because they do not generate valid affine transformations. Mentioned behaviour may be observed in Fig. 5.

### B. Setup of method parameters

The second part of experimental verification addresses the parameter setup. Two standard annotation quality measures are used: *recall* and *precision*. Unlike in automatic image annotation, these quality measures may be calculated in two different manners. The first one is the classic approach, based

TABLE I  
QUALITY OF RESULTS FOR VARIOUS SETUP OF METHOD PARAMETERS. EXTENDED PROPAGATION PROVIDES HIGHER PRECISION COMPARING TO BASIC PROPAGATION.

<i>thres. t</i>	<i>prec. [obj.]</i>	<i>recall [obj.]</i>	<i>prec. [area]</i>	<i>recall [area]</i>
Basic propagation				
0.00	0.92	0.81	0.86	0.43
0.25	0.95	0.80	0.89	0.43
0.33	0.96	0.80	0.89	0.43
0.50	0.96	0.80	0.90	0.43
0.66	0.96	0.80	0.89	0.43
0.75	0.96	0.78	0.89	0.43
Extended propagation				
0.00	0.99	0.69	0.95	0.40
0.25	1.00	0.68	0.96	0.38
0.33	1.00	0.67	0.95	0.39
0.50	1.00	0.66	0.96	0.40
0.66	1.00	0.62	0.96	0.37
0.75	1.00	0.58	0.97	0.36

on the presence or absence of words in annotations. The second one is based on the quality of generated outlines. Outline pixels on the image may be considered as false positives, false negatives and true positives (true negatives are irrelevant) and used in complex measures: *recall* and *precision*.

Two variants of the proposed method are taken into consideration: basic propagation (see Sec. II-B1) and extended propagation (see Sec. II-B2). Results achieved for the basic propagation reach precision near 95% and recall near 80%. However, the key goal is to get as high precision as possible. Given the extended propagation, precision is equal to 100%. Recall is lowered to 68% because higher requirements of the propagation routine. Summary of the method quality is presented in Tab. I.

Cutoff threshold  $t$  is the second parameter of the method. The threshold is used in the supporting set acceptance tests (eqs. 5, 8 and 9). The larger the threshold, the more similar size of the detected fragments is required. According to the performed tests,  $t = 0.25$  is sufficient. However, given the need of background matches rejection (see Sec. II-D), suggested threshold value is higher and equal to  $t = 0.50$ .

### C. Quality of generated annotations

The third part of the experimental verification is the analysis of annotation quality. Table II contains detailed results calculated for each word from the dictionary.

Taking into account the initial requirements, the most interesting column is the *false positives* one. No *false positives* are found during our experiments with the extended propagation. Proposed extended propagation routine with three built-in validation tests is sufficient to eliminate *false positive* matches. The quality of generated outlines plays the key role. Better outlines cause larger outline intersections with smaller relative size changes. Better intersections are a better confirmation of near-duplicate matches, and a better confirmation of generated annotations.

The second interesting observation is related to *false negatives*. Words with the lowest recall (mostly 0) are the least

TABLE II

QUALITY OF GENERATED ANNOTATIONS. ONLY THE PRESENCE OF WORDS IS TAKEN INTO ACCOUNT. LOCALIZATION OF WORDS IS NOT TAKEN INTO ACCOUNT.

class	TP	FP	FN	prec.	recall
B1	7	0	0	1.00	1.00
B2	0	0	5	–	0.00
B3	0	0	3	–	0.00
H1	3	0	1	1.00	0.75
H2	0	0	2	–	0.00
HD	0	0	2	–	0.00
IP	0	0	2	–	0.00
K1	12	0	0	1.00	1.00
K2	3	0	4	1.00	0.42
K3	7	0	0	1.00	1.00
K4	0	0	3	–	0.00
K5	7	0	0	1.00	1.00
P1	7	0	1	1.00	0.87
P2	0	0	2	–	0.00
P3	20	0	1	1.00	0.95
P4	2	0	5	1.00	0.28
P5	5	0	1	1.00	0.83
R1	4	0	0	1.00	1.00
R2	0	0	2	–	0.00
R3	15	0	0	1.00	1.00
T1	0	0	3	–	0.00
T2	0	0	3	–	0.00
Z1	2	0	3	1.00	0.40
Z2	4	0	0	1.00	1.00
Z3	5	0	0	1.00	1.00
Z4	0	0	4	–	0.00
Z5	0	0	2	–	0.00
Z6	0	0	2	–	0.00
Z7	0	0	2	–	0.00
all	103	0	53	1.00	0.66

frequent ones in the training set. This confirms the second mentioned flaw of the proposed annotator (see Sec. II-D). In case of more frequent words, recall grows up. A sufficiently large number of diversified training examples makes near-duplicate matching possible.

#### D. Quality of annotated words localization

The last part of the presented experimental verification is assessment of object localization quality. The assessment requires additional information, *not available* in the training set, i.e. outlines of all annotated objects. This allows to verify the quality of localization up to single pixels.

The quality measurement is performed using *precision* and *recall* based on outline intersections. Detailed results are shown in Tab. III. Precision of results is slightly lower than 100%. It is directly related to the image fragment matching method: outlines of generated fragments are in sometimes larger than the true boundaries of objects. Annotation outline is a sum of several matching outlines, i.e. it contains all false positives of these outlines. One of possible solutions is to use intersection of all outlines, however this causes a very large drop of recall.

#### IV. SUMMARY

A new method of automatic image annotation is presented. The method is called *Automatic Image Annotation by Matching* (AIAM). To annotate a previously unseen image, word

TABLE III

QUALITY OF GENERATED OUTLINES. QUALITY MEASURES ARE CALCULATED USING EXPECTED AND GENERATED OUTLINE INTERSECTIONS.

class	prec.	recall	class	prec.	recall
B1	0.91	0.35	B2	–	0.00
B3	–	0.00	H1	0.99	0.64
H2	–	0.00	HD	–	0.00
IP	–	0.00	K1	0.99	0.86
K2	0.98	0.30	K3	0.91	0.85
K4	–	0.00	K5	0.94	0.91
P1	0.90	0.48	P2	–	0.00
P3	0.97	0.78	P4	0.99	0.09
P5	0.98	0.68	R1	0.98	0.63
R2	–	0.00	R3	0.99	0.85
T1	–	0.00	T2	–	0.00
Z1	1.00	0.04	Z2	1.00	0.17
Z3	0.99	0.74	Z4	–	0.00
Z5	–	0.00	Z6	–	0.00
Z7	–	0.00			
all				0.96	0.49

propagation is used. Words are propagated between images sharing near-duplicate visual content. Near-duplicates are detected using low level key-points (*SIFT*) and global affine geometry.

Two routines of near-duplicate image matching verification are proposed: basic and extended one. Triples of images sharing common visual content are found. Each accepted triple is a sufficient premise to propagate a single word from the training set image into the final annotation.

For a sub-domain of images (with the presence of near-duplicates) the method is able to get very high precision (up to 100%). Recall achieved for the test image set reaches 68%. Another interesting feature of the proposed annotator, is the ability to localize fragments of image representing annotated words. The major feature and the limitation of the proposed approach is the limitation to near-duplicates. Words with very similar visual appearance may be used in the dictionary. All other words will be automatically discarded, because near-duplicate matches will not be found.

Further research will cover two main areas. The first one is the speed-up of matching routines, while keeping high quality of outlines (partially done already). The second one will focus on increase of recall, within the near-duplicate detection framework. Once there two goal are reached, we expect to get a much more viable annotator.

#### REFERENCES

- [1] P. Duygulu, K. Barnard, N. de Freitas and David Forsyth, Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary, Proceedings of Seventh European Conference on Computer Vision (ECCV'02), vol. 4, pp. 97-112, 2002.
- [2] M. Kurzynski, Objects Recognition: statistical methods (in Polish), Wroclaw University of Technology Publishers, 1997.
- [3] J. Jeon, V. Lavrenko, R. Manmatha, Automatic Image Annotation and Retrieval using Cross-Media Relevance Models, Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 119-126, 2003.
- [4] V. Lavrenko, R. Manmatha, J. Jeon, A Model for Learning the Semantics of Pictures, Proceedings of NIPS, MIT Press, 2003.

- [5] S. L. Feng, R. Manmatha, V. Lavrenko, Multiple Bernoulli relevance models for image and video annotation, Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04), Vol. 2, pp. 1002-1009, 2004.
- [6] B. Shah, R. Benton, Z.H. Wu and V. Raghavan, Automatic and Semi-Automatic Techniques for Image Annotation, Semantic Based Visual Information Retrieval, pp. 112-134, 2007.
- [7] G. Carneiro and N. Vasconcelos, Formulating Semantic Image Annotation as a Supervised Learning Problem, Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2, pp. 163-168, 2005.
- [8] M. Stanek, B. Broda and H. Kwaśnicka, PATSI – Photo Annotation through Finding Similar Images with Multivariate Gaussian Models, Proceedings of the ICCVG (2), pp. 284-291, 2010.
- [9] D. G. Lowe, Object recognition from local scale-invariant features, Proc. 7th IEEE Int. Conf. Computer Vision, Vol. 2, pp. 1150–1157, 1999.
- [10] H. Jégou, M. Douze and C. Schmid, Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search, Proceedings of the 10th European Conference on Computer Vision, vol. I, pp. 304-317, 2008.
- [11] D. Yang and A. Śluzek, A low-dimensional local descriptor incorporating TPS warping for image matching, Image and Vision Computing, vol. 28(8), pp. 1184-1195, 2010.
- [12] C. Schmid and R. Mohr, Object recognition using local characterization and semi-local constraints, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19(5), pp. 530-534, 1997.
- [13] M. Paradowski and A. Śluzek, Local Keypoints and Global Affine Geometry: Triangles and Ellipses for Image Fragment Matching, Innovations in Intelligent Image Analysis, SCI 339, pp. 195-224, 2010.





# How to predict future in a world of antibody-antigen chromosomes

Zbigniew Pliszka and Olgierd Unold, *Member, IEEE*

**Abstract**—The paper deals with a representation of the antibody-antigen chromosomes. The proposed new binary decoding allows us to prove the dependence between subsequent generations of chromosomes, using quick and simple operations on chromosomes indices, instead of processing the binary strings. Some formal properties of the immune system were expressed based on this representation. A consistency theoretical proof for epistatic properties as well as exploration possibilities of a crossover operator was given.

**Index Terms**—Genetic algorithm, Crossover, Binary coding, Hadamard representation, Artificial Immune System.

## I. INTRODUCTION

**A**RTIFICIAL Immune Systems (AISs) constitute currently a significant trend in the studies on biologically inspired calculations [2], [14]. Much work on AIS has concentrated on simple extraction of biological metaphors and direct application. There is very limited work on the more theoretical aspects of AIS, especially on the formal proofs of AIS algorithms [13]. A complete proof for a specific multiobjective clonal selection algorithm using Markov chains has been given in [15]. In [1] Markov chain model of the B-cell algorithm has been developed to show a convergence proof, and also a mathematical model of the mutation operator.

Work in theoretical immunology has developed various representations for the interactions between antibody and antigen, and affinity metrics for modeling these such interactions. These antibody-antigen binding models were proposed for describing antibody cross-reactivity [5].

In this paper we attempt to model binary space, which includes both antigens and antibodies, and try to theoretically predict their further development based on knowledge of the first (initial) population. Antibody-antigen chromosomes are represented as binary, fixed-length chromosomes, using an alternative to zero-one decoding technique, called Hadamard representation.

In addition to above mentioned aim, the present paper seeks to address potential capabilities of a crossover operator. There is much criticism of the role of the crossover in evolutionary algorithm (EA) literature. Several authors [11], [3], [4] have pointed out that the crossover causes the premature convergence of the EA, i.e. the EA loses population diversity before some goal is met. Some of them argue that for problems of nontrivial size and difficulty, the contribution of crossover

search is marginal [8]. Spears [10] claims that mutation is more powerful than crossover in terms of exploration, although the two operators can be treated as two forms of a searching operator. Experiments conducted by Schaffer and Eshelman [12] have been indicated that population of chromosomes manipulated by crossover contains epistatic interaction.

A consistency theoretical proof for epistatic properties as well as exploration possibilities of a crossover operator is given in this paper.

This paper extends the results of our previous study [9] by (1) using different notation that leads to a shift of the indices, (2) defining a crossover operator in  $H^n$  space, (3) introducing new property of the immune system called *Expansive system with global range*, (4) proving formal properties of a crossover operator (**Theorem 1**), (5) and supporting the states of AIS by redefined examples. In addition, some remarks on **Schema Theorem** used in Hadamard space are made.

This paper is organized as follows. Section 2 describes binary representation used in the research, next Section 3 gives some formal properties of populations in AIS. Section 4 formulates a concept of an ancestral population. Finally Section 5 concludes the paper with future works.

## II. BINARY REPRESENTATION

### A. Hadamard model

In the study [9] the search space  $\{0, 1\}^n$  was replaced by  $\{-1, 1\}^n$  (so called a Hadamard representation [6]). Thanks to use a new binary model the requirement of orthogonal columns pairs is omitted. Subject of this study is the following set:

$$H^n = \{(h_{s,n}, h_{s,n-1}, \dots, h_{s,2}, h_{s,1}) : \forall s \in \{0, 1, \dots, 2^n - 1\} \\ \forall i \in \{1, 2, \dots, n\} \quad h_{s,i} \in \{-1, 1\}\} \quad (1)$$

Its elements represent all possible binary chromosomes of equal length  $n$ , where  $n$  is a natural number higher than 1. The proposed representation has one, apparently insignificant property, which distinguishes it from the binary representation: a square of each coordinates is equaled 1. This fact draws two subsequent conclusions: the sum of the squares of coordinate of each element of the  $H^n$  space is constant and equals this space dimension, and there is no element with zero coordinates. The collection of these simple facts allows for the formulation of rules for phenotypes (indices) and development of automate methods of moving frame  $H^n$ , as well as determination of the distance (level of differentiation) between the elements of this space.

Z. Pliszka is with Wroclaw Public Library, Sztabowa 95, 53-310 Wroclaw, Poland.

O. Unold is with Institute of Computer Engineering, Control and Robotics, Wroclaw University of Technology, Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland, e-mail: olgierd.unold@pwr.wroc.pl.

At the beginning, we determined the order of the indexing of points in  $H^n$  and their four representations, which we will use alternating (see Table I).

TABLE I  
INDEXING AND REPRESENTATION OF POINTS IN  $H^n$  SPACE.

Element's symbol	Decimal representation	Binary representation	Hadamard representation
$r_0$	0	(0,0,...,0,0,0)	( 1, 1,..., 1, 1, 1)
$r_1$	1	(0,0,...,0,0,1)	( 1, 1,..., 1, 1,-1)
$r_2$	2	(0,0,...,0,1,0)	( 1, 1,..., 1,-1, 1)
$r_3$	3	(0,0,...,0,1,1)	( 1, 1,..., 1,-1, -1)
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r_{2^n-3}$	$2^n-3$	(1,1,...,1, 0, 1)	(-1,-1,...,-1, 1,-1)
$r_{2^n-2}$	$2^n-2$	(1,1,...,1, 1, 0)	(-1,-1,...,-1,-1, 1)
$r_{2^n-1}$	$2^n-1$	(1,1,...,1, 1, 1)	(-1,-1,...,-1,-1,-1)

The number of points included in  $H^n$  is equal to  $|H^n| = 2^n$ . For each element in the binary representation there are numerous functions transforming the elements of this representation to the elements of the Hadamard representation and inversely (see [9]).

Index  $s$  of the element  $r_s$  having a binary representation  $(b_{s,n}, \dots, b_{s,1})$  and the Hadamard representation  $(h_{s,n}, \dots, h_{s,1})$ , and equal to the value of the decimal representation, can be calculated from one of the formulas:

$$ID(r_s) = s = \sum_{t=1}^n 2^{t-1} b_{s,t} = \sum_{t=1}^n 2^{t-2} (1 - h_{s,t}) \quad (2)$$

In cases of doubt, we will use the function symbol  $ID(r_s)$ , otherwise the sign of the index  $s$  will be used.

### B. The distance in $H^n$

The distance of two points  $r_t = (h_{t,n}, h_{t,n-1}, \dots, h_{t,2}, h_{t,1})$  and  $r_k = (h_{k,n}, h_{k,n-1}, \dots, h_{k,2}, h_{k,1})$  in  $H^n$  space is measured according to the following equation:

$$\forall r_t, r_k \in H^n \quad w(r_t, r_k) = n - \frac{1}{4} \sum_{i=1}^n (h_{t,i} + h_{k,i})^2 \quad (3)$$

The distance defined in that way will always be a nonnegative integer, which will tell us on how many coordinates in Hadamard representation the values differ (exactly as in the binary representation) (see Table I). In addition,  $H^n$  space with the  $w$  distance determined in this way is metric and has many interesting properties (proved in [9]). For example, there is an algorithmic method for construction a table of distances between any elements from the  $H^n$  space (without calculating the distance between elements). Bearing in mind that the space  $H^n$  contains  $2^n$  elements, and the space  $H^{n+1}$  contains  $2^{n+1} = 2 \cdot 2^n$ , the distance table  $W_{n+1}$  of dimension  $2^{n+1} \times 2^{n+1}$  is to be obtained from the table  $W_n$  of dimension  $2^n \times 2^n$  according to the following formula:

$$\langle W_n \rangle \mapsto \left\langle \begin{array}{cc} W_n & \langle W_{n+1} \rangle \\ \langle W_{n+1} \rangle^T & W_n \end{array} \right\rangle,$$

where +1 means addition to each element of the table a value of 1, and  $W_n^T$  the transpose of a matrix  $W_n$ . The number situated on the cross of the  $k$ -row with the  $t$ -column corresponds to  $w(r_k, r_t)$  distance. Exceptionally, for the purposes of this algorithm, we assume that  $W_0 = \langle 0 \rangle$ , because, as already mentioned, the spaces  $H^n$  where  $n > 1$  are considered in this study. The distance table for  $n = 4$  is presented in Table II.

In the work [9] the definition of the polar points was introduced. Two points  $r_t$  and  $r_k$  in  $H^n$  are called polar points if and only if for each coordinate these points have opposite values.

$$\forall j \in \{1, \dots, n\} \quad h_{t,j} = -h_{k,j}$$

According to the formula (3), the distance between polar points is constant and equals  $w(r_t, r_k) = n$ . Some other properties of polar points were given in [9].

### C. Crossover in $H^n$

Any two points from  $H^n$  can be provided in accordance with the definition (1) as:

$$\begin{aligned} r_t &= (r_{t,n}, r_{t,n-1}, \dots, r_{t,2}, r_{t,1}) \\ r_k &= (r_{k,n}, r_{k,n-1}, \dots, r_{k,2}, r_{k,1}) \end{aligned}$$

Assuming that  $c$  is the position of the cutting operation in a crossover counted from the right, fulfilling the inequality  $0 \leq c \leq n$ , we define the operation of the crossing-over an element  $r_t$  with an element  $r_k$  after the locus  $c$  as follows (note that crossover operator corresponds to 1-point-crossover):

$$\begin{aligned} K(\{r_t, r_k\}, c) \mapsto \\ \{ &(r_{t,n}, r_{t,n-1}, \dots, r_{t,c+1}, r_{k,c}, r_{k,c-1}, \dots, r_{k,1}), \\ &(r_{k,n}, r_{k,n-1}, \dots, r_{k,c+1}, r_{t,c}, r_{t,c-1}, \dots, r_{t,1}) \} \end{aligned}$$

The newly received elements belong to the  $H^n$  and can be symbolically represented:

$$r_s = (r_{t,n}, r_{t,n-1}, \dots, r_{t,c+1}, r_{k,c}, r_{k,c-1}, \dots, r_{k,1}) \quad (4)$$

$$r_q = (r_{k,n}, r_{k,n-1}, \dots, r_{k,c+1}, r_{t,c}, r_{t,c-1}, \dots, r_{t,1}) \quad (5)$$

Now, the operation of the crossover can be written as

$$K(\{r_t, r_k\}, c) \mapsto \{r_s, r_q\}, \quad (6)$$

where the indices  $s$  and  $q$  can be taken from the formulas ([9]):

$$s = t - (t \bmod 2^c) + (k \bmod 2^c) \quad (7)$$

$$q = k - (k \bmod 2^c) + (t \bmod 2^c) \quad (8)$$

## III. PROPERTIES OF ARTIFICIAL IMMUNE SYSTEM IN $H^n$ SPACE

This section depicts some properties of AIS, in which antibody-antigen chromosomes are represented using Hadamard encoding. Introduced properties are illustrated by the examples based on the content of Table II.

TABLE II  
THE DISTANCE TABLE BETWEEN ANY ELEMENTS IN  $H^4$ .

	$r_0$	$r_1$	$r_2$	$r_3$	$r_4$	$r_5$	$r_6$	$r_7$	$r_8$	$r_9$	$r_{10}$	$r_{11}$	$r_{12}$	$r_{13}$	$r_{14}$	$r_{15}$
$r_0$	0	1	1	2	1	2	2	3	1	2	2	3	2	3	3	4
$r_1$	1	0	2	1	2	1	3	2	2	1	3	2	3	2	4	3
$r_2$	1	2	0	1	2	3	1	2	2	3	1	2	3	4	2	3
$r_3$	2	1	1	0	3	2	2	1	3	2	2	1	4	3	3	2
$r_4$	1	2	2	3	0	1	1	2	2	3	3	4	1	2	2	3
$r_5$	2	1	3	2	1	0	2	1	3	2	4	3	2	1	3	2
$r_6$	2	3	1	2	1	2	0	1	3	4	2	3	2	3	1	2
$r_7$	3	2	2	1	2	1	1	0	4	3	3	2	3	2	2	1
$r_8$	1	2	2	3	2	3	3	4	0	1	1	2	1	2	2	3
$r_9$	2	1	3	2	3	2	4	3	1	0	2	1	2	1	3	2
$r_{10}$	2	3	1	2	3	4	2	3	1	2	0	1	2	3	1	2
$r_{11}$	3	2	2	1	4	3	3	2	2	1	1	0	3	2	2	1
$r_{12}$	2	3	3	4	1	2	2	3	1	2	2	3	0	1	1	2
$r_{13}$	3	2	4	3	2	1	3	2	2	1	3	2	1	0	2	1
$r_{14}$	3	4	2	3	2	3	1	2	2	3	1	2	1	2	0	1
$r_{15}$	4	3	3	2	3	2	2	1	3	2	2	1	2	1	1	0

### A. Radius of tolerance

A radius of tolerance  $R$  is understood as the border value enabling a mutual recognition of elements in  $H^n$  space.

Two elements  $x, y \in H^n$  recognize or do not tolerate each other if the distance between them is higher than the radius of tolerance.

$$w(x, y) > R \quad (9)$$

where  $R$  complies with the inequality:  $0 \leq R \leq n$ .

Elements  $x, y \in H^n$  complying the weak inequality

$$w(x, y) \leq R \quad (10)$$

will be described as not recognizing or tolerating each other.

#### Example 0

In the examples considered here we use the  $H^4$  space, whose distance tables are presented in Table II. Moreover, for all the demonstrated examples we assume the value of the radius of tolerance  $R = 2$ .

### B. Self-aggression

System  $B_k \subseteq H^n$  undergoes self-aggression if elements  $x, y$  occur, which recognize each other and belong to this system.

$$\exists x, y \in B_k : w(x, y) > R \quad (11)$$

#### Example 1

In  $H^4$  the systems undergoing self-aggression are for example:

$$B_8 = \{r_0, r_1, r_2, r_4, r_8, r_3, r_5, r_6\} \text{ where } w(r_1, r_6) = 3$$

$$B_4 = \{r_3, r_5, r_6, r_9\} \text{ where } w(r_6, r_9) = 4$$

System  $B_k \subseteq H^n$  is free of self-aggression if any two elements belonging to this system do not recognize themselves.

$$\forall x, y \in B_k \quad w(x, y) \leq R \quad (12)$$

#### Example 2

Free systems of self-aggression, when  $R = 2$ :

$$B_5 = \{r_0, r_1, r_2, r_4, r_8\}$$

$$B_3 = \{r_3, r_5, r_6\}$$

$$B_2 = \{r_0, r_1\}$$

Let us notice that system  $B_2$  is free of self-aggression also when  $R = 1$ .

Systems undergoing self-aggression have elements (chromosomes) dispersed in the space under consideration, systems free of self-aggression are centered around a certain element (chromosome) and we have a suspicion that this is a local extremum for many used objective functions.

### C. Dazzling distance set

A dazzling distance set of a system  $B_k \subseteq H^n$  is a set of points of  $H^n$  recognized by any point of  $B_k$ .

$$P(B_k) = \{z \in H^n : \exists x \in B_k \wedge w(x, z) > R\} \quad (13)$$

#### Example 3

For  $B_3 = \{r_3, r_5, r_6\}$  from the **Example 2** the dazzling distance set is:

$$P(B_3) = \{r_1, r_2, r_4, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}\}$$

If  $B_k$  undergoes self-aggression then some points belonging to  $B_k$  simultaneously belong to  $P(B_k)$ , which means that

$$B_k \cap P(B_k) \neq \emptyset$$

because, according to (11) there is a pair of points  $x, y \in B_k$  that  $w(x, y) > R$  and  $y \in H^n$ , what gives (13).

#### Example 4

The system  $B_4 = \{r_3, r_5, r_6, r_9\}$  from the **Example 1** is in such a state:

$$P(B_4) = \{r_1, r_2, r_4, r_6, r_7, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}\}$$

and we have:

$$B_4 \cap P(B_4) = \{r_6, r_9\} \neq \emptyset$$

Otherwise, if  $B_k$  is free of self-aggression, then  $B_k$  and  $P(B_k)$  are disjunctive sets, which can be presented as follows:

$$B_k \cap P(B_k) = \emptyset \quad (14)$$

since, according to (12), any two elements  $x, y \in B_k$  satisfy the inequality  $w(x, y) \leq R$ , which contradicts (13).

**Example 5**

$B_3$  (described in **Example 2** and **3**) is free of self-aggression, for which identity occurs:

$$B_3 \cap P(B_3) = \emptyset$$

**D. Complete system**

System  $B_k$  is complete if its dazzling distance set contains its whole completion  $\overline{B_k} = H^n \setminus B_k$ .

$$\overline{B_k} \subseteq P(B_k) \quad (15)$$

**Example 6**

The conditions of the complete system are fulfilled by  $B_7 = \{r_0, r_1, r_2, r_3, r_4\}$ , for which following identities occur:

$$P(B_7) = \{r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}\}$$

$$\overline{B_7} = \{r_5, r_6, r_7, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}\},$$

Thus, a relation occurs:

$$\overline{B_7} \subseteq P(B_7)$$

The statement, that we are dealing with a complete system, gives us confidence that in many problems we control the entire space under consideration, using only a certain part of the elements (chromosomes) of that space. In many issues, the important task to deal with is to set the minimum complete systems (i.e. which contain the smallest number of elements) for a given space.

**E. Balanced system**

System  $B_k$  is balanced if at the same time it is a system free of self-aggression, and complete. System  $B_k$  satisfies the equality

$$\overline{B_k} = P(B_k)$$

The relationship  $\overline{B_k} \subseteq P(B_k)$  we have from the definition of a complete system (see (15)). Inverse relationship  $\overline{B_k} \supseteq P(B_k)$ , we get as a request from two facts:  $P(B_k) \subseteq H^n = B_k \cup \overline{B_k}$ , as well as  $B_k$  as a system free of self-aggression satisfies (14), and therefore  $\overline{B_k} \supseteq P(B_k)$ .

**Example 7**

This time let us assume that  $B_5 = \{r_0, r_1, r_2, r_4, r_8\}$  ( $B_5$  is taken from the Example 2). For such a system the following identities are fulfilled:

$$P(B_5) = \{r_3, r_5, r_6, r_7, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}\}$$

$$\overline{B_5} = \{r_3, r_5, r_6, r_7, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}\}$$

and

$$\overline{B_5} = P(B_5)$$

**F. Extensive system**

We call  $B_k \subseteq H^n$  an extensive system if each crossing-over of its elements results in offspring, which also belongs to this system.

$$\forall x, y \in B_k \subseteq H^n \quad \forall c \in \{0, 1, \dots, n\} \quad K(\{x, y\}, c) \subseteq B_k$$

**Example 8**

The examples of the extensive systems are presented below:

$$B_2 = \{r_0, r_1\}$$

$$B_4 = \{r_0, r_1, r_2, r_3\}$$

$$H^4$$

$$B_1 = \{r_0\}$$

To check the extensibility of  $B_2$  and  $B_4$  systems, equations (7) and (8) can be used. It can be noticed that both the singleton system and any  $H^n$  space, as a whole, are extensive systems.

**G. Expansive system**

A system  $B_k$  is expansive if it possesses elements (not necessary different), which after a crossing-over produce elements out of the system.

$$\exists x, y \in B_k \subseteq H^n \quad \exists c \in \{0, 1, \dots, n\} : K(\{x, y\}, c) \notin B_k$$

**Example 9**

Let us assume  $B_9 = \{r_0, r_1, r_2\} \subseteq H^4$ . Assuming in equations (6) (7) and (8) fore  $c = 1, t = 1, k = 2$  we have:

$$K(\{r_1, r_2\}, 1) \mapsto \{r_0, r_3\} \notin B_9$$

due to the element  $r_3 \notin B_9$ .

**H. Expansive system with global range**

We say that the expansive system  $B_k \subseteq H^n$  is global in range, if each element from the completion of  $B_k$  can be obtained only as a result of crossing-over of its elements and offsprings.

**Example 10**

Note that the set of elements  $B_{11} = \{r_0, r_{15}\}$  is the expansive system with global range. These two elements are sufficient to be the ancestors of all space  $H^n$ . And more generally, any system consisting of two polar points, is a expansive system with global range (proof to be found below).

**Definition 1**

An *initial* or *primary population* is a set of chromosomes (elements) from the  $H^n$  space, which receives an evolutionary process (or program) input. We assume that all elements of such a set take part in the first selection process for the parent pool.

**Definition 2**

We say that the population is *ancestral*, if all its elements can be obtained from a primary population as a result of the assembling only crossing-overs.

Note that this definition does not reject the elements created by the assembly of other operations (for example, mutation

or inversion) on the initial population, but it requires the existence of the potential emergence of such elements by submitting only the crossing-overs of the initial population and its posterity. In this case, we also have the ancestral population.

#### IV. ANCESTRAL POPULATION

##### Theorem 1

The whole space  $H^n$  is the ancestral population if and only if there are the elements in the primary population  $P$ , which have the following properties:

for each position (locus), we have two elements from  $P$  having different (in terms of dual opposing) values.

##### Proof

$\Rightarrow$  (proof by contradiction)

Suppose that for the initial population  $P \subseteq H^n$  there is the locus  $l \in \{1, \dots, n\}$ , with the property that all elements from  $P$  have one value (in the case of binary representation, for example 0, and Hadamard representation 1). Then, according to formulas (4) and (5), the value of the position  $l$  will not change, regardless of the choice of parental pairs, the choice of the value  $c$  as a point of crossing-over, and the number of crossing-over operations. So we had no opportunity to receive elements with the value on position  $l$  different, than the one which have all the elements of  $P$ . And that means that we do not receive as a result of crossing-overs the elements from  $H^n$  with the opposite value (from our example is a binary value 1 or Hadamard value -1) to be set in position  $l$  in the space  $P$ . This would contradict the assumption that all space  $H^n$  is the ancestral population. Thus, the implication in this direction is true.

$\Leftarrow$

In the proof of equivalence in the other direction, we use the Restore Pattern using Crossovers (RPC) Algorithm (see Algorithm 1). Let the pattern  $W$  be an arbitrary element in space  $H^n$ . Meeting the objectives of **Theorem T1**, irrespective of the value that we are going to set to fixed, but any position (locus)  $c$  of the pattern  $W$ , we will always find a chromosome in the initial population with a value at that position of searched pattern  $W$ . This assures us that the inner loop **L1** always ends up with a variable *found* with a value of true. This, in turn, runs a block **BL1**, which carries the crossing-overs and, possibly, newly formed chromosomes attach to the pool taking part in further operations. The algorithm assumed that the offsprings replace parents. In a case when offsprings join to the current population and parents would remain in it, a block **BL1** should look like:

```

begin of BL11
  G := first_element_from(K({G,B[i]},c))
  B := B  $\cup$  K(K({G,B[i]},c),c)
  maks := maks + 2
end of BL11

```

And in a case when we want do attach to the pool the element matching the pattern, a block **BL1** should look like:

```

begin of BL12
  G := first_element_from(K({G,B[i]},c))
  B := B  $\cup$  {G}

```

---

#### Algorithm 1 RPC Algorithm

---

##### Input:

n {the length of chromosome}  
W[1,...,n]{the table containing the pattern of the chromosome}  
j {the number of chromosomes in a population}  
PB[1,...,j][1,...,n] {the table of tables containing the chromosomes of the initial population}

##### Definition:

B[1,...,j+2n][1,...,n] {the table of tables containing the chromosomes of the current population}

##### begin

B := BP {insert BP into the first n-positions of B}

maks := j

c := n

G := B[1]

##### repeat

  i := 0

  found := false

##### repeat

**begin of L1**

      i := i + 1

**if** W[c] = B[i][c] **then**

        found := true

**end if**

**end of L1**

**until** (found or (i = maks))

**if** found **then**

**begin of BL1**

      G := first\_element\_from(K({G,B[i]},c))

      B := (B  $\setminus$  K({G, B[i]},c))  $\cup$  K({G,B[i]},c)

**end of BL1**

**end if**

  c := c - 1

**until** ((not found) or (c = 0))

##### Output:

**if** found **then**

  G

**else**

  false

**end if**

**end**

---

  maks := maks + 1

**end of BL12**

The function first\_element\_from() returns, according to (6), chromosom resulting from crossing-over, having at locus  $c$  a value equals to the value of the pattern at the same position (see (4) and (5)).

Summing up, since the RPC Algorithm, meeting the objectives-led part of the Theorem, is able from the initial population—using only crossing-overs—create any element from the space  $H^n$ , so  $H^n$  is the ancestral population.

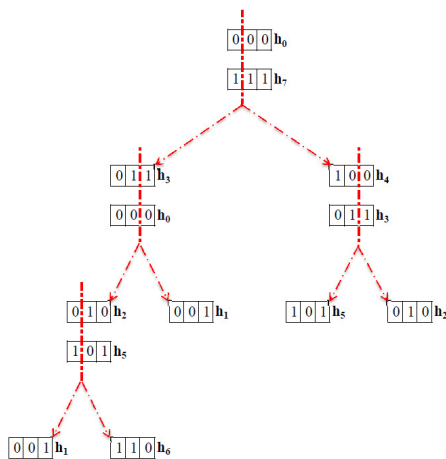


Fig. 1. Exemplary ancestral population with the primary population  $h_0$  and  $h_7$ .

### End of Proof

At the end, as a conclusion of the above **Theorem 1** let us write, without any proof, convenient Theorem in applications:

### Theorem 2

If a primary population  $P \subseteq H^n$  contains the pair of polar points, then the whole space  $H^n$  is an ancestral population.

For example, having two polar points  $h_0$  and  $h_7$  as a primary population from  $H^3$  we are able after four crossovers obtain all 8 points from the space in question, according to **Theorem 2** (see Figure 1). More crossover operations are needed when using natural selection and random points of crossing.

At this point, it seems natural to mention the **Schema Theorem** [7], one of the fundamental theorems of genetic algorithms. The **Theorem 1** shows that, if not all space  $H^n$  is the ancestral population, there must be positions (perhaps one such a position) in all chromosomes of the initial population, having the same value (locus). Thus, we should look for above-average schemes at those positions. Crossover operation is only possible to duplicate chromosomes with such a chosen scheme. Changes of this trend can only be caused by other operations (for example, mutation or inversion). Given that the vast majority of models considered, crossing is an operation with a much greater likelihood of occurrences in relation to other operations, reducing the quantity or even just reducing the growth of occurrences in the next population will be proportionately less likely.

Introduced concepts allow us to distinct and classify different populations, what is more to penetrate into the potential future directions of their evolution (states reachable, unreachable, etc.) regardless of the selected crossover algorithms, selection of parents, or the elimination of individuals. And then, we should be able to compare the genetic algorithms in terms of efficiency and optimization.

Comparing the two algorithms, we need to ensure comparability of the population, on which we conduct experiments.

It is obvious that the same algorithm, e.g. over the population of the class of expansive systems, has a chance of finding new solutions in successive generations, but over the populations form extensive class, after reviewing the current population, better solutions are no longer found.

### V. CONCLUSIONS AND FUTURE WORK

The introduced Hadamard representation allows us to prove the dependence between subsequent generations of binary chromosomes encoding antibody and antigen space. Some properties of this representation were pointed out, which allows for quick and simple operations on chromosomes indices, instead of processing the binary sequences. The main contributions of this paper are to introduce new property of the immune system called *Expansive system with global range*, and to prove for epistatic properties as well as exploration possibilities of a crossover operator. Some remarks on Schema Theorem over  $\{-1, 1\}^n$  space were also made.

Future research will show to what extent the Hadamard chromosomes are exploitable.

### REFERENCES

- [1] Clark E., Hone A., Timmis J. (2005), *A Markov Chain Model of the B-cell Algorithm*, In: LNCS 3627, Springer, 318–330.
- [2] Dasgupta D. (1999), *Artificial Immune Systems and their Applications*, Springer-Verlag, Berlin, Heidelberg.
- [3] Fogel, D., Atmar, J. W. (1990), *Comparing genetic operators with Gaussian mutations in simulated evolutionary processes using linear systems*, Biological Cybernetic, 63: 111–114.
- [4] Forest, S., Javornik, B., Smith, R.E., Perelson, A.S. (1993), *Using genetic algorithms to explore pattern recognition in the immune system*, Evolutionary Computation, 1: 191–211.
- [5] Freitas A., Timmis J. (2007), *Revisiting the Foundations of Artificial Immune Systems for Data Mining*, IEEE Trans. Evol. Comp. 11(4) 521–540.
- [6] Hadamard J. (1893), *Résolution d'une Question Relative aux Déterminants*, Bull. Sci. Math. 2(17), 240–246.
- [7] Holland J. (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.
- [8] Park, K., Carter, B. (1994), *On the effectiveness of genetic search in combinatorial optimization*, Tech. Report BU-CS-94-010, Computer Sci. Department, Boston University.
- [9] Pliszka Z., Unold O. (2010), *Metric Properties of Populations in Artificial Immune Systems*, In: Proceedings of the International Multiconference on Computer Science and Information Technology (AAIA'10), Wisla, Poland, 113–119.
- [10] Spears, W. M. (1992), *Crossover or mutation?*, D. Whitley ed., Proc. Of the 2-nd Foundations of Genetic Algorithms Workshop, Morgan Kaufman, 221–237.
- [11] Spears, W.M. (1994), *Simple population schemes*, In: Proc. Of the 1994 Evolutionary Programming Conference, World Scientific: 296–317.
- [12] Schaffer, J. D., Eshelman, L. J. (1991), *On crossover as an evolutionary viable strategy*, Proceedings of the Fourth International Conference on Genetic Algorithms, La Jolla, CA: Morgan Kaufmann, 61–68.
- [13] Timmis J., Hone A. N. W., Stibord T., Clark E. (2008), *Theoretical Advances in Artificial Immune Systems*, Theoretical Computer Science, Vol. 403, Issue 1, 11–32.
- [14] Wierchoń S.T. (2001), *Sztuczne Systemy Immunologiczne. Teoria i zastosowania*, Akademicka Oficyna Wydawnicza EXIT, Warszawa (in Polish).
- [15] Villalobos-Arias M., Coello Coello C. A., Hernandez-Lerma O. (2004), *Convergence Analysis of a Multiobjective Artificial Immune System Algorithm*, In: LNCS 3239, Springer, Berlin, Heidelberg, 226–235.

# The Fuzzy Genetic Strategy for Multiobjective Optimization

Krzysztof Pytel

Faculty of Physics and Applied Informatics

University of Lodz,

Lodz, Poland

Email: kpytel@uni.lodz.pl

**Abstract**—This paper presents the idea of fuzzy controlling of evolution in the genetic algorithm (GA) for multiobjective optimization. The genetic algorithm uses the Fuzzy Logic Controller (FLC), which manages the process of selection of individuals to the parents' pool and mutation of their genes. The FLC modifies the probability of selection and mutation of individuals' genes, so algorithms possess improved convergence and maintenance of suitable genetic variety of individuals. We accepted the well-known LOTZ problem as a benchmark for experiments. In the experiments we investigated the operating time and the number of fitness function calls needed to finish optimization. We compared results of the elementary algorithms and the modified algorithm with the modification of probability of selection and mutation of individuals. Some good results have been obtained during the experiments.

## I. INTRODUCTION

IN MANY practical problems, it's often expected that several indicators achieve optimal value at the same time, which is called multi-objective optimization problem [1][6][10]. These multiple objectives, often conflicting with each other, can accept the maximum or minimum in other points of the search space. The multi-objective optimization problem (MOP) can be stated as follows:

$$\begin{cases} \text{maximize } F(x) = [f_1(x)f_2(x)f_3(x)\dots f_m(x)] \\ \text{subject to: } g_j(x) \leq 0 \text{ for } j = 1, 2, \dots, k \\ x \in S \end{cases} \quad (1)$$

where

$$x = [x_1x_2x_3\dots x_n] \in \mathfrak{R}, n \in N, \quad (2)$$

is an  $n$ -dimensional vector of decision variables,

$$F(x) = [f_1(x)f_2(x)f_3(x)\dots f_m(x)], \quad (3)$$

are objective functions, and  $g_i(x)$  are constrains.

Let us choose an optimization problem, with  $m$  objectives, which are, without loss of generality, all to be maximized. The set of potential solutions can be parted to two subsets: dominated and not dominated.

Let  $x, y \in S$ ,  $x$  is said to be dominated by  $y$ , if  $f_i(y) \geq f_i(x)$  for all  $i = 1, 2, \dots, m$  and  $f_j(y) > f_j(x)$  for at least one index  $j$ . A solution  $x^* \in S$  is said to be pareto-optimal if there does not exist another solution  $x$ , such that  $x^*$  is dominated by  $x$ .  $F(x^*)$  is then called a pareto-optimal objective vector. The set of all the pareto-optimal objective vectors is called

a pareto-optimal front. A set of pareto-optimal solutions is usually found as a result of multiobjective optimization. The decision-maker can use his preferred method to choose the final solution from the pareto-optimal set.

Genetic algorithms stand for a class of stochastic optimization methods that simulate the process of natural evolution [3][5][7][9]. In a genetic algorithm, a population of strings (called chromosomes), which encode candidate solutions (called individuals) to an optimization problem, evolves toward better solutions. They usually search for approximate solutions of composite optimization problems. A characteristic feature of genetic algorithms is that in the process of the evolution they do not use the specific knowledge for given problem, except of fitness function assigned to all individuals. The specific knowledge for a given problem can set a trend for evolution and improve the efficiency of the algorithm.

The genetic algorithm consists of the following steps:

- 1) the choice of the first generation,
- 2) the estimation of an individuals' fitness,
- 3) the check of the stop condition,
- 4) the selection of individuals to the parents' pool,
- 5) the creation of a new generation with the use of operators of crossing and mutation,
- 6) the printing of the best solution.

The source code of genetic algorithm was published in [7]. We used this code as elementary genetic algorithm.

There are two parameters in elementary genetic algorithms which determine evolution: the probability of selection to the parents' pool and the probability of mutation. GA can be improved by the knowledge of experts. Experts can predict the course of the process of evolution. The experts' knowledge about evolution has a descriptive character and is often subjective, so we use the fuzzy logic controller to set a trend of evolution.

## II. ADAPTATION OF THE PROBABILITY OF SELECTION AND MUTATION

The probability of selection determines the ability of an individual to act as a parent and produce offspring. The chances of the individual for transferring its genetic material to the next generation increase with the probability of selection. Well-adapted individuals are the most wanted ones in the parents' pool. However, weak individuals should also hit for the



parents' pool in order to prevent violent loss of their genetic material and premature convergence. We suggest introduction of an additional FLC for evaluating each individual in the population. The FLC modifies the probability of selection using the following rules:

- enlarge the probability of selection for the individuals with the value of the fitness function of above the average in generations in which the average value of the fitness function grows with relation to the preceding generation,
- don't change the probability of selection for individuals with the value of the fitness function equal to the average in generations in which the average value of the fitness function does not change the relation to the preceding generation,
- diminish the probability of selection for individuals with the value of the fitness function below the average in generations in which the average value of the fitness function decreases with relation to of the preceding generation.

The FLC calculates the adaptation ratio for all individuals based on two parameters:

- the increase of the average fitness function for the whole population (determines the change of the fitness function between the current and the previous populations)

$$\Delta F_{pop} = F_{pop}^{(T)} - F_{pop}^{(T-1)} \quad (4)$$

- an individual's fitness (determine the difference between the average value of the fitness function in the population and an individual's fitness)

$$W_i = F_i^{(T)} - F_{pop}^{(T)} \quad (5)$$

where:

$F_i^{(T)}$  - the fitness function of individual  $i$  in moment  $T$ ,  
 $F_{pop}^{(T)}$  - the average fitness function of the whole population in moment  $T$ .

The FLC uses the center of gravity [9] defuzzification method. As the result from the controller we accepted:

$wps_i$  - the adaptation ratio of the probability of selection of individual  $i$ .

The modified probability of selection of individual  $i$  obeys the formula:

$$ps'(i) = ps(i) * wps_i \text{ for } i = 1, \dots, N, \quad (6)$$

where:

$ps'(i)$  - the modified probability of selection of individual  $i$ ,  
 $ps(i)$  - the probability of selection of individual  $i$ ,  
 $wps_i$  - the adaptation ratio of the probability of selection calculated by the FLC for the individual  $i$ .

The knowledge base of FLC is shown in Table 1 (fuzzy values of the adaptation ratio of probability of selection for individual  $i$ )

Values in the table are:

TABLE I  
FUZZY VALUES OF THE ADAPTATION RATIO OF PROBABILITY OF SELECTION FOR INDIVIDUAL  $i$

		$\Delta F_{pop}$			
		NS	ZERO	PS	PL
$W_i$	S	VS	S	A	A
	A	S	A	A	L
	L	S	A	L	L

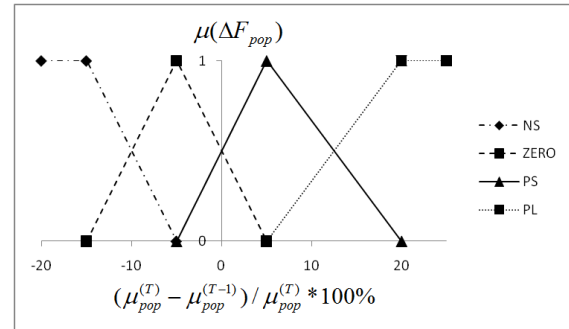


Fig. 1. The membership functions of the increase of the average fitness function for the whole population

- fuzzy sets of increase fitness function for whole population  $\Delta F_{pop}$ ,
  - o NS - negative small,
  - o ZERO - closed to zero,
  - o PS - positive small,
  - o PL - positive large,
- fuzzy sets of an individual's fitness  $W_i$  and fuzzy sets of ratio  $wps_i$ ,
  - o VS - very small,
  - o S - small,
  - o A - average,
  - o L - large.

Figures 1 - 3 shows the membership functions of the increase of the average fitness function for the whole population, an individual's fitness and the adaptation ratio of the probability of selection.

The probability of mutation determines the ability of the algorithm to explore and exploit the search space. In the initial period, mutations are frequent in order to find the solution in the whole search space (exploration mode). In the final period, mutations are rarer than at the start, so the algorithm can search the earlier established areas of possible optima (exploitation mode). The mutation of the gene can cause that the new, well adapted individual will translocate the population to the area of the total optimum. We suggest introduction of an additional FLC for evaluating each individual in the population. The FLC modifies the probability of mutation using the following rules:

- enlarge the probability of mutation of individuals with the value of the fitness function of less then the average in generations in which the average value of the fitness function decreases with relation to the preceding generation,



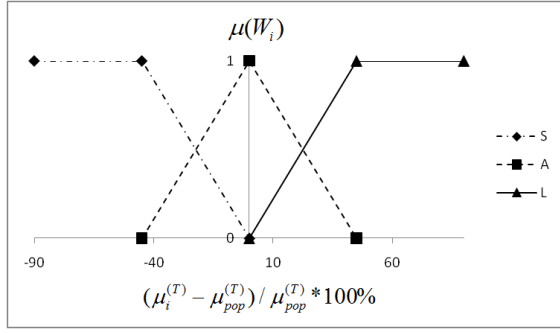


Fig. 2. The membership functions of an individual's fitness

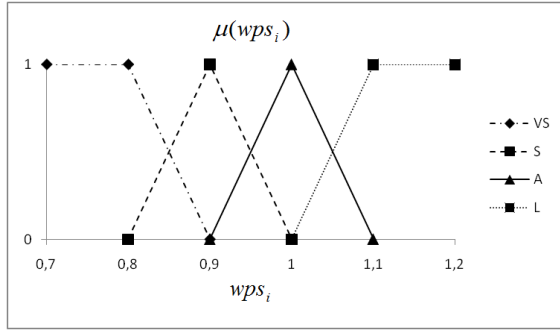


Fig. 3. The membership functions of the adaptation ratio of the probability of selection

- don't change the probability of mutation of individuals with the value of the fitness function equal to the average in generations in which the average value of the fitness function does not change in relation to the preceding generation,
- diminish the probability of mutation of individuals with the value of the fitness function above the average in generations in which the average value of the fitness function increases with relation to the preceding generation.

The FLC calculates the adaptation ratio for all individuals based on two parameters:

- the population's quality (defines the difference between the fitness function of the best individual discovered so far and the average fitness function of the current population)

$$\Delta F_{pop} = F_{max} - F_{pop}^{(T)} \quad (7)$$

- the individual's fitness (the same the parameter was used for the modification of the probability of selection)

$$W_i = F_i^{(T)} - F_{pop}^{(T)} \quad (8)$$

where:

$F_i^{(T)}$  - the fitness function of individual  $i$  in moment  $T$ ,  
 $F_{pop}^{(T)}$  - the average fitness function for the whole population in moment  $T$ .

The FLC uses the center of gravity defuzzification method. As the result from the controller we accepted:

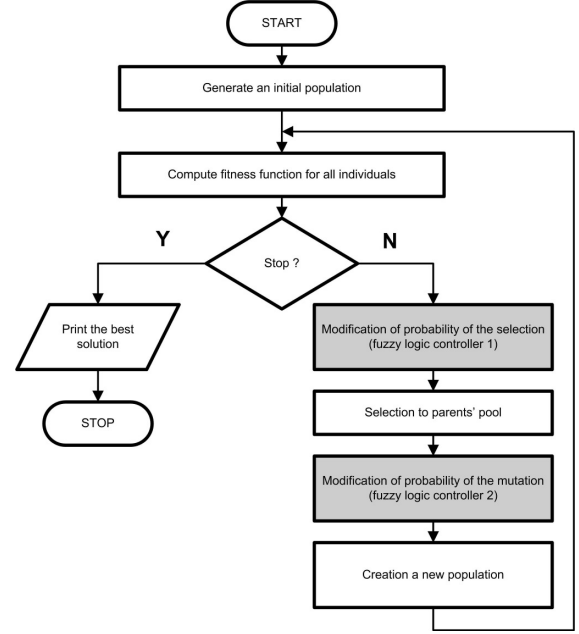


Fig. 4. The block scheme of the modified genetic algorithm

$wpm_i$  - the adaptation ratio of the probability of mutation of individual  $i$ .

The modified probability of mutation of an individual  $i$  obeys the formula:

$$pm'(i) = pm(i) * wpm_i \text{ for } i = 1, \dots, N, \quad (9)$$

where:

- $pm'(i)$  - the modified probability of mutation of individual  $i$ ,
- $pm(i)$  - the probability of mutation of individual  $i$ ,
- $wpm_i$  - the adaptation ratio of the probability of mutation calculated by the FLC for the individual  $i$ .

Figure 4 shows the block scheme of the modified genetic algorithm (the block of the fuzzy logic is noted with the grading). The construction of FLC for modification of the mutation is almost the same as FLC for modification of the selection. The construction of the fuzzy logic controller in details is considered in [8].

### III. COMPUTATIONAL EXPERIMENTS

The goal of the experiment is the verification of the idea of fuzzy controlling of evolution in the modified genetic algorithm for multiobjective optimization. The FLC estimates all individuals and modifies their probability of selection and mutation. The algorithm looks for any pareto-optimal solution. The LOTZ (Leading Ones, Trailing Zeroes) problem with the size from 50 to 100 was chosen as the test function. The LOTZ can be stated as the maximization problem of two objectives.

$$LOTZ1(x) = \sum_{i=1}^n \prod_{j=1}^i x_j, \quad (10)$$

TABLE II  
THE AVERAGE VALUES OF THE RUNNING TIME AND THE NUMBER OF FITNESS FUNCTION CALLS

		Elementary	Modification of mutation	Modification of selection	Modification of selection and mutation	SEMO	NSGA2
LOTZ50	time [s]	0,063	0,097	0,133	0,199	17,15	14,26
	number of fitness function calls	25743	15196	22985	21350	7700	10700
LOTZ60	time [s]	0,172	0,207	0,361	0,575	38,32	30,41
	number of fitness function calls	60245	31401	57052	59160	36500	27500
LOTZ70	time [s]	0,564	0,528	0,92	1,309	49,76	31,13
	number of fitness function calls	164034	75335	134509	126560	44000	28500
LOTZ80	time [s]	1,098	1,036	1,956	3,377	49,78	57,11
	number of fitness function calls	275626	137082	267112	309490	45000	49000
LOTZ100	time [s]	5,233	3,932	7,879	12,99	51,09	77,57
	number of fitness function calls	1129902	460269	984622	1125690	48000	82000

$$LOTZ2(x) = \sum_{i=1}^n \prod_{j=i}^n (1 - x_j), \quad (11)$$

$$LOTZ(x) = (LOTZ1(x), LOTZ2(x)), \quad (12)$$

where:  $x = (x_1, x_2, \dots, x_n) \in \{0, 1\}$ .

We compared algorithms with modification of selection and mutation with elementary genetic algorithm. The first population was generated randomly. All the algorithms started at the same point in the search space. The algorithms' parameters used in the experiment:

- the genes of individuals are represented by binary numbers,
- the probability of crossover = 0,8,
- the probability of mutation = 0,15,
- the number of individuals in the population = 10,
- the algorithms were stopped after finding any pareto-optimal solution.

To measure the achievements of modified algorithms, we choosed two other algorithms SEMO and NSGA2, usually used for solving LOTZ problem. SEMO is a population-based evolutionary algorithm for multiobjective optimization proposed in [4]. NSGA2 is an elitist multiobjective evolutionary algorithm proposed in [2]. We used a PISA-implementation of the algorithms written by Marco Laumanns. The SEMO and NSGA2 find a set of pareto optimal solutions, so they were stopped after finding any pareto optimal solution. For SEMO and NSGA2 we use:

- one-bit mutation,
- population size 100,
- probability of mutation 0,15
- probability of recombination 0,8.

Each algorithm was executed 10 times. In Table 2 there are average values of the running time and the number of fitness function calls obtained by the algorithms.

The graph in Figure. 5 illustrates the average running time of the algorithms.

The graph in Figure. 6 illustrates the average number of fitness function calls needed by the algorithms.

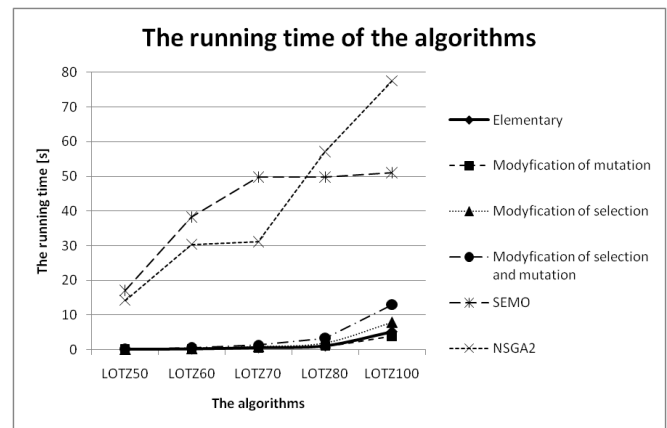


Fig. 5. The average running time of the algorithms

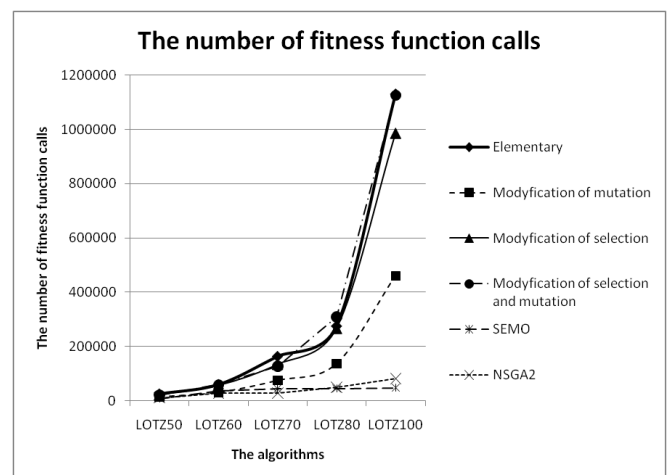


Fig. 6. The average number of the fitness function calls needed by the algorithms

## IV. CONCLUSION

Modified algorithms need less fitness function calls in all experiments than elementary algorithms, but it need more fitness function calls than algorithm SEMO and NSGA2. The running time of modified algorithms is noticeably shorter than the algorithm SEMO and NSGA2.

The adaptation of parameters of the algorithms demands additional computational effort. The running time of the functions of the large computational complexity in the large search space can be diminished. The FLC effectively manages evolution in genetic algorithms by modification of the probability of selection and mutation.

The algorithms looked for any pareto-optimal solution. In the future, we are going to check whether the adaptation of parameters of the algorithm can direct evolution toward the point in pareto front preferred by the decision-maker.

## REFERENCES

- [1] Coello, C.A.C. *The EMOO Repository: A Resource for Doing Research in Evolutionary Multiobjective Optimization* - IEEE Computational Intelligence Magazine, 45, February 2006.
- [2] Deb K., Agrawal S., Pratap A., Meyarivan T., *A fast and elitist multiobjective genetic algorithm: Nsga-II*. IEEE Trans. Evolutionary Computation, 6(2):182197, 2002.
- [3] Kwasnicka H., *Evolutionary Computation in Artificial Intelligence*, Publishing House of the Wrocław University of Technology, Wrocław, Poland, (1999) (in Polish).
- [4] Laumanns M., Thiele L., Zitzler E., *Running time analysis of multiobjective evolutionary algorithms on pseudo-boolean functions*, IEEE Trans. Evolutionary Computation, 8(2):170182, 2004.
- [5] Lobo F.G., Lima C.F., Michalewicz Z., *Parameter setting in evolutionary algorithms*, Springer, 2007.
- [6] Masatoshi Sakawa, *Genetic Algorithms and Fuzzy Multiobjective Optimization*, Kluwer Academic Publications, Boston 2002.
- [7] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*, - Springer Verlag, Berlin (1992).
- [8] Pytel K., Nawarycz T., *Analysis of the Distribution of Individuals in Modified Genetic Algorithms* [in] Rutkowski L., Scherer R., Tadeusiewicz R., Zadeh L., urada J., *Artificial Intelligence and Soft Computing*, Springer-Verlag Berlin Heidelberg 2010.
- [9] Rutkowska D., Pilinski M., Rutkowski L., *Neural Networks, Genetic Algorithms and Fuzzy Systems*, PWN Scientific Publisher, Warsaw, (1997) (in Polish).
- [10] Zitzler E., *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, PhD thesis, ETH Zrich 1999.



## New property for rule interestingness measures

Izabela Szczęch

Institute of Computing Science,  
Poznań University of Technology,  
60–965 Poznań, Poland,  
Email: iszczech@cs.put.poznan.pl

Salvatore Greco

Faculty of Economics,  
University of Catania,  
Corso Italia, 55,  
95129 Catania, Italy  
Email: salgreco@unicit.it

Roman Słowiński

Institute of Computing Science,  
Poznań University of Technology,  
60–965 Poznań, Poland,  
Systems Research Institute,  
Polish Academy of Sciences,  
01–447 Warsaw, Poland  
Email: slowinski@cs.put.poznan.pl

**Abstract**—The paper considers interestingness measures for evaluation of rules induced from data with respect to two properties: property of Bayesian confirmation and property  $Ex_1$  concerning the behavior of measures in case of entailment or refutation of the conclusion by the rule's premise. We demonstrate that property  $Ex_1$ , even though created for confirmation measures, does not fully reflect the concept of confirmation. We propose a modification of this property, called weak  $Ex_1$ , that deploys the concept of confirmation in its larger sense and allows to escape paradoxes that might appear when using measures satisfying the original  $Ex_1$  property.

### I. INTRODUCTION

DISCOVERING knowledge from data aims at finding “valid, novel and potentially useful” [5] patterns often expressed as “if... then...” rules. Typically, the number of rules generated from massive datasets is quite large, but only some of them are likely to be useful for the domain expert analyzing the data. In order to measure the relevance and utility of the discovered rules, quantitative measures, also known as interestingness or attractiveness measures, have been proposed and studied. Among the most commonly used ones there are: *support*, *confidence*, *lift*, *rule interest function*, *dependency factor*, etc. There is a rich discussion about interestingness measures for rules in data mining (see, for example, [1],[8], [12] for exhaustive reviews of the subject) as each of the measures proposed in the literature reflects different characteristics of rules. The discussion was also extended in [9], by an important issue concerning the possibility of using Bayesian confirmation measures (i.e. measures quantifying the degree to which a piece of evidence provides “support for or against” a hypothesis [7]) as interestingness measures for evaluation of rules. Moreover, the research in [13], [14], [15] shows that discovering patterns in data can be represented in terms of Bayes’ theorem. In this context, a variety of non-equivalent confirmation measures should be regarded as a useful tool able to discriminate the most interesting rules discovered by induction from data.

However, due to the plurality of ordinally non-equivalent measures and because there is no agreement which measure is the best, the choice of an interestingness measure for a particular application is non-trivial. To help to analyze measures and overcome the problem of their vast variety, some properties have been proposed. They express the user's expectations towards the behavior of measures in particular situations e.g., one could desire to use only such measures that reward the rules having a greater number of objects supporting the pattern. In general, properties group the measures according to similarities in their characteristics, thus using the measures which satisfy the desirable properties one can avoid considering unimportant rules. Different properties have been proposed and surveyed in [2], [4], [8], [9], [18].

This article concerns Bayesian confirmation measures with respect to their properties. We analyze a property denoted as  $Ex_1$ , introduced in [3], assuring that any conclusively confirmatory rule is assigned a higher value of a confirmation measure than any rule which is not conclusively confirmatory, and any conclusively disconfirmatory rule is assigned a lower value than any rule which is not conclusively disconfirmatory. We propose a modification of  $Ex_1$ , called weak  $Ex_1$ , that deploys the concept of confirmation in its larger sense and allows to escape paradoxes that might occur when using measures with  $Ex_1$  property.

The article is organized as follows. In Section 2 there are preliminaries on rules and their quantitative description. In section 3, we investigate two properties of measures, being the property of Bayesian confirmation and property  $Ex_1$ . Section 4 shows specific rankings of rules obtained using measures satisfying the property  $Ex_1$  and explains the paradox appearing in such situations. Section 5 introduces a proposition of modification of property  $Ex_1$  into weak  $Ex_1$  and shows that substituting  $Ex_1$  by weak  $Ex_1$  we escape the paradoxes. Finally, Section 6 presents conclusions.

## II. PRELIMINARIES

A rule induced from a dataset  $U$  shall be denoted by  $E \rightarrow H$  (read as “if  $E$ , then  $H$ ”). It consists of a premise (evidence)  $E$  and a conclusion (hypothesis)  $H$ .

In general, by  $\text{sup}(\gamma)$  we denote the number of objects in the dataset for which  $\gamma$  is true, e.g.,  $\text{sup}(E)$  is the number of objects in the dataset satisfying the premise, and  $\text{sup}(H, E)$  is the number of objects satisfying both the premise and the conclusion of a  $E \rightarrow H$  rule.

Moreover, the following notation shall be used throughout the paper:  $a = \text{sup}(H, E)$ ,  $b = \text{sup}(H, \neg E)$ ,  $c = \text{sup}(\neg H, E)$ ,  $d = \text{sup}(\neg H, \neg E)$ . It corresponds to a 2x2 contingency table of the premise and the conclusion.

TABLE I. CONTINGENCY TABLE OF  $E$  AND  $H$

	$H$	$\neg H$	
$E$	$a$	$c$	$a+c$
$\neg E$	$b$	$d$	$b+d$
	$a+b$	$c+d$	$ U $

Observe that  $b$  can be interpreted as the number of objects that do not satisfy the premise but satisfy the conclusion of the  $E \rightarrow H$  rule. Analogously,  $c = \text{sup}(\neg H, E)$  can be construed as the number of objects in the dataset that satisfy the premise but do not satisfy the conclusion of the  $E \rightarrow H$  rule, and  $d = \text{sup}(\neg H, \neg E)$  can be interpreted as the number of objects in the dataset that satisfy neither the premise nor the conclusion of the  $E \rightarrow H$  rule. Moreover, the following relations occur:  $a+c = \text{sup}(E)$ ,  $a+b = \text{sup}(H)$ ,  $b+d = \text{sup}(\neg E)$ ,  $c+d = \text{sup}(\neg H)$ , and the cardinality of the dataset  $U$ , denoted by  $|U|$ , is the sum of  $a$ ,  $b$ ,  $c$  and  $d$ .

Reasoning in terms of  $a$ ,  $b$ ,  $c$  and  $d$  is natural and intuitive for data mining techniques since all observations are gathered in some kind of an information table describing each object by a set of attributes. However,  $a$ ,  $b$ ,  $c$  and  $d$  can also be regarded as frequencies that can be used to estimate probabilities: e.g.,  $\text{Pr}(E) = (a+c)/|U|$  or  $\text{Pr}(H) = (a+b)/|U|$ .

## III. PROPERTIES OF INTERESTINGNESS MEASURES

The problem of choosing an appropriate interestingness measure for a certain application is non-trivial because the number and variety of measures proposed in the literature is overwhelming. To help to analyze measures, some properties have been proposed, expressing the user's expectations towards the behavior of measures in particular situations. Properties of measures group them according to similarities in their characteristics. Using the measures which satisfy the desirable properties, one can avoid considering unimportant rules.

Our analysis of properties is conducted from the view point of Bayesian confirmation theory. We propose a modification of property  $\text{Ex}_1$ , called *weak*  $\text{Ex}_1$ , that deploys the concept of confirmation in its larger sense. We

demonstrate that using property  $\text{Ex}_1$  can lead to paradoxical situations and thus, we propose to substitute  $\text{Ex}_1$  by weak  $\text{Ex}_1$ .

### A. Property of Bayesian confirmation

Bayesian confirmation theory assumes the existence of probability  $\text{Pr}$ . Given a proposition  $X$ ,  $\text{Pr}(X)$  represents the probability of  $X$ , and given  $X$  and  $Y$ ,  $\text{Pr}(X|Y)$  is the probability of  $X$  given  $Y$ , i.e.  $\text{Pr}(X|Y) = \text{Pr}(X \wedge Y)/\text{Pr}(Y)$ .

Generally speaking, a measure possessing the property of Bayesian confirmation is expected to obtain values greater than 0 when the premise of a rule confirms the conclusion of a rule, values equal to 0 when the rule's premise and conclusion are neutral to each other, and finally, values smaller than 0 when the premise disconfirms the conclusion.

Formally, an interestingness measure  $c(H, E)$  has the property of Bayesian confirmation if and only if it satisfies the following conditions (BC):

$$c(H, E) \begin{cases} > 0 & \text{if } \text{Pr}(H|E) > \text{Pr}(H), \\ = 0 & \text{if } \text{Pr}(H|E) = \text{Pr}(H), \\ < 0 & \text{if } \text{Pr}(H|E) < \text{Pr}(H). \end{cases} \quad (1)$$

The (BC) definition identifies confirmation with an increase in the probability of the conclusion  $H$  provided by the premise  $E$ , neutrality with the lack of influence of the premise  $E$  on the probability of conclusion  $H$ , and finally disconfirmation with a decrease of probability of the conclusion  $H$  imposed by the premise  $E$  [2].

It is important to note that there are many different, but logically equivalent, ways of expressing that  $E$  confirms  $H$ :

- $\text{Pr}(H|E) > \text{Pr}(H)$
- $\text{Pr}(H|E) > \text{Pr}(H|\neg E)$
- $\text{Pr}(H|E) > \text{Pr}(E|\neg H)$

Since they are equivalent (see also [7], [11]), one can also express the (BC) conditions as:

$$c(H, E) \begin{cases} > 0 & \text{if } \text{Pr}(H|E) > \text{Pr}(H|\neg E), \\ = 0 & \text{if } \text{Pr}(H|E) = \text{Pr}(H|\neg E), \\ < 0 & \text{if } \text{Pr}(H|E) < \text{Pr}(H|\neg E). \end{cases} \quad (2)$$

To avoid ambiguity, we shall denote the above formulation as (BC'). According to it  $E$  confirms  $H$  when  $E$  raises the probability of  $H$ , and  $E$  raises the probability of  $H$  if the probability of  $H$  given  $E$  is higher than the probability of  $H$  given non  $E$ .

Measures that possess the property of confirmation are referred to as *confirmation measures* or *measures of confirmation*. For a given rule  $E \rightarrow H$ , interestingness measures with the property of confirmation express the credibility of the following proposition: *H is satisfied more frequently when E is satisfied, rather than when E is not satisfied*. By using interestingness measures that possess this property one can filter out rules which are misleading and disconfirm the user, and this way, limit the set of induced rules only to those that are meaningful [17]. Let us also stress that the catalogue of confirmation measures available

in the literature is quite large and the condition (BC) (or (BC')) equivalently) does not favor one single measure as the most adequate [3], [6].

The discussion brought up in [9] about using the confirmation measures as interestingness measures for decision rules within rough set approach and, more generally, within data mining, machine learning and knowledge discovery, leads to the conclusion that the group of measures with property of Bayesian confirmation should be considered a valuable and meaningful tool for assessing the quality of rules induced from data. Using the quantitative confirmation theory for data analysis allows to benefit from the ideas of such prominent researchers as Carnap [2], Hempel [10] and Popper [16].

#### B. Property $Ex_1$

To handle the plurality of alternative confirmation measures, Crupi, Tentori and Gonzalez [3] have proposed a property (principle)  $Ex_1$  resorting to considering inductive logic as an extrapolation from classical deductive logic. On the basis of classical deductive logic they construct a function  $v$ :

$$v(H, E) = \begin{cases} \text{the same positive value, denoted as } V, & \text{if } E \models H; \\ \text{the same negative value, denoted as } -V, & \text{if } E \models \neg H; \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

For any argument  $(H, E)$  function  $v$  assigns it the same positive value  $V$  (e.g., +1) if and only if the premise  $E$  of the rule entails the conclusion  $H$  (i.e.  $E \models H$ ). The same value but of opposite sign  $-V$  (e.g., -1) is assigned if and only if the premise  $E$  refutes the conclusion  $H$  (i.e.  $E \models \neg H$ ). In all other cases (i.e. when the premise is not conclusively confirmatory nor conclusively disconfirmatory for the conclusion) function  $v$  obtains value 0.

Let us observe, that any confirmation measure obtains positive (negative) values whenever function  $v(H, E)$  is positive (negative). However, according to Crupi et al., the relationship between the logical implication or refutation of  $H$  by  $E$ , and the conditional probability of  $H$  subject to  $E$  should go further and demand fulfillment of the following principle ( $Ex_1$ ):

$$\text{if } v(H_1, E_1) > v(H_2, E_2) \text{ then } c(H_1, E_1) > c(H_2, E_2) \quad (4)$$

Property  $Ex_1$  guarantees that the measure will assign a greater value to any conclusively confirmatory rule (i.e. such that  $E \models H$ ) than to any rule which is not conclusively confirmatory. Moreover, rules that are conclusively disconfirmatory (i.e. such that  $E \models \neg H$ ) will obtain smaller values of interestingness measures than any rule which is not conclusively disconfirmatory.

Let us consider an example of drawing cards from a standard deck to review the consequences of property  $Ex_1$  in

three possible situations: conclusively confirmatory, non-conclusively confirmatory (or disconfirmatory), and conclusively disconfirmatory.

A rule  $r_1$ : *if  $x$  is a jack then  $x$  is a face-card* is conclusively confirmatory as the premise (drawing a jack) entails (i.e. confirms in 100%) the conclusion that the drawn card is a face-card. The entailment of the conclusion  $H$  by the premise  $E$  ( $E \models H$ ), implies that there cannot be any counterexamples to the rule (i.e.  $c=0$ ). Such conclusively confirmatory rules should be assigned a maximal value  $V$  of a function  $v(H, E)$ .

An inverse rule  $r_2$ : *if  $x$  is a face-card then  $x$  is a jack* should be regarded as non conclusively confirmatory. Drawing a face-card one can be lucky to get a jack, but it is not a 100% sure situation, therefore the premise does not entail the conclusion and the rule is not conclusively confirmatory. Moreover, the rule is also non conclusively disconfirmatory as the premise does not refute (i.e. disconfirm in 100%) the conclusion. For rules like  $r_2$  function  $v(H, E)$  obtains value 0, which implies that confirmation measures with property  $Ex_1$  assign to such rules smaller values than to conclusively confirmatory rules.

A conclusively disconfirmatory rule could be  $r_3$ : *if  $x$  is seven of spades then  $x$  is a face-card*. Here, the premise of drawing the seven of spades disconfirms in 100% the conclusion that the drawn card is a face-card. The refutation of the conclusion  $H$  by the premise  $E$  ( $E \models \neg H$ ), implies that there cannot be any positive examples to the rule (i.e.  $a=0$ ). Such conclusively disconfirmatory rules should be assigned a minimal value  $-V$  of a function  $v(H, E)$ .

The ordering based on function  $v$ :  $v(r_1) > v(r_2) > v(r_3)$ , implies the following relations for any confirmation measure possessing the  $Ex_1$  property:  $c(r_1) > c(r_2) > c(r_3)$ .

Concluding, measures satisfying property  $Ex_1$  have the ability to rank the rules in such a way that those in which the premise entails the conclusion (e.g., the rule: *if  $x$  is a jack then  $x$  is a face-card*) are on top of the ranking, those in which the premise refutes the conclusion (e.g., *if  $x$  is seven of spades then  $x$  is a face-card*) are on the very bottom, and rules which are neither 100% sure nor 100% false are in between. Let us also remark, that entailment is equivalent to  $\Pr(H|E)=1$ , i.e. to situations when there are no counterexamples to the rule ( $c=0$ ), and that refutation is equivalent to  $\Pr(H|E)=0$ , i.e. to situations when there are no positive examples to the rule ( $a=0$ ).

#### IV. PARADOXES OF $Ex_1$ PROPERTY

Property  $Ex_1$  was introduced to assure that rules for which the premise entails the conclusion (i.e. conclusively confirmatory rules) are assigned a higher value of a confirmation measure than any rule which is not conclusively confirmatory. Furthermore, rules for which the premise refutes the conclusion (i.e. conclusively disconfirmatory rules) are assigned a lower value than any rule which is not conclusively disconfirmatory. Ranking of rules depending on entailment and refutation seems naturally desirable, however boiling the consideration down to only two situations: when

there are no counterexamples to the rule (i.e.  $E \models H$ ,  $c=0$ ), and when there are no positive examples to the rule (i.e.  $E \models \neg H$ ,  $a=0$ ) can result in paradoxes.

Let us explain our point of view by taking into account the formulation of (BC') conditions stating that:

$E$  confirms  $H$  if the probability of  $H$  given  $E$  is higher than the probability of  $H$  not given  $E$ . We believe that it is reasonable to conclude that, *in case of confirmation*, a confirmation measure  $c(H, E)$  should express *how much it is more probable to have  $H$  when  $E$  is present rather than when  $E$  is absent*. In this context, the following example shows a paradox caused by the property of  $Ex_1$ .

Let us consider two cases in which the number of objects in  $U$  is distributed over  $a, b, c$  and  $d$  in the following manner:

Case  $\alpha$ :  $a_\alpha=10, b_\alpha=9, c_\alpha=0, d_\alpha=1$ ;

Case  $\beta$ :  $a_\beta=9, b_\beta=0, c_\beta=1, d_\beta=10$ .

In case  $\alpha$  a rule  $r_\alpha: E_\alpha \rightarrow H_\alpha$  is supported by  $a_\alpha=10$  objects from  $U$ , there are 9 objects supporting the rule's conclusion but not its premise ( $b_\alpha=9$ ), there are no counterexamples to the rule ( $c_\alpha=0$ ), and there is 1 object not supporting the rule's premise nor its conclusion ( $d_\alpha=0$ ). Analogously, in case  $\beta$  a rule  $r_\beta: E_\beta \rightarrow H_\beta$  is supported by  $a_\beta=9$  objects from  $U$ , there are no objects supporting the rule's conclusion but not its premise ( $b_\beta=0$ ), there is only 1 counterexamples to the rule ( $c_\beta=1$ ), and there are 10 object not supporting the rule's premise nor its conclusion ( $d_\beta=10$ ). The rule  $r_\alpha$  is conclusively confirmatory, as it has no counterexamples in  $U$  (i.e. the premise entails the conclusion). On the other hand, the rule  $r_\beta$  is non conclusively confirmatory because there exists in  $U$  one counterexample to that rule. Thus, in case  $\alpha$  the value of a confirmation measure should be greater than in case  $\beta$  if  $Ex_1$  holds.

Let us now also incorporate the idea that a confirmation measure  $c(H, E)$  should express how much it is more probable to have  $H$  when  $E$  is present rather than when  $E$  is absent. One can see that  $\Pr(H_\alpha|E_\alpha) = a_\alpha/(a_\alpha+c_\alpha) = 1$  and  $\Pr(H_\alpha|\neg E_\alpha) = b_\alpha/(b_\alpha+d_\alpha) = 0.9$  in case  $\alpha$ , while in case  $\beta$   $\Pr(H_\beta|E_\beta) = a_\beta/(a_\beta+c_\beta) = 0.9$  and  $\Pr(H_\beta|\neg E_\beta) = b_\beta/(b_\beta+d_\beta) = 0$ . For case  $\alpha$  and  $\beta$ , if  $Ex_1$  holds, passing from the situation when the premise is absent to the situation when the premise is present, we assign a greater value of a confirmation measure when we have a 10% increment of the probability of the conclusion (case  $\alpha$ ) rather than when the same increment is of 90% (case  $\beta$ ). A confirmation measure possessing property  $Ex_1$  favors rule  $r_\alpha$  over  $r_\beta$ , which is a paradox when we analyze how much more probable it is to have the rule's conclusion when the premise is present rather than when it is absent.

Analogously, let us interpret (BC') conditions as:  $E$  disconfirms  $H$  if the probability of  $H$  given  $E$  is smaller than the probability of  $H$  not given  $E$ . Thus, *in case of disconfirmation* a confirmation measure  $c(H, E)$  should express *how much it is less probable to have  $H$  when  $E$  is present rather than when  $E$  is absent*. In this context, the following example shows a paradox caused by the property of  $Ex_1$ .

Let us consider two cases in which the number of objects in  $U$  is distributed over  $a, b, c$  and  $d$  in the following manner:

Case  $\gamma$ :  $a_\gamma=0, b_\gamma=1, c_\gamma=10, d_\gamma=9$ ;

Case  $\delta$ :  $a_\delta=1, b_\delta=10, c_\delta=9, d_\delta=0$ .

In case  $\gamma$  a rule  $r_\gamma: E_\gamma \rightarrow H_\gamma$  is not supported by any object from  $U$  ( $a_\gamma=0$ ), there is 1 object supporting the rule's conclusion but not its premise ( $b_\gamma=1$ ), there are 10 counterexamples to the rule ( $c_\gamma=10$ ), and there are 9 objects not supporting the rule's premise nor its conclusion ( $d_\gamma=9$ ). Analogously, in case  $\delta$  a rule  $r_\delta: E_\delta \rightarrow H_\delta$  is supported by one object from  $U$  ( $a_\delta=1$ ), there are 10 objects supporting the rule's conclusion but not its premise ( $b_\delta=10$ ), there are 9 counterexamples to the rule ( $c_\delta=9$ ), and there are no object not supporting the rule's premise nor its conclusion ( $d_\delta=0$ ). The rule  $r_\gamma$  is conclusively disconfirmatory, as it has no positive examples in  $U$  (i.e. the premise refutes the conclusion). On the other hand, the rule  $r_\delta$  is non conclusively disconfirmatory because there exists in  $U$  one positive example to that rule. Thus, in case  $\gamma$  the disconfirmation should be greater than in case  $\delta$  if  $Ex_1$  holds, i.e. the value of a confirmation measure should be smaller in case  $\gamma$  than in case  $\delta$ .

From the view point of (BC') condition concerning disconfirmation, measure  $c(H, E)$  should express how much it is less probable to have  $H$  when  $E$  is present rather than when  $E$  is absent. The conditional probabilities for the two exemplary cases  $\gamma$  and  $\delta$  are:  $\Pr(H_\gamma|E_\gamma) = a_\gamma/(a_\gamma+c_\gamma) = 0$  and  $\Pr(H_\gamma|\neg E_\gamma) = b_\gamma/(b_\gamma+d_\gamma) = 0.1$ , and  $\Pr(H_\delta|E_\delta) = a_\delta/(a_\delta+c_\delta) = 0.1$  and  $\Pr(H_\delta|\neg E_\delta) = b_\delta/(b_\delta+d_\delta) = 1$ .

For case  $\gamma$  and  $\delta$ , if  $Ex_1$  holds, passing from the situation when the premise is absent to the situation when the premise is present, we should have a smaller value of confirmation measure (greater disconfirmation) when we have a 10% decrement of probability of the conclusion (case  $\gamma$ ) rather than when the same decrement is of 90% (case  $\delta$ ). A confirmation measure possessing property  $Ex_1$  treats rule  $r_\delta$  as less disconfirmatory than  $r_\gamma$ , which is a paradox when we analyze how much less probable it is to have the rule's conclusion when the premise is present rather than when it is absent.

The considerations for cases  $\alpha$ - $\delta$  show that the requirements forming  $Ex_1$  are not sufficient as using measures with this property can lead to paradoxes. Remark that in case of confirmation,  $Ex_1$  concerns situations of entailment, which is equivalent to  $\Pr(H|E)=1$ . However, confirmation should express how much it is more probable to have  $H$  when  $E$  is present rather than when  $E$  is absent. Thus, the requirement  $\Pr(H|E)=1$  is not sufficient, and property  $Ex_1$  should be modified to take into account also the value of  $\Pr(H|\neg E)$ . Analogical requirements concern the case of disconfirmation. These considerations lead to important modifications of property  $Ex_1$ , called weak  $Ex_1$ .

#### V. MODIFICATION OF $EX_1$ INTO WEAK $EX_1$ PROPERTY

Property  $Ex_1$  can be regarded as one-sided because it focuses on situations when  $E \models H$  (i.e. there are no



counterexamples to a rule and  $c=0$ ), and situations when  $E \models \neg H$  (i.e. there are no positive examples to a rule and  $a=0$ ). In our opinion, the concept of confirmation is too complex and rich to be boiled down simply to verification whether there are no counterexamples or no positive examples.

To formulate the proposition of modification of the  $Ex_1$  property, let us recall the interpretation of (BC') conditions:

- in case of confirmation, a confirmation measure  $c(H, E)$  should express how much it is more probable to have  $H$  when  $E$  is present rather than when  $E$  is absent,
- in case of disconfirmation a confirmation measure  $c(H, E)$  should express how much it is less probable to have  $H$  when  $E$  is present rather than when  $E$  is absent.

Taking into account such interpretations we can formulate a property called weak  $Ex_1$ , which generalizes the original  $Ex_1$  property:

$$\text{if } v(H_1, E_1) > v(H_2, E_2) \text{ and } v(H_1, \neg E_1) < v(H_2, \neg E_2) \quad (5) \\ \text{then } c(H_1, E_1) > c(H_2, E_2)$$

Property weak  $Ex_1$  guarantees that a confirmation measure  $c(H, E)$  cannot attain its maximal value unless the two following conditions are satisfied:

- $E \models H$
- $\neg E \models \neg H$

Let us also remark that the condition  $E \models H$  is equivalent to  $\Pr(H|E)=1$  and to  $c=\sup(-H, E)=0$ , because

$$\Pr(H|E) = \frac{\sup(H, E)}{\sup(H, E) + \sup(-H, E)} = \frac{a}{a+c} = 1 \Leftrightarrow c=0.$$

Furthermore, the condition  $\neg E \models \neg H$  can be equivalently expressed as  $\Pr(H|\neg E)=0$  or  $b=\sup(H, \neg E)=0$ , since

$$\Pr(H|\neg E) = \frac{\sup(H, \neg E)}{\sup(H, \neg E) + \sup(-H, \neg E)} = \frac{b}{b+d} = 0 \Leftrightarrow b=0.$$

Analogously, property weak  $Ex_1$  guarantees that the confirmation measure  $c(H, E)$  cannot attain its minimal value unless the two following conditions are satisfied:

- $E \not\models \neg H$
- $\neg E \models H$

Let us note that the condition  $E \not\models \neg H$  is equivalent to  $\Pr(H|\neg E)=0$  and to  $a=\sup(H, E)=0$ , as

$$\Pr(H|\neg E) = \frac{\sup(H, E)}{\sup(H, E) + \sup(-H, E)} = \frac{a}{a+c} = 0 \Leftrightarrow a=0.$$

Moreover, the condition  $\neg E \models H$  can be equivalently expressed as  $\Pr(H|\neg E)=1$  or as  $d=\sup(-H, \neg E)=0$ , because

$$\Pr(H|\neg E) = \frac{\sup(H, \neg E)}{\sup(H, \neg E) + \sup(-H, \neg E)} = \frac{b}{b+d} = 1 \Leftrightarrow d=0.$$

Using the property  $Ex_1$  can lead to unwanted situations where we favor one rule over another contrary to the increase in the confirmation or decrease of disconfirmation.

Our modification of  $Ex_1$  into weak  $Ex_1$  escapes that problem, because the condition  $c=0$  (in case of confirmation) and  $a=0$  (in case of disconfirmation) that were present in original formulation of  $Ex_1$  are now extended to  $c=b=0$  (in case of confirmation) and  $a=d=0$  (in case of disconfirmation). If we consider a confirmation measure that satisfies weak  $Ex_1$ , we do not demand that it should have a greater value in case  $\alpha$  rather than in case  $\beta$ , nor vice versa. Thus, the paradox disappears under conditions of weak  $Ex_1$  property. Moreover, if we consider a confirmation measure that satisfies weak  $Ex_1$ , we do not demand that it should have a smaller value in case  $\gamma$  rather than in case  $\delta$ , nor vice versa. Thus, the paradox disappears under conditions of weak  $Ex_1$ .

The modifications introduced to the  $Ex_1$  property are indispensable and allow to deploy the concept of confirmation in its larger sense. Therefore we postulate to substitute  $Ex_1$  by weak  $Ex_1$  property.

## VI. CONCLUSION

Analysis of rule interestingness measures with respect to their properties is an important research area. It helps to identify groups of measures that are truly meaningful. A group of measures satisfying the Bayesian confirmation property has been identified as important and useful for evaluation of patterns in form of rules [9], [13]. To handle the plurality of alternative, ordinally non-equivalent confirmation measures, property  $Ex_1$  has been proposed [3]. It relates to entailment or refutation of the rule's conclusion by its premise. The formulation of  $Ex_1$  reacts only to the absence of counterexamples or positive examples to a rule. Such approach does not reflect the deep meaning of confirmation stating that a confirmation measure should give an account of the credibility that it is more probable to have the rule's conclusion when the premise is present, rather than when the premise is absent. On the basis of such understanding of the concept of confirmation, we propose modification of  $Ex_1$  property, called weak  $Ex_1$ . It takes into account not only the value of  $\Pr(H|E)$  but also  $\Pr(H|\neg E)$ , and this way fully relates to confirmation concept.

## ACKNOWLEDGMENT

The research of the first and third author has been supported by the Polish Ministry of Science and Higher Education (grant no. N N519 314435).

## REFERENCES

- [1] I. Brzezińska, S. Greco, R. Słowiński, "Mining Pareto-optimal rules with respect to support and anti-support." *Engineering Applications of Artificial Intelligence*, vol. 20, no. 5, 587-600, 2007.
- [2] R. Carnap, *Logical Foundations of Probability*, 2nd ed., University of Chicago Press, Chicago, 1962.
- [3] V. Crupi, K. Tentori, M. Gonzalez, "On Bayesian measures of evidential support: Theoretical and empirical issues," *Philosophy of Science*, vol. 74, 229-252, 2007.

- [4]. E. Eells, B. Fitelson, "Symmetries and asymmetries in evidential support." *Philosophical Studies*, vol. 107, no. 2, 129-142, 2002.
- [5]. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, "From data mining to knowledge discovery: an overview." In: Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthursamy, R. (eds.) *Advances in Knowledge Discovery and Data Mining*, AAAI Press, pp. 1-34, 1996.
- [6]. B. Fitelson, "The Plurality of Bayesian Measures of Confirmation and the Problem of Measure Sensitivity," *Philosophy of Science*, vol. 66, 362-378, 1999.
- [7]. B. Fitelson, "Studies in Bayesian Confirmation Theory," Ph.D. Thesis, University of Wisconsin, Madison, 2001.
- [8]. L. Geng, H.J. Hamilton, "Interestingness Measures for Data Mining: A Survey." *ACM Computing Surveys*, vol. 38, no. 3, article 9, 2006.
- [9]. S. Greco, Z. Pawlak, R. Słowiński, "Can Bayesian confirmation measures be useful for rough set decision rules?" *Engineering Applications of Artificial Intelligence*, 17: 345-361, 2004.
- [10]. C. G. Hempel, "Studies in the logic of confirmation (I)." *Mind* 54, 1-26, 1945.
- [11]. P. Maher, *Confirmation Theory. The Encyclopedia of Philosophy* (2nd ed.). Macmillan Reference, USA, 2005.
- [12]. K. McGarry, "A survey of interestingness measures for knowledge discovery." *The Knowledge Engineering Review*, vol. 20, no.1, 39-61, 2005.
- [13]. Z. Pawlak, "Rough sets, decision algorithms and Bayes' theorem." *European Journal of Operational Research*, 136, pp.181-189, 2002.
- [14]. Z. Pawlak, "Some Issues on Rough Sets." *Transactions on Rough Sets I*, LNCS 3100, pp.1-58, 2004.
- [15]. Z. Pawlak, "Decision Rules and Dependencies." *Fundamenta Informaticae*, 60, pp.33-39, 2004.
- [16]. K. R. Popper, *The Logic of Scientific Discovery*, Hutchinson, London, 1959.
- [17]. I. Szczęch, "Multicriteria Attractiveness Evaluation of Decision and Association Rules." *Transactions on Rough Sets X*, LNCS 5656, pp.197-274, 2009.
- [18]. P.-N. Tan, V. Kumar, J. Srivastava, "Selecting the right objective measure for association analysis." *Information Systems*, vol. 29, no. 4, 293-313, 2004.

# The Add-Value of Cases on WUM Plans Recommendation

Cristina Wanzeller

Escola Superior de Tecnologia e Gestão  
Instituto Politécnico de Viseu e CI&DETS  
Campus Politécnico, 3505-510 Viseu, PORTUGAL  
Email: cwanzeller@di.estv.ipv.pt

Orlando Belo

ALGORITMI R&D Centre  
University of Minho  
PORTUGAL  
Email: obelo@di.uminho.pt

**Abstract**—Web Usage Mining is nowadays extremely useful to a diverse and growing number of users, from all types of organizations trying hard to reach the goals of their Web sites. However, inexperienced users, in particular, face several difficulties on developing and applying this kind of mining processes. One crucial and challenging task is selecting proper mining methods to deal with clickstream data analysis problems. We have been engaged on designing, developing and implementing a case based reasoning system, specifically devoted to assist users on knowledge discovery from clickstream data. The system's main aim is to recommend the most suited mining plans, according to the nature of the problem under analysis. In this paper we present such system, giving emphasis to the retrieving of similar cases using a preliminary constructed case base.

## I. INTRODUCTION

WEB based technology is widespread, but implementing and administrating Web sites still are activities increasingly complicated and very time consuming to most of the organizations. The Web has matured and the users have diverse, rising and unusual requirements. Indeed, site usage differs very often from expectations, demanding deep decision support to orient improvements. Site promoters require objective feedback regarding site effectiveness evaluation and insights how to enhance the Web offers. Hence, knowing and understanding visitors' behaviour is strategic to achieve site goals and the maximize Web's potential.

*Web Usage Mining* (WUM) is related to the application of the general *Knowledge Discovery* (KD) processes to data related with the interaction activity between visitors and Web sites, known as clickstream or usage data. WUM is an important tool to a large diversity and increasing number of decision maker users along the organization, having very different levels of knowledge in this area. Inexperienced users face many difficulties, among them, the crucial challenge of selecting proper *Data Mining* (DM) methods to deal with clickstream data analysis problems. Conversely, skilled users hold acquired know-how that may be especially relevant to the remaining users, surrounded by the same environment and confronted to similar decision problems to solve. Ideally, this know-how should be available in order to be shared and reused across the organization, creating new forms of synergies and empowering potential analysts. Moreover, the

knowledge obtained from these experiences may be reused along diverse organizations, enlarging its application and usefulness.

Having the referred issues in mind, we decided to build a system, named *Mining Plans Selector* (MPS), especially devoted to assist users on developing and applying WUM processes. The past successful WUM exercises of the organization are the base that sustains the followed approach, based on the *Case Based Reasoning* (CBR) paradigm. CBR is a learning and problem solving approach [1], [10], [17], [18]. Instead of relying solely on general knowledge of a problem domain or making associations along the generalized relationships between problem descriptors and conclusions, CBR is able to utilize the specific knowledge of previously experienced, concrete problem situations or cases [8] [18]. A new problem is solved by finding a similarity to the formal approach of the past case and reusing it in the new problem situation. A second important difference is that CBR also is an approach to incremental and sustained learning, since a new experience is retained each time a problem has been solved, making it immediately available for future problems [1].

The MPS system behaves as a corporative tool to capture, manage and reuse the previous WUM application cases. The system's main aim is to suggest the most suited WUM plans, according to the nature of the problem under analysis. The MPS also provides support to collect and organize the knowledge gained from the experience on solving WUM problems, bringing such knowledge up to date and promoting the system's sustained incremental learning. New WUM processes are stored on a collective case base, centralizing a key resource to the MPS's capacity to solve problems in a corporative knowledge base.

Assisting decisions within KD processes is not a new initiative. There are some works that explore the CBR paradigm to undertake related purposes. For instance, the Mining Mart project [13] represents several efforts regarding the reuse of successful data pre-processing processes, appealing to a case based metadata repository. However, to help the users on establishing the mapping between the problem at hands and the stored ones, this system doesn't explore the potential meta-model neither the typical CBR methods,. The main focus of active support lies on the adaptation of the selected case to the current problem. Furthermore, this project is cen-

tered in pre-processing activities, not in DM or KD processes.

Another example is the *MetaL* project [12]. This project involved multiple research and development initiatives, some of which based on the CBR paradigm (e.g. [6], [11]). The main aim was to assist the user in the model selection step of the KD process. The project attention focused mainly on the algorithms selection issue, within regression and classification problems. Contrariwise, our work has a different perspective and scope. MPS is devoted to the WUM specific domain and considers processes development at distinct levels. MPS previews assistance on models selection, comprising diverse DM functions and processes involving transformation operations and multiple stages. Besides, the system reaches a greater level of abstraction.

In this paper we explain the motivation of our work and the followed strategy, and we present briefly the MPS system. We are testing the system more exhaustively, particularly the retrieving of solutions to similar WUM problems using a preliminary constructed case base. This case base contains WUM application examples describing and reproducing experiences available. MPS is able to capture knowledge from experience, using a semi-automatic approach, and retrieves similar WUM processes, giving a specific target dataset and analysis requirements. In section II, we present the challenge we are trying to address. We describe our approach, particularly the case base (section III) and the MPS system (section IV) main characteristics. Additionally, in section V we present some of the most relevant issues involved on the construction of the preliminary case base, and in section VI we discuss the process of retrieving similar WUM application cases and the general results of its evaluation, using the preliminary case base.

## II. MINING CLICKSTREAM DATA

Clickstream or usage data is automatically logged by Web servers, being a very rich and valuable source of visitors' behavior information. This data provides a detailed record of every single action taken by the visitor, besides the outcome of the process, typically captured on traditional off-line interactions. Moreover, clickstream data is captured implicitly without questioning users directly, providing a non-intrusive way to obtain objective feedback. Therefore, exploring WUM to extract knowledge from this and related data (e.g. users' demographic and transactions' information) has potentially enormous benefits to organizations [20]. Some important and actionable areas of WUM exploration consist of Web personalization, business intelligence, system performance improvement and site content and structure enhancement [19]. For instance, known examples of Web personalization, namely automatic recommendation, include Amazon.com's personalized recommendations and music or playlist recommenders such as Mystrand.com commercial systems [14].

Naturally, electronic commerce sites get much attention, both in professional and research arenas. Electronic commerce is considered a "killer" domain for DM since many of

the ingredients necessary for successful DM are easily satisfied, including [8] [9]:

- (i) wide records, i.e. many attributes or variables;
- (ii) many records, i.e. large volume of data;
- (iii) controlled data collection (e.g. electronic data gathering);
- (iv) results can be evaluated and return on investment measured;
- (v) action can easily be taken (e.g. change the site, offer cross-sells).

In electronic commerce the underlying goal is quite objective, typically to increase sales and profit, and may be achieved by understanding properly customer access behavior. Some businesses exist only virtually on the Web and, obviously, improving offers and even previewing needs are crucial to all organization members.

As any other KD process, WUM is an open-ended, exploratory and participant driven process, involving several actions and decisions, which usually comprise [4]: (i) picking relevant data (dataset and variables); (ii) identifying proper DM functions; (iii) choosing suitable models or algorithms and setting its parameters; (vi) transforming data to improve its quality, to better fit the methods assumptions and to answer a concrete analysis problem. Those activities and decisions are not trivial. By the contrary. Selecting proper mining methods, i.e. functions and models, and applying them to the available data are known challenges of the KD process development. Among multiple issues, they require an appropriate reformulation of the practical problem into a DM problem and a deeper technical understanding of the methods, being also influenced by many kinds of factors, often complex and subjective, such as the characteristics of the available data and the process preference or success criteria. Besides, some methods overlap in terms of the problems they can solve. Consequently, KD activities are typically accomplished repetitively, following different directions and testing several variants of each direction. Examples of variants include trying and comparing different attributes selection, data transformations and models parameter's settings. In short, KD and WUM are complex and very time consuming processes, frequently not leading to useful results for a particular goal.

As expected, the Web environment and clickstream data characteristics increase even more the general challenge. Distil the important information from the irrelevant one, deal with too much particularities and rapid changing conditions and get meaning from the data, are only a few subset of such issues. Analysts must tackle (human and not human) visitors' behavior aspects, which, in the last case (not human), skew the results and tend to be progressively more varied and difficult to distinguish. In fact, most of the previously pointed successful ingredients of the electronic commerce domain are also present in other types of Web sites or activities and viewed as truth challenges, requiring greater efficacy on WUM processes. Namely, becomes necessary to treat systematically such huge, complex and constantly growing data source, counting with hard time constraints. Decision makers across the organization demand for fast transformation of this massive data into valuable and actionable knowledge, to

orient new ways of acting and site's improvements leading to revenue. Additionally, in the specific WUM area the problem types, the kinds of mining activities, the related practical applications and the key data items are less studied and structured.

Our underlying goal is to promote a more efficient, effective, and synergetic use of the organization's resources, decreasing the effort and time required to derive useful knowledge, bringing up together multiple valuable contributions to overcome the main difficulties. The focus of our work lies on the WUM processes development challenge of selecting suitable mining methods to apply on a specific clickstream analysis problem. We gave more emphasis to the modeling phase of the WUM process, typically presuming the availability of sources containing pre-processed data, but considering also tasks of the remaining phases. Our idea is that arduous and intensive pre-processing tasks must be centralized in a previous stage, in order to make data available to all the potential analysts in more manageable forms. The primary target of our work is, precisely, the inexperienced analysts, facing problems that may be solved exploring WUM. Consequently, an implicit requirement is to support problem descriptions making use of abstractions related to the real problems to solve and, naturally, to establish direct relationships among such abstractions and the most promising DM methods and approaches.

### III. STRENGTHS AND OPPORTUNITIES OF WUM APPLICATION CASES

The most important learned lesson from 2000 KDD Cup annual competition was the crucial role played by humans in WUM processes development, even when the only interesting success criteria was accuracy or score [8]. Human insight was strategic in tasks as feature selection and construction from hundreds of available attributes and in the choice of mining methods. Indeed, most of the success obtained by experts, when dealing with WUM problems, comes from their acquired know-how. Even they cannot provide general and consistent rules to support problem solving.

Building up WUM application cases has considerable strengths, mainly realized by structuring and memorizing the knowledge acquired from the experience. Hence, we decided to document, catalogue and store WUM past experiences, in a specific oriented knowledge base that could be applied over clickstream data analysis. Examples of past successful solved problems might be the most helpful and convincing form of aid in this scope, since they may: (i) simplify the underlying complexity, providing at the same time the details of a tested and solved situation; (ii) yield context information, making possible to report the solutions along with the respective justifications and obtained discoveries; (iii) promote the mapping of the current problem, against the existent ones.

A straight reuse of WUM solutions is quite possible in this scope, since recurrent problems are common. Still, becomes necessary to enable flexible means for relating new problems and the stored ones, to help users on identifying the most plausible strategies to address the problem at hands. The

CBR paradigm brings a key opportunity to our knowledge base, providing inherently a proper way for attending this demand. CBR methods favor a flexible similarity based comparison, even if the involved features are not objective and precisely defined. CBR can cope with incomplete and subjective information and makes possible to consider only the relevant features and use specific importance levels, increasing the potential of answering the real user needs. Furthermore, CBR provides a sustained incremental learning approach, given that a new experience can be automatically integrated each time a problem is solved, becoming immediately available to apply on future problems [1]. This CBR strength is of great importance to us, due to the constant evolution of WUM and the need to incorporate knowledge about new mining algorithms, tools, types of problems, solving approaches and kinds of discoveries applications.

Defining and representing cases are also crucial issues for CBR. A case may be defined as a contextualized piece of knowledge representing an experience that teaches lessons fundamental to achieving goals [8]. In MPS, a case is a WUM process described by a set of fundamental dimensions (D, T, P, A and K) and, combining the CBR principles, structured in terms of a domain problem and the applied solution (Fig. 1). A problem is essentially defined by:

- characterizations of the available data (D), at dataset and variables level;
- categorizations of the WUM problem type (T), mainly in terms of abstractions such as main underlying activity, analysis goals and practical application areas;
- process evaluation criteria (P) (not shown in Fig. 1).

The applied solution comprises: a sequence of activities (A), including transformation and modeling stages, the involved data and the model parameter settings; prior and derived knowledge (K), concerning to facts that affected the analysis, the extracted knowledge and the relations to such facts; and general information about the WUM process (P).

We also have a context description item to organize cases in terms of a Web site's perspectives or particular sections. This item is a logic container for cases description features. The initial idea was to avoid redundancy on descriptions, since we have several datasets and common features. Though, the context provides flexibility on retaining details from different parts of one Web site. The context may be associated with some aspects of problem description, namely dataset, activity, and specific and general facts.

The most important question concerning datasets is to capture the relevant properties to the particular purpose of DM methods selection. We need a consistent characterization, in order to be able to compare dissimilar datasets. In fact, we compare the metadata, not the clickstream dataset itself. Our strategy is based on a common data characterization approach [11]. This approach has been frequently and successfully used in Meta-Learning, to select adequate learning algorithms. In general terms, we adapted this approach to clickstream data characteristics.

The cases' problem part is used to describe WUM problems and to find out previous similar cases, both defined

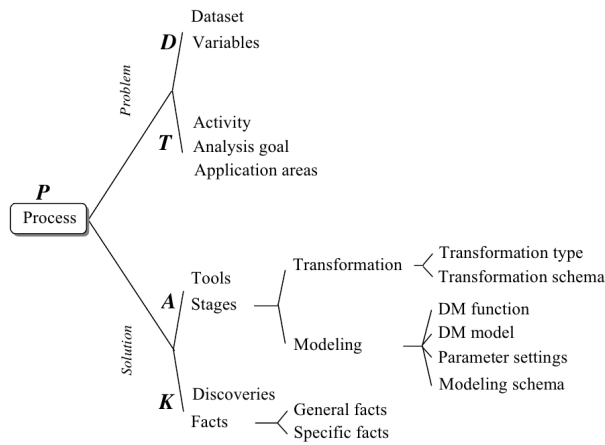


Fig. 1 WUM application case main elements

based on the common features (Table 1). The solution parts of the most similar cases retrieved are used to produce mining plans, forming the recommended solution to the submitted problem.

IV. MINING PLANS SELECTOR SYSTEM

The tasks involved in CBR have been described as a cyclical process, comprising the 4REs i.e. retrieve, reuse, revise and retain [1]. We adapted this widely acknowledge cycle to the activities to perform by the MPS system, devising six constituent steps. These steps form a problem solving and learning from experience strategy, oriented to the WUM so special application domain. Fig. 2 shows the adopted cycle. The original steps from [1] are presented, at italic, to distinguishing them from the added ones.

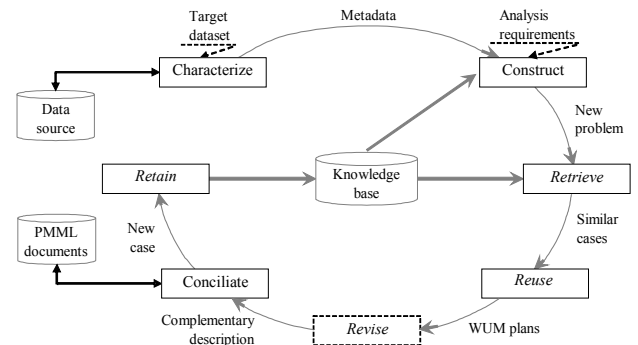


Fig. 2 Adopted CBR cycle

To solve a problem, the MPS system acts tacking as inputs the target dataset and the analysis requirements and delivers WUM plans appropriate to the current problem based on the cases kept on the knowledge base. The problem solving part of the system comprises five steps: Characterize, Construct, Retrieve, Reuse and Revise. One MPS's specific task is to characterize the target data, producing a systematic and consistent meta-representation, comprising different types of data sources. Another particular task is to construct a new WUM problem, guiding, gathering and organizing the user's explicit analysis constraints specification. The retrieve task is a typical one, being used to find out the cases most similar to the target problem. The reuse task generates WUM plans, mostly based on the mining methods and the levels of similarity of the retrieved cases, and considering also the evaluation criteria most important to the analyst. This step does not performs extensive adaptation of the solution to the current problem, namely in the wide sense intended by the original step. Nevertheless, it focus the main parts of the candidate cases that may be transferred to the target problems, by rec-

TABLE I.  
LIST OF PROBLEM DESCRIPTION FEATURES

Category	Features	Similarity measure for:		
		Single values	Set values	
<b>P</b>	Evaluation criteria	Precision, Time of reply, Interpretability, Resources requirements and Implementation simplicity	(NMDc)	
	Process date	Date	(NMD)	
<b>T</b>	Site's activity	(a set of) Activity	(SM)	(HMA)
	DM task	(a set of) Goal (a set of) Application area	(SM) (SM)	(HMA) (HMA)
<b>D</b>	Characteristics at dataset level:	-Number of lines and columns/variables -% of numeric, categorical, temporal and binary columns -Granularity (e.g. session) -Type of visitor's identification -Type of visitor's information recording -Access order and access repetition availability -Access data and hour availability	(NMD) (NMD) (E) (E) (E) (E) (E)	
	Characteristics at variable level:	(a set of) Variable: -Data type -Semantic category -Number of distinct values -Number of null values	(G) (SM) (SM) (NMD) (NMD)	(MA)

ommending mining methods instead of mere cases, preparing the reuse of the methods that constructed the solution. The revise step is accomplished outside of the system, using a KD tool.

Concerning the MPS learning perspective, the system operates accepting heterogeneous descriptions of new WUM processes and acquiring knowledge. MPS uses a semi-automated learning approach, in order to systematize and simplify such arduous activity. The steps included in learning are: *Conciliate* and *Retain*. The accepted incomings are: documents describing mining activities, generated by the KD tool in *Predictive Model Markup Language* (PMML) [15], a XML based standard to define and share statistical, and DM models across compliant applications; the process complementary description, which would be exhaustive, if the used tool does not supports the PMML standard. First, a conciliate task transforms and combines the heterogeneous descriptions items, supplied by user interaction and documents in PMML. Then the traditional retain task essentially augments the knowledge base with a new case, elaborated by integrating and structuring the incoming elements, considering the internal schema of the cases' representation.

#### V. BUILDING UP THE PRELIMINARY CASE BASE

Testing a system like MPS, specifically the problem solving point of view, requires an extensive case base. We must accept large and diverse input datasets, since clickstream data are huge and analyzed in distinct forms. More important, we need a wide set of successful and representative WUM processes, using the existent datasets. Such processes have to include different DM functions and methods and be applied to solve a comprehensive set of typical problems types.

The preliminary case base was build appealing to WUM application examples, based on real data and analyses that were available in the Internet, and some other research works. Some of these analyses reproduce WUM processes developed with such data, published together with the respective sources or in research papers. This option was made to attend the requirements previously explained, as well to overcome the issue of the discoveries success subjectivity, which in the case of simulated examples is relative and difficult to evaluate. So, we provide a greater level of success guaranty. Another advantage of this option was the greater diversity of situations.

About disadvantages, the system was not evaluated according to the previously established and idealized circumstances: pre-processed and quality data; analysis problems one of an organization. Preparing cases, in these conditions, is a more complicated and time consuming task. The analysis of each dataset is extended for longer periods, since it requires data transformation efforts and, mainly, the understanding of these data and its surrounding context. Furthermore, the case base profile changed slightly, since it concerns to more than one organization. However, first, this case base provides wide application to different kinds of organization and situations. Second, the system proved that was able to retain details from varied environments. The context

description, previously described, was very useful to deal with this new scenario.

Regarding the prepared cases main characteristics, we emphasize the diversity among the series of original data, from which the used datasets were derived. The original data vary from Web servers logs, in its rude form, to data already pre-processed, and in some cases with distinct series of data devoted for the treatment of different problems. This is precisely the situation of the 2000 KDD Cup case study [8]. One used three datasets of this source and multiple analyses based on them. In this type of situation we mainly filtered and derived new features, taking into account data quality and relevance to the problem at hands. Other datasets were pre-processed and used to generate multiple datasets, such as, for example, page view or access level clickstreams, aggregations at session level and binary matrices (e.g. sessions X accessed pages). Other known and available used examples were the *msnbc* [5] and *ECML\PKDD 2005 Discovery Challenge* [2] datasets and reported experiences, both about clickstream data analysis.

The construction of the preliminary case base provided the way to conduct experimental tests of the semi-automated learning approach of the system. This approach proved to be very useful on decreasing the efforts on processes extensive descriptions and to reduce the dependency from WUM experts. The well known datasets we mentioned are very long, being very handy to have automatic ways for capturing dataset metadata. Thus, dataset characterization was tested under demanding conditions and the respective metadata was successfully captured. Besides, usually we have processes with several stages, including each one the selection of numerous variables and the specification of several values of parameter settings. The learning approach was used, with success for all the WUM processes from which was possible to obtain PMML documents, despite the need to complement the description through explicit user interaction. We may conclude that the learning approach is effective. The problem solving part experimental results are discussed in the next section.

#### VI. RETRIEVING SIMILAR WUM PROCESSES

The retrieve step plays a vital role on problem solving. This step selects the most plausible cases to found the construction of mining plans to recommend, according to the target problem. The variants of problem description that might be submitted to the system are diverse, but may be systematized into three main types, related with the previously mentioned dimensions: oriented by the target dataset (dimension D), by other kinds of constraints (dimensions P and T) or both (dimensions D, P and T). Table 1 shows the problem description features and the measures used to assess the level of the similarity (defined on Table 2).

The similitude assessment approach devised over WUM problems comprises the modelling of the following types of measures:

- local similarity measures for simple (single-value) attributes;

- local similitude measures for structured (multiple or set value) features (namely MA and HMA measures);
- global similarity measures (G) defined through an aggregation function and a weight model.

The global similitude combines the local similarity values of several features (e.g. through a weight average function), giving an overall measure. It is applied at variable's and case's level. The local similarity measures are defined over the descriptors and depend mainly on the features domain, besides the intended semantic. Concerning simple (single-value) features, the local similitude of categorical descriptors is essentially based on exact (E) matches (e.g. for binary attributes) or is expressed in form of similarity matrices (SM), which establish each pairwise similitude level (e.g. for some symbolic descriptors). To compare numeric simple features, we adopted similarity measures mainly based on the normalized Manhattan distance.

We also need similarity measures for complex descriptors, modeled as set-value features, containing atomic values (e.g. application areas) or objects having themselves specific properties. For instance, variables have specific properties (e.g. data type) and may occur in different number for each

dataset. Indeed, these needs were the main issue faced under the similarity assessment. For instance, it appears when matching the variables from the target and each case. We have to compare two sets of variables, with inconstant and possibly distinct cardinality, where each variable has its own features. There are multiple proposals in the literature to deal with related issues (e.g. [3], [7], and [16]). Even so, we explored a number of them, for instance, the measures suggested on [7], and the comparative tests performed lead us into tailored or extended (MA and HMA) measures, better fitting our purposes, as reported in [21].

Concerning the retrieve evaluation, the general and specific tests performed so far demonstrate to the system's effectiveness. The specific tests included the comparison among distinct types of objects, such as series of variables and datasets. Regarding datasets, the system discriminates the most similar and dissimilar ones, based on the adopted features and proposed as the most relevant ones for selecting mining methods and approaches. The results conform to the intuitive notion of similarity among datasets, based on the general idea about each one. For instance, some identified trends were the following:

TABLE II.  
LIST OF MAIN USED SIMILARITY MEASURES

Description	Measure
(G) Weight average (global similarity function)	$Sim_{global}(t, c) = \frac{\sum_{f=1}^n Sim_{local}(t.f, c.f) * w_f}{\sum_{f=1}^n w_f}$
(NMD) Normalized Manhattan distance	$Sim_{Local}(t.f, c.f) = 1 - \frac{ t.f - c.f }{f_{max} - f_{min}}$
(NMDc) Normalized Manhattan distance changed	$Sim_{Local}'(t.f, c.f) = \begin{cases} 1 & c.f \geq t.f \\ 1 - \frac{ t.f - c.f }{f_{max} - f_{min}} & c.f < t.f \end{cases}$
(E) Exact (text or binary)	$Sim_{Local}(t.f, c.f) = \begin{cases} 1 & c.f = t.f \\ 0 & c.f \neq t.f \end{cases}$
(SM) Similarity matrix	
(MA) Maximums Average	$Sim_{MA}(A, B) = \frac{1}{n_A + n_B} \sum_{a \in A} \max(sim(a, b)) + \sum_{b \in B} \max(sim(a, b))$
(HMA) Half maximums Average	$Sim_{HMA}(A, B) = \frac{1}{n_A} \sum_{a \in A} \max(sim(a, b)) \quad A \subset \text{Target set}, B \subset \text{Case set}$
$t, c$ – target and case (or part of them) $t.f, c.f$ – values of each feature $f$ $Sim_{local}$ – local similarity measure $n$ – number of features $w_f$ – feature $f$ importance weighting	$f_{max}, f_{min}$ – maximum and minimum values (observed) on feature $f$ $A, B$ – two sets, such that $a \in A$ and $b \in B$ $sim(a, b)$ – similarity between each pair of elements of the two sets $n_A, n_B$ – cardinality of the sets $A$ and $B$



- When the target dataset is a binary matrix: the most similar datasets are also binary matrices; the most dissimilarity datasets are common and, frequently, datasets having access granularity.
- When the dataset has access granularity: the most similar datasets also have access granularity; the most dissimilar datasets are usually the same.
- When the target data set has session or other granularity (e.g. visitor) and is not a binary matrix: there is not a simple and strait similarity pattern (justified by the variety of attributes gathered at these levels); the most dissimilar datasets have mainly access granularity.

In terms of general tests, the remain descriptors also reflect influent factors and affect cases retrieving, contributing for establishing the bridge between analysis requirements and suited mining methods and approaches. The system relates WUM processes based on similar intentions and applications, but not necessarily coincident. Since the data characterization descriptors are in majority, within the problem description, by default their relative importance is greater and the system tends to select processes based on similar datasets. This default behaviour accords to the intended one and is considered a good result. In fact, the dataset characteristics are always a crucial (predictive) factor, since models properties and assumptions, and even other factors (e.g. goals), frequently, demand for some specific data. Furthermore, the system provides means to change the default behaviour and to improve the problem specification, namely exact filtering criteria, specific descriptors importance levels and the exclusion of irrelevant (or unknown) descriptors.

## VII. CONCLUSIONS

Rapidly changing conditions and the global competition have brought tremendous pressure into organizations way of life, demanding an effective presence on the Web and a more responsive and proactive attitude to realize its full potential. WUM is one crucial tool to bridge the gap between massive clickstream data and actionable knowledge, in order to devise Web site's opportune enhancements. However, WUM learning curve is a serious obstacle to inexperienced users, being pertinent to have a strategy showing the way how to proceed.

The proposed and developed work aims at promoting a more efficient, effective and synergetic exploration of WUM, decreasing the effort and time required to derive useful knowledge from clickstream data. To achieve this aim we designed, developed and implemented a prototype of a CBR system, specifically devoted to assist users on WUM processes, mainly on selecting proper mining methods and approaches to address analysis problems. The system also provides support to users on documenting and organizing the knowledge gained from the experience on solving new WUM problems, through a semi-automatic learning approach. The previous collected and stored WUM application cases are therefore the base that sustains the recommendation of mining plans to solve new problems.

We believe that the MPS system is a good tool for knowledge creation, sharing and reuse. The system is based on abstractions related to the real problems to solve, meaning that it could serve the particular needs of less skilled users, wishing to learn how to handle a concrete problem, being also useful to specialists interested in reusing successful solutions, instead of solving the problems from scratch. Data is always growing and is increasingly stored by organizations. DM tools are gaining more importance and KD processes are becoming more useful and widespread, although they remaining complex.

In this paper we described our system, focusing the retrieving of similar cases and its evaluation using a preliminary case base. These prepared cases reproduce WUM exercises descriptions publicly available, overcoming the need of a wide set of examples and the issue of the discoveries success subjectivity. The used datasets and the reproduced WUM processes are challenging and real applications of WUM, proving demanding conditions to test the MPS system. The cases contain some diversity of circumstances, which is beneficial to sustain the construction of a repository of this nature. The system's evaluation appealing to this preliminary case base also points to the system's effectiveness. A drawback to point out is the intentional generality of some abstractions used to categorize problems (e.g. analysis goals and application areas), which restricted their diversity. The potential of the approach has not been completely explored, since greater levels of abstraction might be achieved, enlarging the case base and developing further such categorizations.

For the future we plan to further evaluate the current implementation. This will be realized through the preparation of more cases and, particularly, within the context of a study case, based on a concrete target organization.

## REFERENCES

- [1] A. Aamodt and E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations and Systems Approaches" in *Artificial Intelligence Communications (AICOM)*, IOS Press, vol. 7, no 1, pp. 39-59, 1994.
- [2] *ECMLPKDD 2005* conference web site. <http://www.liaad.up.pt/~ecmlpkdd05/>. Access June 2011.
- [3] T. Eiter and H. Mannila, "Distance Measures for Point Sets and their Computation", *Acta Informatica*, vol. 34, no 2, pp. 109-133, 1997.
- [4] U. Fayyad, G. Piatetsky-Shapiro and P. Smyth, "The KDD Process for Extracting Useful Knowledge from Volumes of Data", *Communications of the ACM*, vol. 39, no 11, pp. 27-41, 1996.
- [5] S. Hettich and S. D. Bay, the UCI KDD Archive [<http://kdd.ics.uci.edu>]. Irvine, CA: University of California, Department of Information and Computer Science, 1999.
- [6] M. Hilario and A. Kalousis, "Fusion of Meta-Knowledge and Meta-Data for Case-Based Model Selection", in *Proc. of the 5th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD '2001)*, Springer pp. 180-191, 2001.
- [7] M. Hilario and A. Kalousis, "Representational Issues in Meta-Learning", in *Proc. of the 20th International Conf. on Machine Learning (ICML '03)*, AAAI Press, pp. 313-320, 2003.

- [8] R. Kohavi, C. Brodley, B. Frasca, L. Mason, and Z. Zheng "KDD-Cup 2000 Organizers' Report: Peeling the Onion", SIGKDD Explorations, vol. 2 no 2, pp. 86-98, 2000.
- [9] R. Kohavi and F. Provost, "Applications of Data Mining to E-commerce", (editorial), Special issue of the International Journal on Data Mining and Knowledge Discovery, 2001.
- [10] J. Kolodner, "Case-Based Reasoning", Morgan Kaufman, San Francisco, CA, 1993.
- [11] C. Lindner and R. Studer, "AST: Support for Algorithm Selection with a CBR Approach", in Proc. of the 3rd European Conference on Principles of Data Mining and Knowledge Discovery (PKDD'1999), Springer, pp. 418-423, 1999.
- [12] MetaL project <http://www.metal-kdd.org/> Access June 2011.
- [13] K. Morik and M. Scholz "The MiningMart Approach to Knowledge Discovery in Databases", in Intelligent Technologies for Information Analysis, Springer, 2004.
- [14] B. Mobasher, "Data Mining for Web Personalization", lecture Notes in Computer Science, 4321, 90-135, 2006.
- [15] Predictive Model Markup Language. Data Mining Group. <http://www.dmg.org/index.html>. Access June 2011.
- [16] J. Ramon "Clustering and Instance Based Learning in First Order logic" PhD thesis, K.U. Leuven, Belgium, 2002.
- [17] C. Riesbeck and R. Schank, "Inside Case-based reasoning", Lawrence Erlbaum, 1989.
- [18] R. Schank, "Dynamic Memory: A theory of learning in computers and people", Cambridge University Press, 1982.
- [19] J. Srivastava, R. Cooley, M. Deshpande and P.-N. Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data", in SIGKDD Explorations, vol. 1, no 2, pp. 1-12, 2000.
- [20] J. Srivastava, P. Desikan and V. Kumar, "Web Mining - Accomplishments and Future Directions", invited paper in National Science Foundation Workshop on Next Generation Data Mining, Baltimore, MD, 2002.
- [21] C. Wanzeller and O. Belo, "Similarity Assessment in a CBR Application for Clickstream Data Mining Plans Selection" in Proc. of the 9th International Conference on Enterprise Information Systems (ICEIS' 2007), Funchal, Madeira, Portugal, 2007.

# Problem of website structure discovery and quality valuation

Dmitrij Żatuchin

Institute of Informatics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27,  
50-370 Wrocław, Poland,

Email: Dmitrij.Zatuchin@pwr.wroc.pl

**Abstract**—Navigation as a part of an interface was always an important issue of a design process. Because information architecture and the navigation of current websites are very complex, especially of e-commerce websites or information portals, it is very hard to analyze or redesign a structure in a manual way. In order to solve the problem of automation of website structure analysis, there should be defined its model. Also, during the study of a subject it was found, that there is a lack of a quality estimator, which allows to valuate in a vary moments the quality of the structure. Observation of a structure quality gives possibility to analyze and decide when the structure should be changed basing on decision rules or calculated thresholds for analyzed amount of time.

The main aim of this study is to describe a model for website structure representation, derive the quality estimator, define and solve the problem of website structure discovery and quality valuation utilizing the proposed metric.

Finally, experiment with utilization of proposed methods is presented.

## I. INTRODUCTION

THE main task of the website is to provide the content and functionalities of the system. Such functionalities usually are placed to the sub-pages of the website. The main task of the navigation is to service efficiently and effectively users' requests, which are provided through these functionalities. Website usability is the measure of a success which users experience while interacting with the system. This is the extent to which users can achieve the desired objectives during their visit. Some of the website usability factors include: the compatibility of the site layout, ease of use of a search engine, adequate links that provide instant access to information, a site map which serves as a table of content for the whole site, legible fonts and appropriate use of colors to highlight and organize information or functionalities. All these factors contribute to the ease of use and make a visit on the website useful and enjoyable [1]. In 2001 Donahue [2] pointed, that difficult navigation with the limited flexibility constitutes the major problem of the usability. Therefore, a good solution for the navigation should be provided [3]. According to the National Institute of Standards and Technology [4] ease of navigation is essential for users at all lev-

*The research in this paper has been partially supported by: European Union in the scope of the European Regional Development Fund program no. POIG.01.03.01-00-008/08 and European Social Fund – fellowship “Młoda Kadra”.*

els of proficiency in using computers in order to navigate and obtain the desired information on the websites. It can be stated, that the number of planned functionalities at the design stage influences on the final number of pages and links [5]. Also it may be stated, that the increase of the quality of the interface is possible by increasing the quality of the structure of the navigation.

Users may come to the website in many ways: through the main page, as a result of the reference link from other websites, from the search results or an advertisement. Moreover, users have different goals and their objectives are very diverse. Navigation should support those differences through a variety of solutions: a hierarchical, task-oriented, chronological, alphabetical, and based on the popularity of information architecture [1]. It is required in order to avoid situation, that user will be trapped inside one page or reach the orphaned page [6]. On the static website or dynamic one with a limited number of links the navigation structure has a large impact on the quality of delivered content. Therefore, evaluation and improvement of the website structure becomes a key issue what repeatedly was underlined by researchers and experts of Human-Computer Interaction field of studies [7], [8], [9], [10], [11], [12], [13], [14], [15].

## II. WEBSITE STRUCTURE DISCOVERY

For the problem of quality valuation of the website structure there are some estimators in the literature of subject. Unfortunately, none of them treat the structure as a network and take into account such information as connections between pages, popularity of single pages, their position or utilization of edges connecting these pages among the website. Few proposed [16], [17] treat the website as a tree and in order to estimate aptitude of a structure they utilize knowledge such as click number into a page and distance from a root item. What is important is that distance from a root items is treated like in height in tree structures, what is not true for the website.

The motivation for solving the problem of a structure discovery with maximum data utilization and the problem of a structure quality measurement comes from a need to increase the efficiency of the interface redesign process. It can be reached by including into the process an automatic analysis of usage data and generating recommendations to change

existing website structures into modified ones – adapted to users' needs.

#### A. Model of a website structure

The mere idea of using graphs to model the structure of the website is widely used by practitioners and researchers. First, Perkowitz [18] used to describe links on a single webpage with the graph structures. In Garofalakis' studies [19] model of a website was described as a tree, because of optimization simplification. Yen et al. [20] defined the environment consisting of three layers of evaluation and expansion of website projects using graphs for modeling. In 2008 Yang defined the conceptual model for the structure [21] thus using ontological paradigm. Another way to model the website is to use Markov chains [22]. Also there can be found works inspired by PageRank algorithm [23], where web mining is used in order to optimize website structure [24].

The proposed model of a structure is understood as a set of unique pages which may be reached with an internal linking through navigation elements (Fig. 1b). The structure is mapped by a set of nodes and connections (Fig. 1a,c). The first node in a structure is called a root node, which is accessible through the default domain or IP address in a browser and returns the main page of a website. A navigation element may occur on one or many pages, but each time it may consist of a different set of pages and links. Connections between pages may be directed or undirected (Fig. 1c).

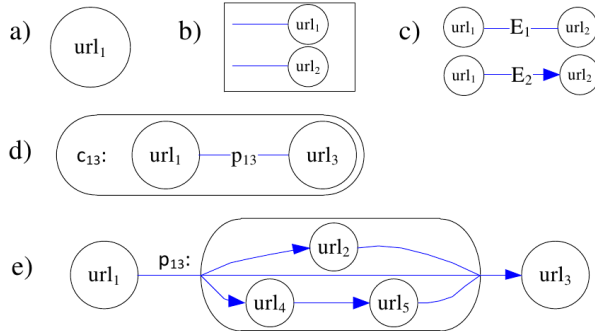


Fig. 1. Elements of website structure: a) a subsite, b) a navigation element, c) connection between two sites, d) path, e) connection.

Such defined website  $l$  is modeled as a graph structure in equation (1) of  $N$  nodes (pages) and  $M$  edges (links), where  $URL_l$  is a finite set of pages (1) which are in one domain in the range of  $l$  website and  $E_l$  (1) is a finite set of links which does not contain loops or reverse connections, where  $e_{li}$  is an edge from  $url_{lm}$  to  $url_{ln}$  to .

$$G_l = (URL_l, E_l, P_l, C_l), \quad URL_l = \{url_{li} \mid i \in [1, N]\} \quad (1)$$

$$E_l = \{e_{li} \mid e(m, n), m \neq n, m, n \in [1, N], i \in [1, M]\}$$

Basic elements of a structure model paths  $P$  (Fig. 1d) and connections  $C$  (Fig. 1e).

Path is a set of edges, which are the passage from one node to another. Paths are directed – if between nodes there is one

edge, or transitive – if on a path there is more than one intermediary node. Set of paths  $P_l$  is defined in (2).

$$P_l = \{p_{lij}, i, j \in [1, N] \mid \{e_{li} < e_{lj} \mid \exists p_{liz} \wedge \exists p_{lzz}, z \in [1, N]\}\} \quad (2)$$

A connection between two nodes exists if it is possible to pass from one node to another. The set  $C_l$  contains all connections in a graph, see equation (3).

$$C_l = \{c_{lij}, i, j \in [1, N] \mid \forall c_{lij} \exists p_{lij} \rightarrow url_{li} < url_{lj}\} \quad (3)$$

The length of from to is an average length of all paths between nodes (4).

$$D(c_{lij}) = \overline{p_{lij}} \quad (4)$$

Nodes and edges have additional characteristics and indicators, which will be used in a proposal of formulation of a quality estimator.

#### B. Node characteristics and indicators

Every node in the structure has a distance from the root node, which is calculated according to the distance function in equation (5). For the root node the distance equals 1. Pages are usually visited by users in such way, that nodes on lower levels are rarely visited. The form of equation was derived as a result of analysis of statistics gathered by Google Analytics of approximately 500 000 different users visiting 12 websites from different categories and left by them paths of visits.

$$d(url_{li}) = pow(\min|P(url_{l1}, url_{li})| + 1, \frac{1}{\sqrt{2}}) \quad (5)$$

Another characteristic for a node is *input-output* defined in equation (6), which is the sum of number of nodes, which edges point to the processed node and number of outgoing edges from the processed node (Fig. 2a).

$$io(url_{li}) = |\{e_{lj} \mid \exists e(i, n) \vee \exists e(m, i); i \in [1, N], j \in [1, M]\}| \quad (6)$$

Node indicators determine the characteristics based on usage data. These are:

- $OccU(url_{li}, [t, t + \tau_k])$  – specifies the number of edges used in all paths of all users of the website during the interval of time  $[t, t + \tau_k]$ ;
- $PopU(url_{li}, [t, t + \tau_k])$  – popularity of a node relatively to the entire structure of the website at the interval of time  $[t, t + \tau_k]$ , defined as equation (7);

$$PopU(url_{li}, [t, t + \tau_k]) = \frac{OccU_k(url_{li})}{N_l} \quad (7)$$

- $Acc(url_{li})$  – availability of a node in the structure defined in (8), is an ease for  $url_{li}$  to be visited by users and depends on the distance from the root and  $io$  characteristic;

$$Acc(url_{li}) = \frac{io(url_{li})}{d(url_{li})} \quad (8)$$

#### C. Edge characteristics and indicators

Characteristics for edges are:

- $Out(e_{ij})$  – specifies the number of all edges that come from the same node as the beginning of the edge. This indicator is illustrated in Fig. 2b and is defined as (9);

$$out(e_{ij}) = |\{e_{ij} | parent(e_{ij}) = url_{lm} \wedge parent(e_{ij}) = url_{lm}\}| \quad (9)$$

- $Reach(e_{ij})$  characteristic defined as (10) returns the number of nodes that can be achieved through following the edge (Fig. 2c). If a node, which indicates an edge, is a leaf, then the value of the characteristic equals 0.

$$reach(e_{ij}) = |\{url_{lj} | e_{ij} \in p_{lxj}; x, j \in [1, N]\}| \quad (10)$$

Indicators are defined as follow:

- $OccE(e_{ij}, [t, t+\tau_k])$  – specifies the number of occurrences of edges in all paths in the interval of time  $[t, t+\tau_k]$ ;
- The indicator  $r$  in (11) is an information about traffic between two nodes;

$$r(m, n) = \sum_{i=1}^{|p_{lij}|} OccE(e_{ij}, [t, t+\tau_k]), e_{ij} \in p_{lij} \quad (11)$$

- $PopE(e_{ij}, [t, t+\tau_k])$  is the edge's popularity (12) – a number of occurrence of an edge in all paths obtained from usage data in the interval of time  $[t, t+\tau_k]$ ;

$$PopE(e_{ij}, [t, t+\tau_k]) = \frac{OccE(e_{ij}, [t, t+\tau_k])}{|P_l|} \quad (12)$$

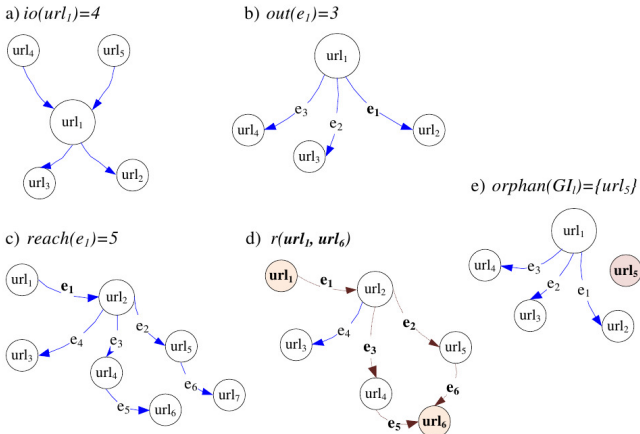


Fig. 2. Illustration of some characteristics of a model in a website structure.

#### D. Problem of a website structure discovery

Structure discovery is a required procedure for quality valuation task.

**For given:** URL address of the  $l$  website; usage data collected for the website  $l$ ; time  $k$ ;

**Determine:** website structure based on navigation patterns modeled as a graph  $GI_k$ .

There are several limitations: orphan pages (Fig. 1e) must be excluded; all internal content resources are excluded (i.e. mp3, avi, flv, swf etc. files), non-significant nodes in folders (e.g. /css, /media, /js, /admin) are filtered; limited response time for server; threshold of usage for every node (i.e. less than 1.5%) are ignored.

The proposed solution combines two approaches found in the literature:

- The structure discovered on the basis of usage. Such approach includes such nodes and edges, which are present in navigation patterns.
- The website crawled with a robot.

The structure is fully discovered after the results of two approaches are merged. It will effect with the structure containing detail information of both the navigation and the usage of nodes and edges. The solution of this task consists of three subtasks:

- The scanning task of structure;
- The process of exclusion of orphan nodes;
- The task of sampling and applying data obtained from statistics module.

An algorithm for website discovery is proposed (Fig.3). The scanning task is solved by using the method of multithreaded searching by a limited number of crawlers. It is assumed that the tested website has a system for monitoring the usage data. Such an external system through a connector does exchange data with the website.

Orphan nodes in the structure of the website are frequently the result of a human error. These are not nodes intently hidden from the user in the navigation, because such a node is discovered during obtaining the usage data.

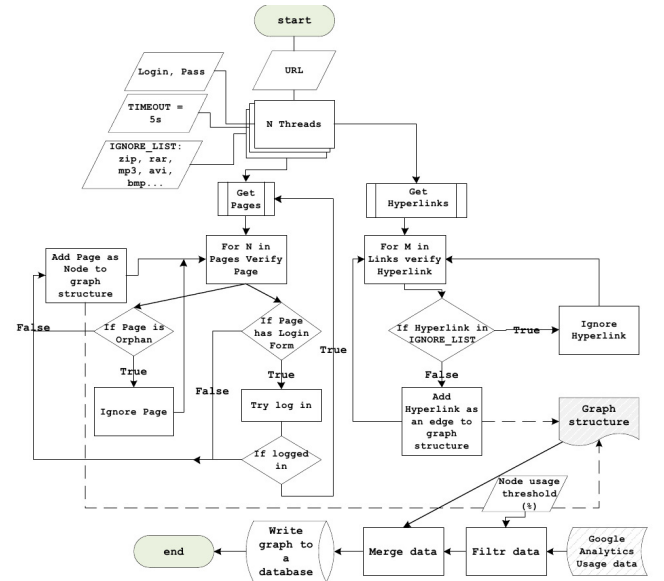


Fig. 3. An algorithm for discovering the website structure.

The node may contain links to external websites, but there is no reference to it from the navigation. Orphan exclusion is done in order to reduce the number of nodes that adversely affect the value of the quality estimator of the structure. Finding orphan nodes may help to understand which pages have not been connected with the others. This task is solved by using a recursive analysis of all the pages that contain at least one working internal link in order to detect all isolated nodes.



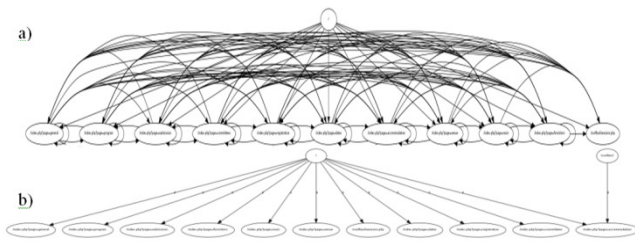


Fig. 4. A conference website [icss.pwr.wroc.pl](http://icss.pwr.wroc.pl), 13 nodes and 124 edges. a) Graph structure before MST processing; b) after processing with MST algorithm.

### E. Examples of discovered website structures

Using the proposed method of website representation it is possible to map simple structure with no connections between nodes. Such structure is constructed as a tree (Fig.4b, Fig.6). Because most of websites are graphs, in order to simplify the structure for analysis in terms of distribution of information in the nodes of the graph, a Minimum Spanning Tree (MST) algorithm is applied.

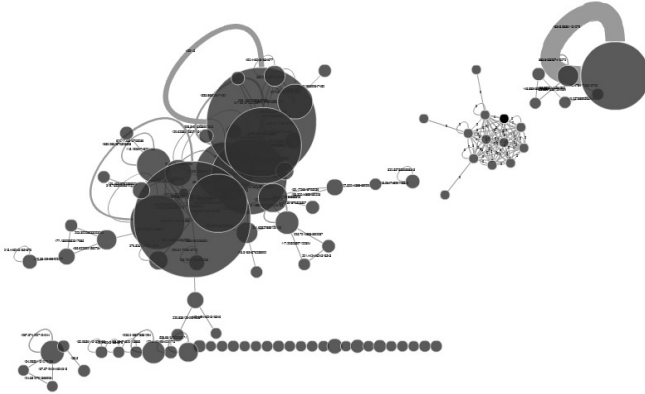


Fig. 5. Student's portal [edukacja.pwr.wroc.pl](http://edukacja.pwr.wroc.pl), 41 nodes, 142 edges with applied usage data discovered between 2010-08-01 and 2011-05-05.

Structures of e-commerce websites are very complex, manual modification of the structure is done by an information architect and it takes long time to process it manually. Such a structure can be optimized using only the recommendations or intuitive knowledge. Application of MST is used to solve this problem. Developed method of structure discovery is regulated with a filtering parameter which allows filtering nodes to address name or percent of usage. It is reasonable to set usage data parameter to a high value ( $>2\%$ ) for complex structures.

The spectrum of possible application of described method is shown on figures (Fig.4ab, Fig.5, Fig.6, Fig.7).

### III. PROBLEM OF QUALITY VALUATION OF WEBSITE STRUCTURE

The problem is: for given graph  $GI_t$  and history usage data in the interval of time  $[t, t+k]$  determine the quality of the website structure. Note that for various periods of time  $\tau$  the value of the quality estimator will be different. Assuming that the location of each node and the way of connections

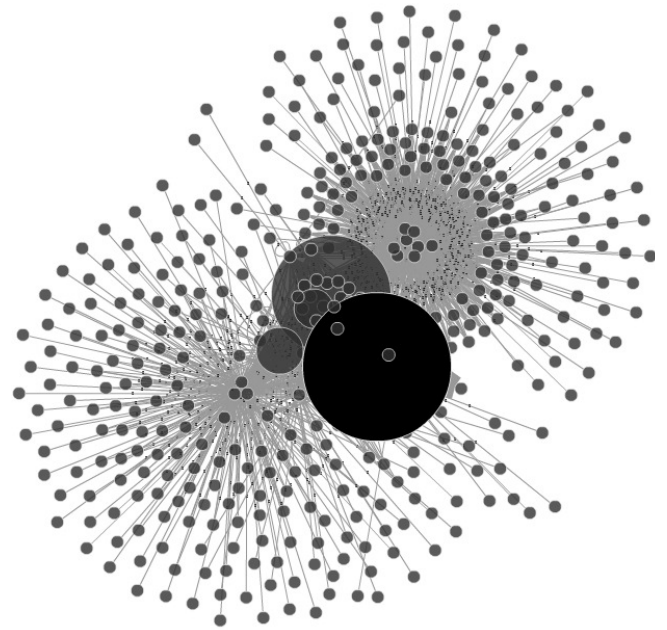


Fig. 6. E-commerce website structure with 638 nodes with usage data discovered between 2009-05-05 and 2011-05-05.

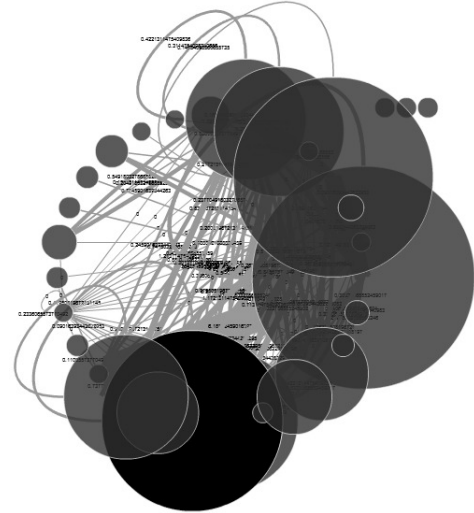


Fig. 7. Hair beauty website structure with 30 nodes and 177 edges and usage data discovered between 2010-10-01 and 2011-06-01.

between them designates how users operate the website, the quality estimator should be dependent on the characteristics of the nodes and edges. Therefore there are defined quality criteria: number of node usage; number of edge usage; node position in the structure; the importance of a node; the importance of an edge.

Usage data results with information whether the originally designed structure is suitable for users or not. The quality measure should include popularity of nodes and edges. Therefore indicators of nodes and edges as defined in Section 2.B and 2.C are applied to develop the components of an estimator. Therefore, the node influence on the structure is defined as (13) and an indicator of the connectivity degree of an edge is defined as (14).

$$ImpU_k(url_{li}) = PopU_k(url_{li}) \cdot Acc(url_{li}) \quad (13)$$

$$CD_k(e_{li}) = PopE_k(e_{li}) \cdot \frac{reach(e_{li})/(N-1)}{out(e_{li})} \quad (14)$$

The quality estimator of the whole structure will be the sum of all values  $ImpU$  of the nodes proportional to the sum of edges values of  $CD$  and will be called Graph Energy and defined as (15).

$$En_k(GI) = \sum_{n=1}^N ImpU_k(url_{ln}) \cdot \sum_{m=1}^M CD_k(e_{lm}) \quad (15)$$

For the bare structure of the website it will be equal to 0 until the usage data will be applied. As the number of users during observed interval of time  $\tau_k$  may vary and thus generate different traffic within nodes and edges, characteristics of the graph in time  $\tau_{k+1}$  should be normalized proportionally to the value in  $\tau_k$  period.

#### A. Complexity of a website structure discovery algorithm

The initial method in order to discover all connections within graph has a complexity of  $O(N!)$  and is a NP-hard, therefore the proposed method of website discovery explores only the paths which are actually used by users in selected interval of time. Computational complexity of method consists of the execution of subtasks:

- Scanning task (Fig. 3) is of  $O(k(N+M))$ , where  $k$  is the maximum depth of the graph.
- Usage data extraction is of  $O(N+M)$  complexity, because of need to obtain data of every node and every edge.
- Calculation of the graph characteristics:  $io$  has  $O(N)$ ,  $out$  is of  $O(M)$  complexity,  $reach$  has  $O(M)$ ,  $d$  is of  $O(M+N \cdot \log N)$ .

The total complexity of the quality estimation after simplification is  $O(N \log N + M)$ . The crucial for this method is number of nodes in the structure.

#### B. Experiments

To test proposed methods different websites were analyzed and monitored. In Fig.8 there are two of them – the conference website ([www.icss.pwr.wroc.pl](http://www.icss.pwr.wroc.pl)) and the website of hair beauty salon ([www.salonczare.pl](http://www.salonczare.pl)). For both, the structure was scanned (Fig. 4a, Fig.7), then usage data was recorded and merged with the graph discovered by the crawler. The sampling of the Graph Energy was done daily (Fig.8), so the tendency could be better observed. There are two lines on Fig.8 – blue stands for the equation (15), and orange is a modified equation (15) with changed operation between  $ImpU$  and  $CD$  to the sum. Orange is less resistant for small changes, and blue is appropriate for long-term observations.

For the conference website, there were periods, where energy of the graph was lower or higher than average, and they are considered as potential points of change in the website structure, especially concerning the schedule of conference. Intuitively it is understood that the conference's structure should change in different periods of time i.e. paper submission, accommodation page before the conference start. As

the graph energy is higher, the structure of the website is utilized better by users and they may reach goals more efficiently. Observation of changes in the quality of the structure contributes to detection of the moments in which there are derogations from certain value  $En(\tau_k)$ . For the hair beauty website the lower line shows the  $En(\tau_k)$  (Fig.8b) and is inside lower and higher limit all the time with only twice short alarms detected. For the high and low limit Shewhart control chart [25] was utilized and for observing tendencies and making the decision of structure change Nine Nelson Rules [26] were applied.

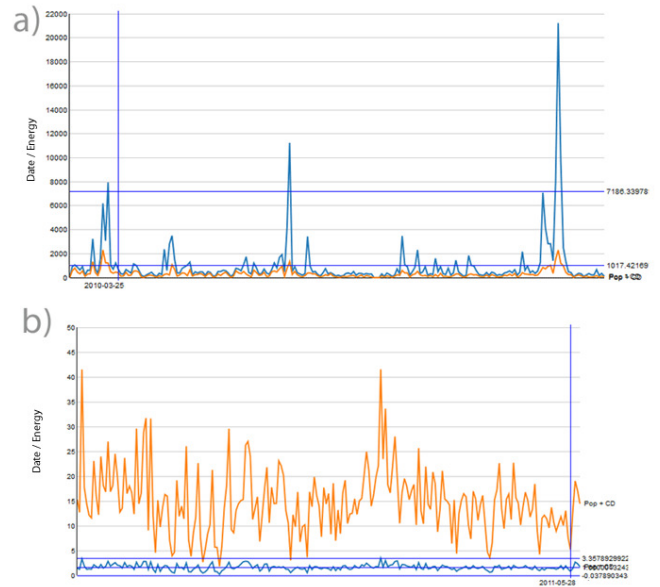


Fig. 8. Graph energy diagram of:

a) a conference website, b) a hair beauty salon website.

In order to compare results of two websites from the same categories (i.e. two beauty salons) the normalization of edge's and node's indicators should be done. More data on experiment and framework can be found in [27].

#### IV. SUMMARY

The main aim of this study was to state and propose a solution for structure discovering problem and quality valuation problem. A website structure quality estimator gives the ability to evaluate a website's navigation conformity to the way of how real users do use the website after its release to the general public. Besides consistency and repeatability, automation provides increased cost benefits to the developer. It improves the website redesign process and saves time.

In the future research there will be concerned following tasks. First, the problem of website structure optimization will be stated and the usage of quality metric based on Graph Energy will be proposed. Second, the problem of usage change detection will be stated and algorithms for it will be proposed. Third, the experimental framework will be upgraded with the new methods. Forth, the solution for discovering groups of users and recommending adapted structures to them individually will be proposed. The solution of all

discussed tasks will be an important step forward in studies on automated analysis, optimization problem and synthesis of website structures.

#### REFERENCES

- [1] Douglas K. Van Duyne, James A. Landay, and Jason I. Hong, *The Design of Sites*. Upper Saddle River, NJ: Pearson Education, Inc., 2007.
- [2] George M. Donahue, "Usability and the Bottom Line" *IEEE Software* 18 Issue 1, pp. 31-37, 2001.
- [3] James Kalbach, *Designing Web Navigation: Optimizing the User Experience*, 1st ed.: O'Reilly Media, 2007.
- [4] National Institute of Standards & Technology. (2002, May) *WebSAT Evaluation Rules*. [Online]. [http://zing.ncsl.nist.gov/WebTools/WebSAT/websat\\_rules.html](http://zing.ncsl.nist.gov/WebTools/WebSAT/websat_rules.html)
- [5] Witold Suryn, "Software Quality Engineering: The Leverage for Gaining Maturity" in *Maturing Usability*.: Springer-Verlag London Limited, 2008, vol. 1, pp. 33-55.
- [6] Sanjay J. Koyani, Robert W. Bailey, and Janice R. Nal, *The Research-Based Web Design & Usability Guidelines*: U.S. Department of Health and Human Services, 2006.
- [7] Dave Gehrke and Efraim Turban, "Determinants of Successful Website Design: Relative Importance and Recommendations for Effectiveness" in *Thirty-second Annual Hawaii International Conference on System Sciences-Volume 5*, Maui, Hawaii, 1999, p. 5042.
- [8] Jakob Nielsen, *Designing Web usability: The practice of simplicity*. Indianapolis: New Riders Publishing, 1999.
- [9] Ping Zhang and Gisela M. von Dran, "User Expectations and Rankings of Quality Factors" *International Journal of Electronic Commerce* Vol.6 No.2, pp. 9-33, 2002.
- [10] Raquel Benbunan-Fich, "Using protocol analysis to evaluate the usability of a commercial web site" *Journal Information and Management* Vol. 39 Issue 2, 2001.
- [11] Janna B. Arney and Paul J. Lazarony, "An Inclusive Guide To Assessing Web Site Effectiveness" *Journal of College Teaching & Learning* Vol.2, Number 1, pp. 27-36, 2005.
- [12] Jinwoo Kim, Jungwon Lee, Kwanghee Han, and Moonkyu Lee, "Businesses as Buildings: Metrics for the Architectural Quality of Internet Businesses" *Information Systems Research* Vol. 13 No.3, pp. 239-254, 2002.
- [13] Johnathan W. Palmer, "Web site usability, design, and performance metrics" *Information Systems Research* Vol.13 No.2, pp. 151-168, June 2002.
- [14] Layla Hasan and Emad Abuelrub, "Assessing the Quality of Web Sites" *INFOCOMP Journal of Computer Science* Vol.7 No.4, 2008.
- [15] Horton S. Lynch P.J., *Web style guide: basic design principles for creating Web sites*. NJ: Yale University Press, 2009. [Online]. <http://info.med.yale.edu/caim/manual>
- [16] G. Sreedhar, A. A. Chari, and V. V. Venkata Ramana, "Measuring Quality Of Web Site Navigation" *Journal of Theoretical and Applied Information Technology*, pp. 80-86, 2010.
- [17] Wen-long Lin and Ye-zheng Liu, "A Novel Website Structure Optimization Model for More Effective Web" in *Workshop on Knowledge Discovery and Data Mining*, Adelaide, 2008, pp. 36-41.
- [18] Mike Perkowitz and Oren Entzoni, "Adaptive Sites: Automatically Learning from User Access Patterns" Washington, Technical report UW-CSE-97-03-01 1997.
- [19] John Garofalakis, Panagiotis Kappos, and Dimitris Mourloukos, "Web Site Optimization Using Page Popularity" *Web Software*, pp. 22-29, July-August 1999.
- [20] Benjamin Yen, Paul Hu, and May Wang, "Toward an analytical approach for effective Web site design: A framework for modeling, evaluation and enhancement" *Electronic Commerce Research and Applications* 6, pp. 159-170, 2007.
- [21] Sheng-Yuan Yang, "An ontological website models-supported search agent for web services" *Expert Systems with Applications* 35, pp. 2056-2073, 2008.
- [22] Nicoleta David and Liviu Stelian Begu, "A Website Structure Optimization Model" in *ACS'10 Proceedings of the 10th WSEAS international conference on Applied computer science*, Iwate, 2010, pp. 426-429.
- [23] Serge Brin and Larry Page, "The anatomy of a large-scale hypertextual Web search engine" in *Proceedings of the VII International World Wide Web Conference*, in: *Computer Networks and ISDN Systems* vol. 30, 1998, pp. 107-117.
- [24] Jonathan Jeffrey, Peter Karski, Björn Lohrmann, Keivan Kianmehr, and Reda Alhajj, "Optimizing Web Structures Using Web Mining Techniques" in *Intelligent Data Engineering and Automated Learning - IDEAL 2007*, vol. 4881, Birmingham, 2007, pp. 653-662.
- [25] Michele Basseville and Igor V. Nikiforov, *Detection of abrupt changes: Theory and Application*. Englewood Cliffs, N.J.: Prentice-Hall, 1993.
- [26] Lloyd S. Nelson, "Technical Aids," *Journal of Quality Technology*, vol. 16, no. 4, pp. 238-239, 1984.
- [27] Dmitrij Żatuchin, "Webgraph - system do analizy i syntezy struktur serwisów www" *Interfejs użytkownika - Kansei w praktyce*, pp. 72-87, Warsaw, June 2011.



# International Workshop on Artificial Intelligence in Medical Applications

**T**HE WORKSHOP on Artificial Intelligence in Medical Applications – AIMA'2011 – provides an interdisciplinary forum for researchers and developers to present and discuss latest advances in research work as well as prototyped or fielded systems of Artificial Intelligence in the wide and heterogeneous field of medicine, health care and surgery. The workshop covers the whole range of theoretical and practical aspects, technologies and systems based on Artificial Intelligence in the medical domain and aims to bring together specialists for exchanging ideas and promote fruitful discussions.

The topics of interest include, but are not limited to:

- Artificial Intelligence Techniques in Health Sciences
- Knowledge Management of Medical Data
- Data Mining and Knowledge Discovery in Medicine
- Health Care Information Systems
- Clinical Information Systems
- Agent Oriented Techniques in Medicine
- Medical Image Processing and Techniques
- Medical Expert Systems
- Diagnoses and Therapy Support Systems
- Biomedical Applications
- Applications of AI in Health Care and Surgery Systems
- Machine Learning-based Medical Systems
- Medical Data- and Knowledge Bases
- Neural Networks in Medicine
- Ontology and Medical Information
- Social Aspects of AI in Medicine
- Medical Signal and Image Processing and Techniques
- Ambient Intelligence and Pervasive Computing in Medicine and Health Care

## PROGRAM COMMITTEE

- Jan Bazan**, University of Rzeszów, Poland  
**Leontios J. Hadjileontiadis**, Aristotle University of Thessaloniki, Greece  
**Hugo Gamboa**, New University of Lisbon, Portugal  
**Florin Gorunescu**, University of Medicine and Pharmacy of Craiova, Romania  
**Jerzy W. Grzymala-Busse**, University of Kansas, USA  
**Aboul Ella Hassanien**, Cairo University, Egypt  
**Yutaka Hata**, Graduate School of Engineering, University of Hyogo, Japan  
**Zdzisław S. Hippe**, University of Information Technology and Management in Rzeszów, Poland  
**Juliusz L. Kulikowski**, Polish Academy of Sciences, Poland  
**Lukasz Piątek**, University of Information Technology and Management in Rzeszów, Poland  
**Thomas Schrader**, University of Applied Sciences in Brandenburg, Germany  
**Andrzej S. Sluzek**, Nanyang Technological University, Singapore  
**Andrzej Skowron**, University of Warsaw, Poland  
**Paweł Strumiłło**, Technical University of Łódź, Poland  
**Ingo Timm**, University of Frankfurt, Germany  
**Daming Wei**, Hangzhou Dianzi University, China  
**Zygmunt Wróbel**, University of Silesia, Poland  
**Jerzy Wtorek**, Gdańsk University of Technology, Poland

## ORGANIZING COMMITTEE

- Krzysztof Pancierz** (Chair), University of Information Technology and Management in Rzeszów, Poland



# Identification of Patient Deterioration in Vital-Sign Data using One-Class Support Vector Machines

Lei Clifton\*, David A. Clifton\*, Peter J. Watkinson†, and Lionel Tarassenko\*

\*Institute of Biomedical Engineering, Department of Engineering Science, University of Oxford, Oxford, UK  
 {lei.clifton, david.clifton, lionel.tarassenko}@eng.ox.ac.uk

†Nuffield Department of Anaesthetics, University of Oxford, Oxford, UK

**Abstract**—Adverse hospital patient outcomes due to deterioration are often preceded by periods of physiological deterioration that is evident in the vital signs, such as heart rate, respiratory rate, etc. Clinical practice currently relies on periodic, manual observation of vital signs, which typically occurs every 2-to-4 hours in most hospital wards, and so patient deterioration may go unidentified. While continuous patient monitoring systems exist for those patients who are confined to a hospital bed, the false alarm rate of conventional systems is typically so high that the alarms generated by them are ignored. This paper explores the use of machine learning methods for automatically identifying patient deterioration, using data acquired from continuous patient monitors. We compare generative and discriminative techniques (a probabilistic method using a mixture model, and a support vector machine, respectively). It is well-known that parameter tuning affects the performance of such methods, and we propose a method to optimise parameter values using “partial AUC”. We demonstrate the performance of the proposed method using both synthetic data and patient vital-sign data collected from a recent observational clinical study.

**Index Terms**—support vector machine, novelty detection, one-class classification, parameter optimisation, partial AUC.

## I. INTRODUCTION

### A. Detecting Patient Deterioration

**A**DVERSE events in acutely ill hospital patients may occur when their physiological condition is not recognised or acted upon early enough [1]. Clinical guidance in the UK [2] recommends the regular observational recording of certain vital signs (such as heart rate, HR, measured in beats per minute; respiration rate, RR, measured in breaths per minute; blood oxygen saturation, SpO<sub>2</sub>, measured as a percentage; and systolic blood pressure, SysBP, measured in mmHg), combined with the use of early warning score (EWS) systems. The latter involve the clinician applying univariate scoring criteria to each vital sign in turn (e.g., “score 3 if heart rate exceeds 140 beats per minute”), and then escalating care to a higher level if any of the scores assigned to individual vital signs, or the sum of all such scores, exceed some threshold.

This current standard of care has a number of disadvantages. (i) The early-warning scores assigned to each vital sign, and the thresholds against which the scores are compared, are mostly determined heuristically<sup>1</sup>. (ii) EWS systems are used

<sup>1</sup>However, a large evidence base of vital-sign data was used to construct the EWS proposed in [3], which is currently undergoing clinical validation in its own study.

with periodic observation of vital signs, which may be made as infrequently as once every 12 hours in some wards. Patients may deteriorate significantly between observations. (iii) There is a significant error-rate associated with manual scoring, especially in the high-workload setting of a high-dependency clinical ward. (iv) Each vital sign is treated independently and correlations between vital signs are not taken into account.

This paper addresses these four disadvantages by evaluating automated systems, which use novelty detection algorithms.

### B. Novelty Detection for Patient Monitoring

Novelty detection, or one-class classification, involves construction of a model of normality using examples of “normal” system behaviour, and which then classifies test data as either “normal” or “abnormal” with respect to that model. This technique is particularly applicable to the monitoring of high-integrity systems, such as jet engines, manufacturing processes, or human patients.

Monitoring high-integrity systems is difficult due to the variability between individual systems of the same system type (such as different human patients of the same demographic background). The few examples of “abnormal” system behaviour that may exist for some population are often inapplicable to the analysis of previously-unseen individuals. For example, a heart rate of 50 beats per minute may be indicative of considerable physiological abnormality in one hospital patient, while it may be entirely normal for a fitter patient of the same age and background.

Finally, high-integrity systems typically exhibit a high degree of structural complexity, and can often comprise many sub-systems that interact in a non-linear manner. Thus, the potential space of “abnormality” is extremely large, and so the large resultant number of failure modes is often poorly understood. For example, the exact response of a particular human’s physiology to a given failure mode (such as, for example, deterioration leading to myocardial infarction) will vary significantly between patients, and what data exist are insufficient for constructing accurate models of these failure states, typically being derived from a small number of patients, with differing co-morbidities, lifestyles, etc.

Novelty detection avoids such problems by modelling the “normal” mode of operation of the system, which is often well-understood because most high-integrity systems function

“normally” most of the time, and then looking for deviations from that normal model.

This is appropriate for the monitoring of physiological condition in patients, because sufficient data exist from “stable” patients such that a model of the well-understood “normal” state of these patients may be constructed. Physiological deterioration may then be detected as being corresponding departures in the vital signs from that “normal” state.

### C. Overview

This paper considers the one-class support vector machine (SVM), which is a commonly-used method of performing novelty detection. Its formulation is briefly recapped in section II, where disadvantages arising from the setting of its parameters are discussed. Following discussion on the topic of evaluating classifiers in section III, a method of optimising parameter values in a one-class SVM is described in section IV, in light of those evaluation methods. The method is illustrated using simulated data in section V and patient vital-sign data, collected from an observational clinical study, in section VI. Limitations of the method, and potential future extensions, are discussed in section VII.

## II. ONE-CLASS SVMs

The one-class SVM is a frequently-employed method of performing novelty detection, and it has been applied to many such problems, including jet engine condition monitoring [4], signal segmentation [5], and fMRI analysis [6], among many others, a review of which may be found in [7].

### A. Formulation

This paper considers the one-class SVM formulation proposed by [8], in which a quantity  $l$  of  $d$ -dimensional data  $\{\mathbf{x}_1, \dots, \mathbf{x}_l\} \in \mathbb{R}^d$  are mapped into a (potentially infinite-dimensional) feature space  $\mathbb{F}$  by some non-linear transformation  $\Phi: \mathbb{R}^d \rightarrow \mathbb{F}$ . A kernel function  $k$  provides the dot product between pairs of transformed data in  $\mathbb{F}$ :

$$k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \quad (1)$$

A Gaussian kernel allows any data-point to be separated from the origin in  $\mathbb{F}$  [8], hence is chosen for us in the work described by this paper:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2) \quad (2)$$

where  $\sigma$  is the width parameter associated with the Gaussian kernel.

The decision boundary is a hyperplane in feature space  $\mathbb{F}$ , found by minimising the weighted sum of a support vector-type regulariser and an empirical error term depending on an overall margin variable  $\rho$  and individual errors  $\xi_i$ ,

$$\min_{w \in \mathbb{F}, \xi_i \in \mathbb{R}^l, \rho \in \mathbb{R}} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i - \rho \quad (3)$$

$$\text{subject to } w \cdot \Phi(\mathbf{x}_i) \geq \rho - \xi_i, \quad \xi_i \geq 0 \quad (4)$$

where  $w$  is a weight vector in the feature space, and  $C$  is a user-specified penalty parameter, with a larger  $C$  corresponding to a higher penalty to errors [9].

The decision function in feature space  $\mathbb{F}$  is:

$$z(\mathbf{x}) = w_o \cdot \Phi(\mathbf{x}) - \rho_o \quad (5)$$

with parameters

$$w_o = \sum_{i=1}^{N_s} \alpha_i \Phi(\mathbf{s}_i) \quad (6)$$

$$\rho_o = \frac{1}{N_s} \sum_{j=1}^{N_s} \sum_{i=1}^{N_s} \alpha_i k(\mathbf{s}_i, \mathbf{s}_j), \quad (7)$$

where  $\mathbf{s}_i$  are the support vectors, of which there are  $N_s$ , and where  $k$  is the Gaussian kernel defined in (1). We note that  $w_o \in \mathbb{F}$ ,  $\rho_o \in \mathbb{R}$ , and that  $\alpha_i$  are Lagrangian multipliers used to solve the dual formulation, more details of which may be found in [8] and which are not reproduced here. The “abnormal” data (i.e., those outside the single, “normal” training class) take negative values of  $z(\mathbf{x})$ , while “normal” data take positive values.

We note in passing that this approach is typically employed in favour of the one-class formulation proposed in [10], [11], in which a hypersphere of minimum radius is found to enclose the data in  $\mathbb{F}$ . The interested reader is directed to a useful tutorial for this latter method in [12].

## III. RECEIVER OPERATING CHARACTERISTIC (ROC) CURVES

The performance of a two-class decision rule can be summarised in a receiver operating characteristic (ROC) curve, which plots the true-positive rate on the vertical axis against the false-positive rate (FPR) on the horizontal axis, as the decision threshold varies [13]. Although an ROC curve gives a more thorough evaluation of classifier performance than a confusion matrix, it is difficult to compare two ROC curves. One possible comparison is to consider the area-under-the-ROC-curve (AUC), which integrates the FPR over varying thresholds. AUC is independent of a fixed decision threshold, and is invariant to prior class probabilities [14]. AUC represents the probability that a randomly chosen positive observation is correctly classified, and therefore a higher value of AUC indicates better separation between the two classes [14], [15].

For the novelty detection approach taken by this work, we label the “normal” data as “negative” cases, for ROC analysis, and the “abnormal” data as “positive” cases. Most practical novelty detection systems require low FPRs, and so we are most interested in the ROC curve for low values of FPR when evaluating the performance of a novelty detector. (Its performance at higher FPRs is irrelevant, and possibly confounding, because these represent choices of decision threshold that would never be used in practice.)

We therefore consider *partial AUC*, to restrict evaluation of the classifier only over those ranges of decision threshold that

are likely to be used in practice. Partial AUC is defined as the integral area between two false-positive rates [16]. Unlike AUC, whose maximum value is always 1, partial AUC depends on the two chosen false-positive rates, over which the ROC curve is integrated. We will use this partial AUC metric for optimisation of SVM parameters.

#### IV. PARAMETER OPTIMISATION FOR A ONE-CLASS SVM

##### A. Choosing appropriate parameter values

For the case of a Gaussian kernel  $k(\mathbf{x}_i, \mathbf{x}_j)$ , it is important to choose an appropriate value for the bandwidth parameter  $\sigma$ . Larger values of  $\sigma$  result in smoother decision boundaries, which therefore tend to exhibit lower variance (i.e., better ability to generalise to previously-unseen data), at the expense of increased bias (i.e., under-fitting the “normal” data space, as represented by the “normal” training data). Conversely, smaller values of  $\sigma$  provide decreased bias (i.e., a closer fit to the “normal” data space, as represented by the “normal” training data), but at the expense of increased variance (i.e., they are less able to generalise successfully to previously-unseen data). The “optimal” value for  $\sigma$  will depend on the distribution of the particular dataset under consideration, and it is not usually obvious how one should choose the value of  $\sigma$ . Typically, a cross-validation exercise is performed, where one uses a validation set as an estimation of how well the system will perform in practice, when presented with previously-unseen test data.

For a Gaussian kernel  $k(\mathbf{x}_i, \mathbf{x}_j)$ , the quantity  $-\log k(\mathbf{x}_i, \mathbf{x}_j)$  is the Euclidean distance between two observations scaled by a factor  $1/2\sigma^2$ . Based on this link between  $\sigma$  and Euclidean distance, we propose the following method to determine an appropriate value for  $\sigma$  in our discriminative case, adapted from a similar method proposed in [17] for selecting  $\sigma$  when estimating pdfs for probabilistic inference within a generative framework:

- 1) First, we calculate the local average Euclidean distance  $\Delta_i$  of  $K$  nearest neighbours from each observation in the training set, where  $K = \sqrt{l}$ ,

$$\Delta_i = \frac{1}{K} \sum_{j \in \mathcal{D}} \|\mathbf{x}_i, \mathbf{x}_j\|, \quad \forall i = 1 \dots l \quad (8)$$

where  $\mathcal{D}$  is the set of  $K$  nearest neighbours for  $\mathbf{x}_i$ .

- 2) Next, the global average distance  $\Delta_G$  is found by averaging  $\Delta_i$  over all the training data,  $\Delta_G = l^{-1} \sum_i \Delta_i$ . The value of  $\Delta_G$  provides a guide for the range of  $\sigma$ , where we define  $\sigma = \kappa \times \Delta_G$ , where  $\kappa$  is a linking constant between the value of  $\sigma$  and the global average distance  $\Delta_G$  of any dataset. Therefore,  $\kappa$  provides a guide for the appropriate value of  $\sigma$ , which is independent of  $l$ . Once an appropriate value of  $\kappa$  is chosen for one dataset, it provides a good starting point for another dataset with similar dynamics (e.g., for another patient vital-sign dataset, allowing the value of  $\kappa$  to be reused from previous analyses, when the dataset has changed).

The other parameter to optimise in a one-class SVM is  $\nu$ , which will be defined as follows. The support vector constraints [9] in terms of the penalty parameter  $C$  from (3) are

$$\sum_i \alpha_i = 1, \quad 0 \leq \alpha_i \leq C. \quad (9)$$

allowing us to state<sup>2</sup> that  $1/l \leq C \leq 1$ . Also,  $C$  may be written

$$C = \frac{1}{\nu l} \quad (10)$$

so we have  $1/l \leq \nu \leq 1$ . Therefore,  $\nu$  and  $C$  take values in the same range.

The parameter  $\nu$  serves as an upper bound on the proportion of training observations that lie on the wrong side of the hyperplane, and is also a lower bound on the fraction of support vectors among normal training data [8]; i.e.  $\nu \leq N_s/l$ . Parameter  $\nu$  is used in this investigation instead of  $C$ , due to its clear meaning, as described above; the value of  $C$  can be easily recovered using (10).

Based on the above discussion, we will optimise parameters  $(\kappa, \nu)$  in section IV-B for the SVM approach taken in this paper.

##### B. Parameter optimisation using partial AUC

Combining the suggestions of the previous two sections, we take the following approach to optimising  $(\kappa, \nu)$ .

**STEP 1:** Choose a pair of parameter values  $(\kappa, \nu)$ .

**STEP 2:** Use the chosen  $(\kappa, \nu)$  to train a one-class SVM, which is dependent on a training set of “normal” data.

**STEP 3:** Use the resulting SVM to classify a validation dataset, which comprises both “normal” and “abnormal” data in equal quantity.

**STEP 4:** Compute partial AUC, using the validation results obtained in the previous step.

**STEP 5:** Repeat **STEP 1-4** using different values of  $(\kappa, \nu)$ , typically using a grid search. Choose the  $(\kappa, \nu)$  with the maximum partial AUC.

We assume the presence of *some* examples of “abnormal” behaviour, which are placed within the validation set for the purposes of parameter optimisation. However, as noted previously, these are likely to be small in quantity compared with the number of “normal” observations, and hence the training set is entirely comprised of “normal” data, and a one-class approach is taken.

A commonly-employed alternative which uses only “normal” data [12], [18] is to vary the SVM parameters until some fixed value of the false-positive classification rate is achieved (e.g., 0.05) when presented with the training set of “normal” examples. However, as demonstrated in [4], the overall expected performance of the one-class SVM can be improved by setting parameters by taking into account any available examples of “abnormal” data that may be available, even if they are few in comparison to the number of “normal”

<sup>2</sup>where the lower constraint arises because, in the worst case, we have all training data as support vectors and  $N_s = l$ , and therefore  $C \geq 1/l$  in order for  $\sum_i \alpha_i = 1$ . The upper constraint arises because  $\alpha_i \leq C$ .

training data. Therefore, we adopt the latter approach in our formulation, and include any available “abnormal” data in our validation set, described in the algorithm above.

## V. ILLUSTRATION USING SYNTHETIC DATA

The optimisation method described in the previous section is illustrated in this section using bivariate artificial data. We compare results obtained with the SVM to a commonly-used generative method of novelty, that of the Gaussian mixture model (GMM). The latter technique has been used for novelty detection in many applications, the details of which are surveyed in [7]. It is of particular interest to the investigation described by this paper, because previous work in performing novelty detection in patient vital-signs has used a generative approach, comprising a mixture of Gaussian kernels [19], [20].

### A. Methodology

The dataset employed in this section comprises 200 “normal” and 200 “abnormal” data, with distributions that are concave in support, as shown in figure 1. For the purposes of evaluating novelty detection algorithms, 50% of both datasets were held back for testing. 40% of the “normal” data are used for training, with the remaining 10%, and all remaining “abnormal” data, used for validation.

For the SVM, the training and validation procedures (in which parameter optimisation occurs) proceed as described in section IV-B.

The GMM is defined by the pdf

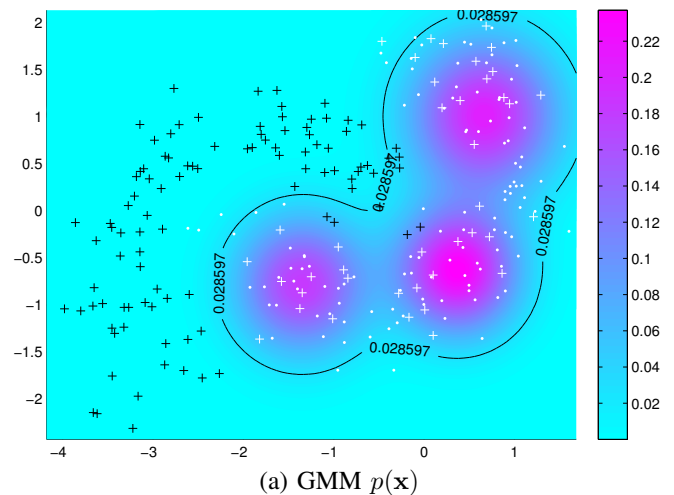
$$p(\mathbf{x}) = \sum_{i=1}^M \pi_i p(\mathbf{x}|\boldsymbol{\theta}_i) \quad (11)$$

which is comprised of  $M$  component distributions, each of which has a prior probability  $\pi_i$  and a likelihood  $p(\mathbf{x}|\boldsymbol{\theta}_i) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ , where  $\boldsymbol{\mu}_i$  and  $\boldsymbol{\Sigma}_i$  have their usual meanings of the centre and covariance matrix for multivariate Gaussian  $i$ , respectively. The training procedure finds appropriate values for the quantities shown in (11); in this case, the maximum likelihood estimates were determined using expectation maximisation [21].

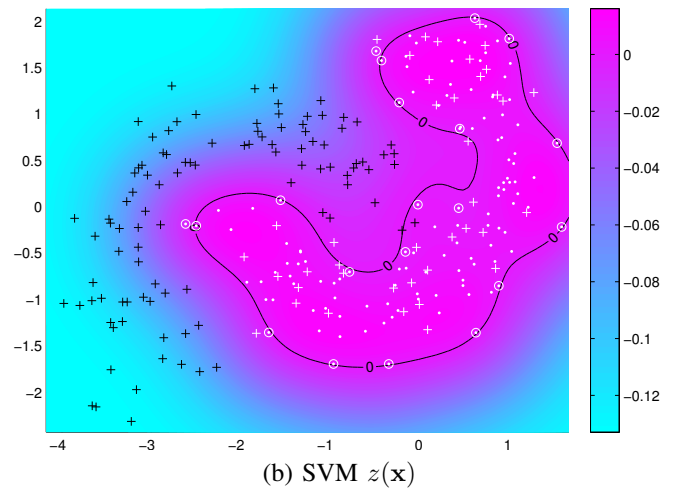
Novelty detection with a GMM is typically performed by setting a decision threshold on the pdf  $p(\mathbf{x})$ , which, as with the decision boundary of the SVM, may be set (i) using only “normal” data, whereby one finds that threshold that yields some pre-determined false-positive rate, or (ii) using a validation set that contains any available “abnormal” examples. In order to allow a direct comparison with the SVM, the latter approach was taken for setting the GMM decision threshold, and partial AUC was again used as the optimisation metric, the minimum of which (with respect to the validation set) yields the “optimal” value of the decision threshold on the pdf  $p(\mathbf{x})$ .

### B. Results

Figure 1 shows the output of both GMM and SVM novelty detectors when presented with the previously-unseen test dataset. The upper plot shows  $p(\mathbf{x})$  for the GMM, where the decision threshold on the pdf is shown as a black contour



(a) GMM  $p(\mathbf{x})$



(b) SVM  $z(\mathbf{x})$

Fig. 1. (a) GMM output  $p(\mathbf{x})$  using the “optimal” parameter values determined using validation; “normal” training data are shown by white  $\{\cdot\}$ , and “normal” and “abnormal” test data are shown by white  $\{+\}$  and black  $\{+\}$ , respectively. (b) SVM output  $z(\mathbf{x})$  using the “optimal” parameter values determined using validation; labelling scheme as above, with support vectors circled.

on the pdf, which describes the locus of the “normal” training data (shown as white dots). The test data are shown as crosses in white and black for “normal” and “abnormal” classes, respectively. It may be seen that the training and validation procedure resulted in  $M = 3$  component distributions being used, where these distributions were constrained to have isotropic covariance matrices (although each may be assigned a different determinant during training).

The lower plot shows  $z(\mathbf{x})$  for the SVM, as defined in (5), where the decision threshold (which occurs at  $z(\mathbf{x}) = 0$  for a SVM) is shown as a black contour. The symbols used are the same as those for the GMM plot, described above, where training examples that are support vectors ( $\alpha_i > 0$ ) are circled in white.

Table I shows results obtained from both GMM and SVM when applied to the previously-unseen test data. Defining true-

TABLE I

NOVELTY DETECTION PERFORMANCE OF GMM AND SVM APPLIED TO TEST DATA, AT “OPTIMAL” THRESHOLD. ONE STANDARD DEVIATION ON THE RESULT IS SHOWN FOR EACH, USING THE RESULTS OF 50 EXPERIMENTS, IN WHICH EACH EXPERIMENT INVOLVES RANDOM SELECTION OF NEW TEST DATA FROM THOSE AVAILABLE, WITH EQUAL NUMBERS OF TEST DATA DRAWN FROM THE “NORMAL” AND “ABNORMAL” CLASSES.

Classifier	Accuracy	Partial AUC	Sensitivity	Specificity
GMM	$0.95 \pm 0.01$	$0.28 \pm 0.01$	$0.95 \pm 0.01$	$0.93 \pm 0.03$
SVM	$0.96 \pm 0.01$	$0.30 \pm 0.01$	$0.99 \pm 0.01$	$0.88 \pm 0.01$

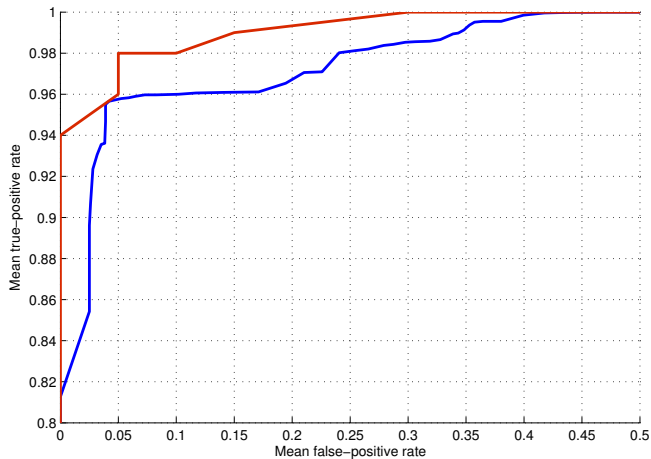


Fig. 2. ROC curve for results obtained using GMM and SVM to classify previously-unseen artificial test data, shown in blue and red, respectively. The mean of 50 experiments has been shown at each point on the ROC curve, where each experiment involves random selection of new test data from those available, with equal numbers of test data drawn from the “normal” and “abnormal” classes.

positive, true-negative, false-positive, and false-negative to be TP, TN, FP, and FN, respectively, then *accuracy* is defined to be  $(TP + TN) / (TP + TN + FP + FN)$ , *sensitivity* is  $TP / (TP + FN)$ , and *specificity* is  $TN / (TN + FP)$ . It may be seen that both methods perform similarly with this simple bivariate example, with the SVM performing marginally better overall, as shown by slightly higher accuracy and AUC results. It achieves this with a higher sensitivity, at its “optimal” threshold, at the cost of a lower specificity. This is confirmed in figure 2, which shows an overall higher ROC curve for the SVM than the GMM.

We conclude that the training and optimising procedures for both techniques result in stable, usable parameter configurations, and hence we now extend our analysis to consider higher-dimensional patient vital-sign data, as acquired from a recent clinical study.

## VI. PATIENT VITAL-SIGN MONITORING

This section reports results obtained from evaluating both GMM and the proposed SVM method for patient vital-sign monitoring.

TABLE II

NUMBER OF CLINICAL OBSERVATIONS IN THE TRAINING, VALIDATION, AND TEST SETS

	Train	Validate	Test
Normal	1,240	65	65
Abnormal	0	65	65

### A. Dataset

We consider vital-sign data acquired from patients in a “step-down unit” (SDU), which is a level of acuity lower than that of the intensive care unit (ICU). There is a significant need for effective novelty detection systems in such wards, because patient deterioration can go unnoticed by clinical staff, leading to adverse patient outcomes. Existing patient monitors generated univariate alarms whenever vital signs exceeded some pre-defined threshold, and often go unheeded due to the high false-positive rate of such alarms, where [22] reported results of a study in which it was deemed that 84% of alarms from conventional continuous patient monitors were false alarms.

The dataset used for the work described by this section comprises measurements of heart rate, respiratory rate, blood oxygen saturation, and systolic blood pressure, acquired once every four hours by ward staff (as is common practice in most SDU-level wards in the UK and the US) at the Oxford Cancer Hospital, Oxford, UK. 1,500 such clinical observations  $x_i \in \mathbb{R}^4$  were acquired from 19 patients, who were recovering from upper gastro-intestinal surgery.

### B. Methodology

130 of the clinical observations were deemed by clinicians to be sufficiently “abnormal” that the patient would require clinical review. The remaining 1,370 were thus classified as being “normal”. The available “abnormal” data are insufficient to train a multi-class classifier, being small in comparison with the number of “normal” data, and therefore the novelty detection approach is justified for this clinical application.

Table II shows how the 1,500 observations may be assigned to each of the training, validation, and test sets. The available examples of abnormality must be split between the validation set (to enable parameter optimisation, as described in section IV-B) and the test set (to allow evaluation of the results); therefore, the 130 examples of abnormality are split equally between validation and test data. Similar numbers of “normal” data are required for each of the validation and test sets; the remainder of the “normal” data are placed into the training set, as shown in the table.

The split between the training, validation, and test sets was performed randomly. In order to test the variability of the results to this random partitioning, 50 experiments were performed, each experiment containing a different random partition of the data between the three sets, and each experiment requiring retraining, revalidation, and retesting, in order to obtain fully independent results for each experiment.



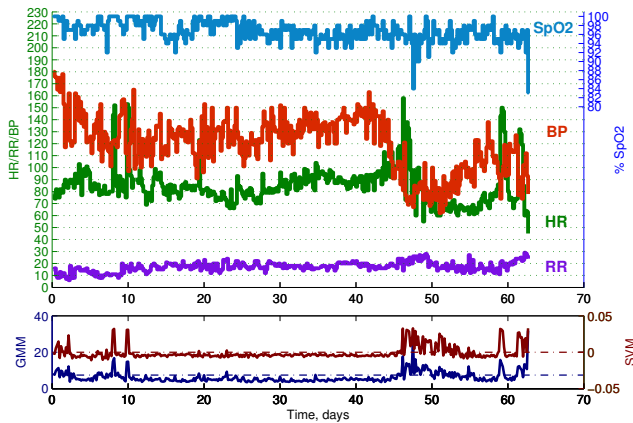


Fig. 3. The upper plot shows time-series of vital signs for an exemplar patient, showing HR, BR, SpO<sub>2</sub>, and BP in green, purple, blue, and red, respectively, with time (in hours) shown on the horizontal axis. The lower plot shows novelty scores derived from GMM output  $-\log p(\mathbf{x})$  and SVM output  $z(\mathbf{x})$  on the same time-base, in blue and red, respectively. Horizontal lines in the lower plot show the decision thresholds for the GMM and SVM in blue and red, respectively.

### C. Results

An example of the application of the techniques to patient vital-sign data is shown in figures 3 and 4.

The first example shows a patient who enters the ward, following surgery, in a state of physiological derangement, as shown by elevated BP (at around 180 mmHg). The patient begins to stabilise, but, after 10 days, episodes of tachycardia (elevated HR, reaching 150 bpm) may be seen, with a corresponding increase in RR from 10 breaths/min to 20 breaths/min. The patient then stabilises again, but deteriorates significantly after 45 days, showing periods of prolonged hypotension (decreases in BP below 60 mmHg), with corresponding tachycardia (HR reaching 160 bpm), and desaturations (SpO<sub>2</sub> decreasing to 84%).

The output of the GMM and SVM are shown beneath, along with their decision thresholds. For the purposes of visualisation, the output of the GMM, which is a density  $p(\mathbf{x})$ , has been scaled into a novelty score  $-\log p(\mathbf{x})$  such that it takes high values for data with low density; i.e., “abnormal” data will take high novelty scores. The SVM output needs no such transformation, as it takes positive values when data on the “non-normal” side of the decision boundary are presented.

It may be seen from the figure that both the SVM and the GMM increase in value during the initial post-surgical period of abnormality, during the transient period around 10 days, and in the final period of deterioration. The similarity of the GMM and SVM output is not accidental, as the  $-\log p(\mathbf{x})$  scaling of the GMM output makes it a comparable score to the SVM, because the SVM asymptotically approaches the level sets on the pdf in its tails [23].

The second example shows a patient who is similarly unstable at the start of their admission to the Cancer Hospital ward, following surgery. This patient exhibits bradycardia (low HR, decreasing to 40 bpm). After 5 days, periods of apnoea

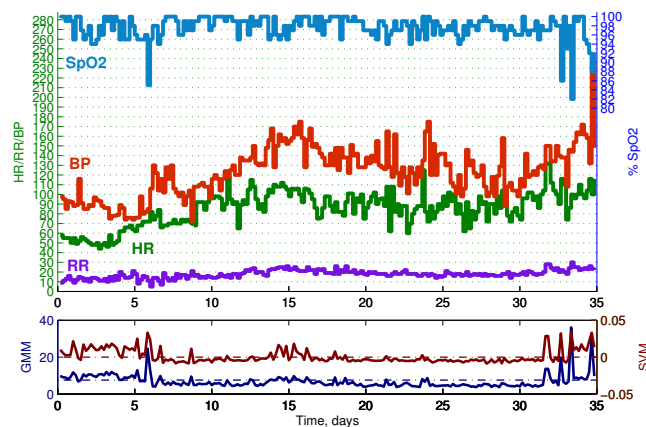


Fig. 4. The upper plot shows time-series of vital signs for a second exemplar patient, showing vital signs and novelty detection output as in the first example.

TABLE III  
NOVELTY DETECTION PERFORMANCE OF GMM AND SVM APPLIED TO TEST CLINICAL DATA, AT “OPTIMAL” THRESHOLD. ONE STANDARD DEVIATION ON THE RESULT IS SHOWN FOR EACH.

Classifier	Accuracy	Partial AUC	Sensitivity	Specificity
GMM	0.92 ± 0.03	0.25 ± 0.02	0.92 ± 0.04	0.92 ± 0.04
SVM	0.95 ± 0.01	0.28 ± 0.02	0.98 ± 0.01	0.92 ± 0.03

are evident (low BR, decreasing below 10 breaths/min) with corresponding decreases in SpO<sub>2</sub>. Periods of abnormal physiology occur after around 15 days, with transient hypertension (increases in BP over 160 mmHg). Finally, a period of extreme deterioration may be seen at the end of the patient stay, with extreme hypertension (BP exceeding 200 mmHg) and corresponding extreme desaturation (SpO<sub>2</sub> decreasing below 82%).

It may be seen that, again, both the GMM and SVM scores increase above their respective decision thresholds for these periods of abnormality.

Table III shows the overall results for both GMM and SVM after 50 experiments. Unlike the bivariate case considered in section V, there is a significant difference between the results obtained from each method. The SVM achieves higher accuracy and partial AUC, as before, but matches the specificity of the GMM (0.92 to 0.98) while improving on the sensitivity (from 0.92 to 0.98).

This is confirmed by the ROC plots shown in figure 5, in which it may be seen that the ROC curve for the SVM is higher than that for the GMM throughout most of the interval on the horizontal axis.

## VII. CONCLUSIONS AND DISCUSSION

We have proposed a method for optimising SVM parameters in a natural manner, considering  $\nu$  and  $\kappa$ , which, we have argued, have a more intuitive interpretation than the conventional  $C$  and  $\sigma$  parameters (assuming that a Gaussian kernel is used).



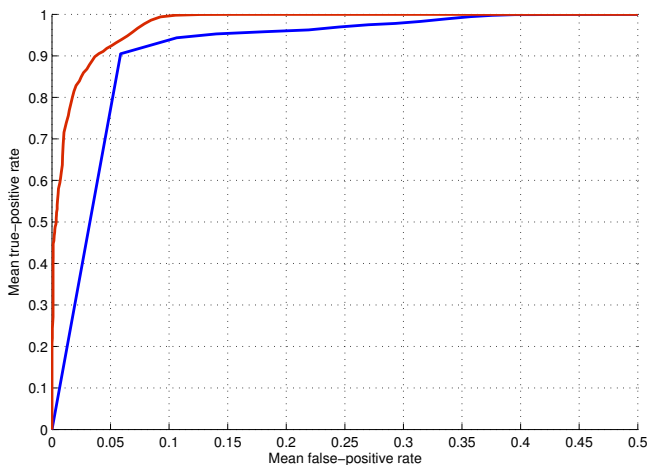


Fig. 5. ROC curve for results obtained using GMM and SVM to classify previously-unseen clinical test data, shown in blue and red, respectively. The mean of 50 experiments has been shown at each point on the ROC curve, where each experiment involves random selection of new test data from those available, with equal numbers of test data drawn from the “normal” and “abnormal” classes.

While we have demonstrated the method for one-class SVMs, they are equally applicable to two- and multi-class SVMs.

Clinical data acquired from a recent observational study of Cancer Hospital patients have been used to demonstrate that automated methods can be used to identify patient deterioration. Existing methods, based on generative mixture distributions have been shown to be outperformed by SVM-based novelty detection on the preliminary data considered so far in this study. This is perhaps unsurprising, given that the SVM minimises its objective function so as to result in a small number of misclassifications in the high-dimensional space of the vital signs. In comparison, generative methods, while offering more functionality than a discriminative method, typically exhibit higher misclassification rates.

The on-going clinical study will result in further data on which to confirm these preliminary findings.

#### ACKNOWLEDGEMENTS

LC was supported by the Overseas Research Students Award Scheme, provided by the UK Government, and later by the NIHR Biomedical Research Centre Programme, Oxford. DAC was funded by the Centre of Excellence in Personalised Healthcare funded by the Wellcome Trust and EPSRC under grant number WT 088877/Z/09/Z. The authors wish to thank Sarah Vollam and Deborah Evans for the collection of clinical data used in this investigation.

#### REFERENCES

- [1] National Patient Safety Association, “Safer care for acutely ill patients: Learning from serious accidents,” NPSA, Tech. Rep., 2007.
- [2] National Institute for Clinical Excellence, “Recognition of and response to acute illness in adults in hospital,” NICE, Tech. Rep., 2007.
- [3] L. Tarassenko, D. Clifton, M. Pinsky, M. Hravnak, J. Woods, and P. Watkinson, “Centile-based early warning scores derived from statistical distributions of vital signs,” *Resuscitation*, no. DOI:10.1016/j.resuscitation.2011.03.006, 2011.
- [4] P. Hayton, L. Tarassenko, B. Scholkopf, and P. Anuzis, “Support vector novelty detection applied to jet engine vibration spectra,” in *Proc. NIPS*, Denver, US, 2000, pp. 946–952.
- [5] A. Gretton and F. Desobry, “On-line one-class support vector machines. an application to signal segmentation,” in *Proc. IEEE ICASSP*, Hong-Kong, China, 2003.
- [6] D. R. Hardoon and L. M. Manevitz, “fMRI analysis via one-class machine learning techniques,” in *Proc. 19th International Joint Conference on Artificial Intelligence (IJCAI)*, Edinburgh, UK, 2005, pp. 1604–1605.
- [7] M. Markou and S. Singh, “Novelty detection: A review - part 2: Neural network based approaches,” *Signal Processing*, vol. 83, no. 12, pp. 2499–2521, 2003.
- [8] B. Scholkopf, J. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the support of a high-dimensional distribution,” *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.
- [9] C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.
- [10] D. M. J. Tax and R. P. W. Duin, “Data domain description using support vectors,” in *Proc. ESAN99*, Brussels, 1999, pp. 251–256.
- [11] D. Tax and R. Duin, “Support vector domain description,” *Pattern Recognition Letters*, vol. 20, pp. 1191–1199, 1999.
- [12] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, 1st ed. Cambridge, UK: Cambridge University Press, 2004.
- [13] A. R. Webb, *Statistical Pattern Recognition*, 2nd ed. Chichester, England: John Wiley and Sons Ltd., 2002.
- [14] A. P. Bradley, “The use of the area under the ROC curve in the evaluation of machine learning algorithms,” *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.
- [15] J. A. Hanley and B. J. McNeil, “The meaning and use of the area under the receiver operating characteristic (ROC) curve,” *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.
- [16] S. H. Park, J. M. Goo, and C. H. Jo, “Receiver Operating Characteristic (ROC) Curve: Practical Review for Radiologists,” *Korean Journal of Radiology*, vol. 5, no. 1, pp. 11–18, 2004.
- [17] C. M. Bishop, “Novelty detection and neural network validation,” *Proceedings of IEE Conference on Vision and Image Signal Processing*, vol. 141, no. 4, pp. 217–222, 1994.
- [18] B. Scholkopf and A. Smola, *Learning with Kernels*, 1st ed. Cambridge, USA: MIT Press, 2002.
- [19] L. Tarassenko, A. Hann, and D. Young, “Integrated monitoring and analysis for early warning of patient deterioration,” *British Journal of Anaesthesia*, vol. 98, no. 1, pp. 149–152, 2007.
- [20] A. Hann, “Multi-parameter monitoring for early warning of patient deterioration,” Ph.D. dissertation, University of Oxford, 2008.
- [21] C. M. Bishop, *Pattern Recognition and Machine Learning*. Berlin: Springer-Verlag, 2006.
- [22] C. Tsien and J. Fackler, “Poor prognosis for existing monitors in the intensive care unit,” *Critical Care Medicine*, vol. 25, no. 4, pp. 614–619, 1997.
- [23] R. Vert and J. Vert, “Consistency and convergence rates of one-class svms and related algorithms,” *Journal of Machine Learning Research*, vol. 7, pp. 817–854, 2006.



# Data Mining Research Trends in Computerized Patient Records

Payam Homayounfar  
Wrocław University of Economics,  
ul. Komandorska 118/120  
53-345 Wrocław, Poland  
Email: p.homayounfar@gmail.com

Mieczysław L. Owoc  
Wrocław University of Economics,  
ul. Komandorska 118/120  
53-345 Wrocław, Poland  
Email: mieczyslaw.owoc@ue.wroc.pl

**Abstract**—Over the last decades has the research on Data Mining made a great progress. Also the Computerized Patient Records (CPR) as part of Hospital Information Systems have improved in terms of usability, content coverage, and diffusion rate. The number of Health Care Organizations using the CPR is growing. Causally determined is the need for techniques and models to provide solutions for decision making based on the data stored from different sources in CPR. This paper provides an overview on the current research trends and shows the impact on the medical domain with the CPR.

## I. INTRODUCTION

INTERNATIONALLY the Computerized Patient Records (CPR) became growingly more important for health care institutions. The number of institutions changing over from the paper based patient files to the CPR are increasing. This evolutionary development will increase with the establishment of Data Mining (DM) and the associated techniques and applications. Before DM the CPR has been known as centralized data storage for patient data with limited the possibilities to analyze, to process, and to use the data for other questions except for some simple cases. This is the reason for using the word CPR instead of Electronic Patient Records which represents the limited analyzability of the patients data in the past.

The amount of medical and patient oriented data stored in CPR has grown strong progressive. The CPR contains medical data, laboratory data, and images from different modalities and organizational data from different sources with the purpose of patients care. DM is the key technology to evaluate, interpret and link information of the large amount of data. DM improves the value of CPR to support the process of decision-making and medical diagnosis [1].

In the context of medical data DM uses algorithms, tools, lifecycles of knowledge, and formalizations to extract patterns, information and knowledge from data stored in the CPR. DM transforms transactional data in the CPR from tacit knowledge into explicit knowledge [2]. In this context it is important to mention the link of DM to Knowledge Management (KM). KM is the system and managerial approach to the gathering, management, use, analysis, sharing, and discovery of knowledge [3]. KM deals with eliciting, representing, and storing explicit data. DM is a subfield of KM and is used as part of the knowledge discovery process. KM and DM have the same fundamental issues and must be com-

binated in the decision making process. Especially in medical applications the interaction and integration of DM and KM is essential [4]-[6].

DM is becoming an area of great interest for clinical practice and research as medical decisions must always be supported by arguments, and the explanation of decisions and predictions should be mandatory for an effective deployment of DM models. DM and KM are the most important technologies for enabling Evidence Based Medicine, which proposes strategies to apply evidence gained from scientific studies for the care of individual patients [7].

There are some scientific research projects with the purpose of merging clinical and research objectives like the I2B2 project at the Harvard University [8].

DM is the essential part of futures CPR. The objective of this paper is to give an overview of current research trends of DM in CPR. Firstly the Computerized Patient Record with its characteristics is described. DM is especially in CRP valuable and essentially necessary. Also the basic tasks of DM will be described in the paper. The following DM models of sophisticated machine learning models will be focused with their impacts on CPR in a separate section: Symbolic Learning vs. Rule induction, Case Based Reasoning, Natural Language Processing, Artificial Neuronal Networks, Bayesian Networks, and Fuzzy Models.

Not in scope of this paper is the view on aspects of technical systems, special tools for the DM techniques, description of underlying methodologies, legal issues, data privacy, and data security.

## II. CHARACTERISTICS OF COMPUTERIZED PATIENT RECORDS

The CPR is a collection of data in a database or repository, which is managed by application programs. It is a key part of hospital information systems. All relevant data and the investigations and interventions for one patient in one health care institute are collected in a structured manner and without redundancy in the electronic patient record. The data is stored on digital media and are always available electronically. The difference to data files in other segments like economical data of a customer is based on the complexity of medical data of patients [1], [4], [9].

The electronic patient file contains data from different areas [10]-[13] like personal details, billing information, case or medical history, clinical test results, diagnoses from dif-

ferent specialists, therapy information, digital pictures from various modalities, pictorial archiving of historical pathologic findings, important treatment data and results of control tests, representation of specific content for the single specialists (Anesthesiology, Radiology, Pathology, Cardiology, Endocrinology, Pharmacology, Odontology, Accident Medicine, ...), and nursing measures.

The complexity is not only based on the broad range and variety of data in the CPR, but also evolved from the input from many different sources of the data. Different sources of technical systems as well as the multiple groups inserting and extracting data in the CPR give a good impression of the complexity of CPR data.

Another characteristic for CPR is the huge amount of medical data. Pictures made with different modalities like Computer Tomography (CT) or Magnetic Resonance Tomography (MRT) as well as the measurement of laboratory and pathological reports produce a very large and storage intensive amount of data for each single patient. Having this in mind it is understandable that data in CPR reach many Giga-Bytes. The trend is rising as the sophisticated technical possibilities are growing and going much and creating more data.

The CPR provides a multidisciplinary information exchange for communication between the different stakeholders in health care institutions like medical doctors, physicians, laboratories, nurses, other members of the health care team and the administration and controlling staff [14].

The purpose of the CPR is divided into following prioritized areas [11]:

1. Patient care: The CPR provides the documented basis for planning care and treatment.
2. Communication: All stakeholders have the same information on a patient. This allows the communication with each other with the same actual data basis on a patient.
3. Legal documentation: Documentation of the treatment as well as the legal forms signed by the patient.
4. Billing and reimbursement: Coded treatment with e.g. International Classification of Diseases (ICD-10) or Diagnosis-related groups (DRG)
5. Research
6. Quality Management

The characteristics of DM in CPR and the uniqueness of medical data are the challenge in this field [4] are the volume and complexity of medical data, the importance of physicians interpretation, the sensitivity and specificity analysis, poor mathematical characterization of medical data, the canonical form, and ethical, legal as well as social issues (Data ownership, privacy and security)

### III. DATA MINING IN CONTEXT WITH KNOWLEDGE MANAGEMENT

Data Mining extracts patterns, explicit knowledge and information from data. The objective of data mining in this context is to support the medical doctor and the health care institution in decision making.

Data mining analyzes data and extracts models that allow the interpretation and transformation of the raw data in the CPR into knowledge. This is the entry point for knowledge management to create tacit knowledge. KM is the system and managerial approach to the gathering, management, use, analysis, sharing, and discovery of knowledge [3], [15]. KM deals with eliciting, representing, and storing explicit data. DM is a subfield of KM and is used as part of the knowledge discovery process. KM and DM have the same fundamental issues and must be combined in the decision making process. Especially in medical applications the interaction and integration of DM and KM is essential [4].

Data analysis in medicine depends more than in other areas on medical background knowledge. Further approaches, such as association and classification rules, joining the declarative nature of rules, and the availability of learning mechanisms are a great potential for effectively merging DM and KM [16].

Over the past years have many ontologies been developed. Ontologies have an important role in DM to facilitate knowledge sharing between different sources. An ontology is a specification of conceptualization. It describes the concepts and relationships that can exist and formalizes the terminology in a domain [17], [18].

### IV. DATA MINING RESEARCH IN COMPUTERIZED PATIENT RECORDS

The shown characteristics of medical and organizational data in CPR lead to the special problem of analyzing and linking data from different sources and qualities together. Especially in the CPR are many information hidden and are important to be revealed. The hidden information in the raw data are also caused by the complexity of the medical domain in CPR. It is easy to lose the track of a disease if different medical doctors make a diagnosis in their own domain and do not compare their findings with each other. Data mining can bridge the important gap and bring together the essence of the information.

Data mining and knowledge creation is more than a set of techniques for data analysis, it is the key for extracting information out of the mentioned data. Without data mining the storage of the data in the CPR would be not necessary as this makes the difference to the patients files based on paper.

Data mining techniques build a group of heterogeneous tools and techniques to different purposes along the process to create knowledge. There are descriptive and predictive models. The descriptive models identify similar patterns in the analyzed date by using classification, association rules and visualization. Predictive models use classification, regression and time series analysis to show the impact of a treatment to a patient based on the data of the past. Another way to categorize most of the data mining techniques distinguishes them into model building and clustering techniques. Model Building seeks to create a predictive model related to a specific question. Depending upon the techniques chosen, a model may be either opaque (results are clear, but the functions are unclear) or transparent (complete knowledge about the model at any prediction). Clustering attempts to segment

a population into one or more groups that have (as far as we are concerned) similar characteristics and are therefore expected to behave in a similar manner.

The following list shows the commonly used techniques of data mining for knowledge discovery [19]-[22]:

1. Summarization: The relevant Data from the CPR have to be generalized and abstracted. The result of this step is the set of task-relevant data.
2. Classification: Classification is the process of assigning data items to one or many predefined classes. The classification model contains a set of classification rules. These rules are also used for future data. It derives a function or model, which determines the class of a model based on its attributes [22]. For the CPR is the definition of the rules a complex task as the rules have to cover a deep understanding of medical and economical knowledge. In the medical diagnosis, the classification is the most critical data mining technique. An example for classification is the definition of a group of patients with high blood pressure and the assignment of equivalent patients to the group. Classification is in the most important task of DM in CPR [20]. Classification is a DK task for which also other Artificial Intelligence approaches like neural networks and decision trees are often used.
3. Association: Search the records and finding association patterns by using defined rules. For example search for a set of symptoms of diseases of patient, hat also occur to patients with other diseases. Automated systems are filtering association rules based on findings from medical transaction databases: Association rules are ranked for medical knowledge by using formal ontologies.
4. Clustering: The clustering identifies classes or groups for a set of objects. The clustering maximizes the similarities of objects assigned to a class. This is based on the criteria defined on the attributes of the objects. After the decision and assignment process of an object to the cluster, the objects are labeled with the corresponding cluster.
5. Trend analysis or time series analysis: The Trend is the result of comparing time related data over a period of time, e.g. blood pressure over a period of one month. The objects are snapshots of entities with certain values that can change over the time.
6. Forecasting: This is the prediction of the value for an object based on the data from the past.
7. Visualization techniques: They help to discover patterns in medical data sets as a starting point. Afterwards other data mining techniques have to be used to determine the details of the patterns.

## V. SETTING THE TREND OF DM WITH A SELECTION OF TECHNIQUES AND MODELS

The selection of DM techniques and models shows the trend of DM in CPR with a brief review of the key concepts.

Since the beginning of DM it was the aim to automatize DM techniques and models and to reduce the participation of human actions to a minimum. The beginning of this chapter describes Machine Learning, before the machine learning methods are described.

Machine Learning algorithms can be divided into supervised and unsupervised learning algorithms. In supervised learning, training examples exist of input-output pair patterns. Learning algorithms try to predict output values based of new examples, based on their input values. In unsupervised there are only input patterns without an explicit output available. Here is the aim of the algorithms to use input values to discover meaningful associations or patterns [3].

### A. Probabilistic and Statistical Models

Probabilistic and statistical analysis techniques and models have a strong theoretical foundation in DM research. Assigned to the statistical techniques are regression analysis, discriminant analysis, time series analysis, principal component analysis, and multi-dimensional scaling. Because of their maturity those models are often used as benchmarks for comparison with newer machine learning techniques [3].

An advanced and popular probabilistic model for CPR is the Bayesian Model. It was originated in pattern recognition research and frequently used to classify different objects into predefined classes based on a set of features. The model stores the probability of each class, each feature, and each feature given each class, based on the training data. New instances will be classified according to the existing probabilities. There are many variations of the Bayesian Model.

An important and popular machine learning technique is the Support Vector Machines (SVM). It is based on statistical learning theory, which aims to find a hyperplane to best separate two or many classes. The applied model has shown encouraging results as it has the performance among other learning methods in document classification [3].

### B. Symbolic Learning

Symbolic learning is implemented by applying algorithms that attempt to induce general concept descriptions that describe different classes of training example [3]. Many algorithms have been developed using algorithms to identify patterns that are useful in generating a concept description. Given a set of objects, symbolic learning created a decision tree that classifies all given objects correctly. At each step, the algorithm finds the attribute that best divides the objects into the different classes by minimizing the uncertainty. This way it is possible to create complete treatment plans in CPR [23]

### C. Case Based Reasoning

CBR is a problem solving paradigm that utilizes the specific knowledge of previously experienced situations or cases. It consists in retrieving past cases that are similar to the current one and in reusing solutions which were used successfully in the past, the current case can be retained. CBR provides a solution for solving new problems and understanding unfamiliar situations.

In medicine, CBR can be seen as a suitable instrument to build decision support tools able to use tacit knowledge [24].

The algorithms find similarities of cases by using tacit knowledge. The data of experiences are compared to similar solutions that were successful in past cases [25].

The classic CBR uses a cycle with the steps of retrieve, reuse, revise, and retain. An example for CBR in using CPR is if a medical doctor wants to decide whether or not to prescribe a special medication for a patient or not. With CBR the decision would include the medical history of the patient and all patients with similar patterns, their physical states, emotional states, observed behaviors, cognitive status, as well as safety concerns. Each case in a CBR application aims to support the decision making process and therefore contains specific information of these factors for an individual patient [25]. The knowledge base is the collection of those cases in a library or the case base. The case base is organized to facilitate retrieval of the most similar, or most useful experiences when a new case arises. The past solutions would be the starting point for solving the new case. This is the first step of CBR, retrieve. Reuse is the step where adjustments are necessary to fit the case to the new situation. The result of the reuse step is a suggested solution. In case of further necessary adjustments is the revise step necessary. Here is the output the tested and repaired case with a confirmed solution. The last step in the CBR cycle enables the system to grow and to learn from the acquired experiences. According to [25] CBR is particularly useful and applicable in health science and the CPR because of established histories for health care professionals, many publications that can be easily encoded in cases, reasoning from many existing examples, extensive data stores are available in CPR, and cases can complement general treatment guidelines to support personalized medical care for individual patients

The formalization of medical facts and their relations results in a high complexity, but there are many examples and cases of diseases, treatments and outcomes. Furthermore, many diseases are not well enough understood in medicine to have universal applicable treatments

The main selection of CBR systems in health care in their order of chronological appearance are SHRINK: Aid with psychiatric diagnosis and treatment [26], MNAOMIA: For diagnosing and treating psychiatric eating disorders [25], PROTOS: Diagnosis of audiological and hearing disorders [25], [27], CASEY: Diagnosis of heart failure patients by comparing them to earlier cardiac patients with known diagnoses [25], ICONS: Therapy planning system that recommends antibiotic therapy for patients with bacterial infections in the intensive care unit (ICU) [28], KASIMIR: Breast cancer decision support system that also takes missing data and the threshold effect into account [29], and HR3Modul: Decision support system for diagnosing stress related disorders, including signal measures like breathing and heart rate expressed as physiological time series [30].

An important trend of CBR for CPR is the integration of multimedia case representation. This allows using CBR for medical image interpretation for comparing pictures of different modalities of different patient [31]. CBR is also useful for including other factors in the decision process like the co-occurrence of multiple diseases, time series features, overlapping diagnostic categories, the need to abstract features

from time series representing temporal history, sensor signals, and continuous data.

#### *D. Natural Language Processing*

The content of CPR include a rich source of data and are often the major bottleneck for the deployment of effective clinical applications because textual information is difficult to access by computerized processes. Natural Language Processing (NLP) systems are automated methods containing some linguistic knowledge that aim to improve the management of information in text [32]. NLP allows the extraction of information and knowledge from medical notes, discharge summaries, and narrative patients reports. Current efforts on the construction of automated systems for filtering rules learned from medical transaction databases is an important area for CPR. The availability of a formal ontology allow the ranking of association rules by clarifying what are the rules confirming available medical knowledge, what are surprising, and which are to be filtered out. Currently, NLP systems in clinical environments process CPR to index and categorize reports, extract, structure, and codify clinical information in the reports to make them usable for other computerized applications, generate text to produce patient profiles and summaries, and improve interfaces to health care systems.

The challenges of NLP in the clinical domain described by [32] are the performance of the application, the availability of clinical text and confidentiality, the evaluation and sharing information across institutions, the Expressiveness as language can describe the same medical concept in many different ways, the heterogeneous formats, as there are no standards for writing a report in CPR, the abbreviated text in medical reports often omit information that can be interfered by health care employees based on their individual knowledge, the interpretation of clinical information as evident part of a medical report, as often further information are necessary to associate findings with potential diagnoses, and rare events can make it difficult to enable enough training examples for stabilization of NLP

NLP is based on in advance prepared formalization of the knowledge. NLP can be useful for ontology development, it can be used as a component in an ontology-driven information system and an NLP application can be enhanced with ontology.

There are different approaches to NLP in the CPR. Most approaches are using a combination of syntactic and semantic linguistic knowledge as well as heuristic domain knowledge [32]. Some use manually developed rules, and others are more statistically oriented. The NLP extraction process has two phases; first the report analyzer processes the report in order to identify segments and to handle irregularities. In the second phase the text analyzer as information extraction component uses linguistic knowledge associated with syntactic and semantic features. Also a conceptual model of the domain is used to structure and encode the clinical information [32]. The output is stored for subsequent use in clinical databases. NLP systems have different components that can vary from case to case. The main components according to [32] are morphological analysis: Process to break up original

words into canonical forms, lexical look up: Words or phrases are matched against a lexicon to determine their syntactic and semantic properties, syntactic analysis: Determination of the structure of a sentence to establish relationships of the words in a sentence, semantic analysis Process to show the clinical relevance of words and phrases, and encoding: Process to map the clinically relevant terms to established concepts. This is important to achieve widespread use of the structured information.

In order for NLP to become a main method in CPR it is important to further develop the standardization of report structures in the CPR as well as the standardization of the information model representing clinical information and vocabularies.

#### *E. Artificial Neural Networks*

Artificial Neural Network (ANN) or Neural Networks are computerized paradigms based on mathematical models with strong pattern recognition capabilities [33]. ANN are also called connectionist systems, parallel distributed systems, or adaptive systems, because they are comprised by a series of interconnected processing elements which work parallel in time [33]. ANN aim to build up information structures according to the human nervous system with a representation of neurons and synapses. An ANN is a graph consisting of many nodes connected to each other by weighted links. The knowledge in the ANN is represented by the totality of nodes and weighted links. Learning algorithms or learning rules can be used to adjust the connection weights in networks to predict or classify unknown examples. ANN work in the training mode and after stabilization in the testing mode. The process of ANN starts with a set of random weights and adjusts its weights automatically according to each learning example in an iterative process until the network stabilizes. This mode is called the learning mode, where the weights of the connections can change in order to respond to a present input.

Different types of ANN can solve many problems, like pattern recognition, pattern completion, determining similarities between pattern and data, interpolation of missing and noisy data, and automatic classification [34].

Many different types of ANN have been developed in the last two decades, e.g. the Self-organizing Map of Kohonen and the Hopfield networks described by Chen [3], [35]. Later in this chapter are Neuro Fuzzy Systems described, that also refer to ANN and are separated because of their growing importance. Particularly in the field of medicine and for usage of DM in CPR are ANN valuable as it is possible to build models with a high complexity, e.g. with multilayer feed forward networks. They can be defined as an array of processing elements arranged in layers. Information flows through each element in an input-output manner, where each element receives signals, manipulates them, and sends the signals to other connected elements.

#### *F. Bayesian Networks*

A particularly useful method for the CPR is represented by the Bayesian Networks (BN) which is used in different areas of medical applications. The BN represents the conjunc-

tion of knowledge representation, automated reasoning, and machine learning. The BN allows to explicitly represent the knowledge available in terms of a directed acyclic graph structure and a collection of conditional probability tables, and to perform probabilistic inference. BN use a directed acyclic graphical model to represent a set of random variables like quantities, latent variables, unknown parameters or hypotheses. Also the conditional relationships and independencies between the random variables are represented in the BN. The graphical structure represents knowledge about an uncertain domain. Each conditional relationship has its own probability function. Learning in BN is performed by intelligent algorithms. Influence diagrams help to generalize BN solve decision problems under uncertainty. In many practical settings the BN is unknown and one needs to learn it from the data. This problem is known as the BN learning problem, which can be stated informally as follows: Given training data and prior information (e.g. expert knowledge, casual relationships), estimate the graph topology (network structure) and the parameters [36]. Graphical models with undirected edges are generally called Markov random fields or Markov networks. These networks provide a simple definition of independence between any two distinct nodes based on the concept of a Markov blanket. Markov networks are popular in fields such as statistical physics and computer vision [36].

Machine learning and system learning for BN is to find the best matching Bayesian network graph with the best data fit for the decision problem.

#### *G. Analytic Learning, Fuzzy Logic, and Neuro Fuzzy Systems*

Knowledge is represented in analytical learning as logical rules and the performance of proofs for the rules. Traditional analytic learning systems depend on hard computing rules. As in the reality there is usually no distinction between values and classes, therefore fuzzy systems have been developed. Other concepts aim to avoid imprecise and vague information as they have a negative influence on the computed results. Fuzzy Systems use deliberately this type of information [34]. The result is often a simpler approach with more suitable models that are easier to handle. In the past Fuzzy Logic was not the first choice for DM in CPR because the simplicity did not fit to the complexity of medical and patient oriented data in CRM, but the trend has changed by combining different simple concepts, eg. Neuro Fuzzy Systems.

Fuzzy logic is an extension of traditional proposition logic. It deals with approximate reasoning by extending the binary membership. In contrast to classical set theory, in which an object or a case either is a member of a given set, fuzzy set theory makes it possible that an object or a case belongs to a set only to a certain degree [34]. Interpretations of membership degrees include similarity, preference and uncertainty. They can state how similar an object or case it to a prototypical one. They can indicate preferences between sub optimal solutions to a clinical problem, or they can model uncertainty in case of an imprecisely described situation or term [34]. For the CPR the set up of a fuzzy system is useful as many medical information are linguistic, vague or imprecisely described because a complete description would be too



complex. However, the limitation of the fuzzy system is reached when fuzzy concepts have to be represented by concrete membership degrees, which ensures that the system works as expected. A fuzzy system can be used to solve a problem, if knowledge exists about the solution in the form of if-then rules.

The development of Neuro Fuzzy Systems enhances the Fuzzy Systems where knowledge is represented in an interpretable manner, by the ability of ANN of learning. The advantage of the combination is that a problem can be solved without the need to analyze the problem itself in detail.

For CPR are the hybrid Neuro Fuzzy models interesting, which combine neuronal networks with fuzzy systems in a homogeneous architecture. The architecture can either be interpreted as a special neuronal network with fuzzy parameters, or as a fuzzy system implemented in a parallel distributed way.

#### H. Evolution Based Models

Evolution based models refer to computer-based methods inspired by biological mechanisms of natural evolution.. Evolution based algorithms have been applied to various optimization problems. They were developed on the basis of genetic principles. A population of potential solutions is initiated in the first step. The iterative process changes the population based on the operations mutation and crossover in different generations. The crossover operation is a high level process that aims at exploitation while mutation is a unary process that aims at exploration [3]. The selection process goes over different generations and selects the best fitting individuals. At the end of the process is the best solution presented. Due to the stochastic and global-search capability this technique is popular in medical informatics research [3].

## VI. CONCLUSION

CPR contain heterogeneous data in heterogeneous information systems and from heterogeneous sources. Not only the information technology improves the complexity of data mining in electronic patient files, but also the business site is very complex in terms of the medical description of doctors who try to describe the disease of a patient. DM is particularly useful in this domain.

The described techniques and methods of DM in CPR prove the fast development of research trends over the last decades. The usage of the research findings and developed new techniques and models will reach the full momentum after the CPR coverage rate has reached a much higher value. Currently, many systems in health care are separated solutions with a low integration rate. The benefits of DM research in CPR will be fully unlocked when the data will be interlinked. All methods have shown that the result of the decision proposal is relying on the quality of the data basis. This is obvious in Data Mining and shows the growing importance for Data Mining research and the usage in CPR. Future internet technologies will allow to use Data Mining in the Web over a broad data basis and link the results to existing CPR. This will allow to access knowledge in a today not known dimension and revolutionize the decision making

process. 'The current solutions of today build the foundation for the future scenarios.

Most of the described examples of DM techniques and methods related to practical problems in CPR are directed on one single problem, e.g. diagnosis for stress related heart attacks. Future trends will be integrating the different approaches, technologies, methodologies, and constructs into a DM framework of methodologies that link together different approaches. The start is already made with the linkage of ANN with Fuzzy Logic into Neuro Fuzzy Systems.

The challenges of data mining will also remain in future to deal with different scientific areas, to understand the patterns, to deal with complex relationships between attributes, interpolate missing or noisy data, mining very large databases, handle changing data and integrate the data with other data base systems. All these challenges are particularly important for CPR.

## REFERENCES

- [1] K.J. Cios and G. W. Moore, "Medical data mining and knowledge discovery," Berlin Heidelberg: Springer, 2001, pp. 1-67.
- [2] S. S. R. Abidi, "Knowledge management in healthcare towards 'knowledge-driven' decision support services," *Intl J Med Inf*, 63, 5-18.
- [3] H. Chen, S. S. Fuller, C. Friedmann, and W. Hersch, "Knowledge management, data mining, and text mining in medical informatics," in *Medical Informatics: Knowledge Management and Data Mining in Biomedicine*, H. Chen, S.S. Fuller, C. Friedmann, and W. Hersch, Eds., New York: Springer Science + Business Media, 2005, pp. 3-22.
- [4] K. J. Cios and G. W. Moore, "Uniqueness of medical data mining," in *Artificial Intelligence in Medicine*, Elsevier, 26 (2002), 2002, 1-24, <http://www.cs.uwm.edu/~mani/fall05/smi/link/pdf/aimj-medkdd1.pdf> (25.5.2011).
- [5] V. Sintchenko, "Information processing in clinical decision making," in *Handbook of research on informatics in healthcare and biomedicine*, INITIAL Lazakidou, Ed., Hersey London: Idea, 2006, pp. 1-8.
- [6] K. P. Soman, S. Diwakar, and V. Ajay, "Insight into data mining theory and practice," New Delhi: Prentice-Hall, 2006, pp. 1-19.
- [7] D. L. Sackett, W. M. Rosenberg, J. A. Gray, R. B. Haynes, and W. S. Richardson, "Evidence based medicine: What it is and what it isn't," *BMJ* 312 (7023), 71-2, 2009.
- [8] D. T. Heinze, M. L. Morsch, B. C. Potter, R. E. Jr. Sheffer, "Medical i2b2 NLP smoking challenge: the A-Life system architecture and methodology", *J Am Med Inform Assoc*, 15(1), 40-3, 2008.
- [9] S. Bullas and J. Bryant, "Complexity and its impacts for health systems Implementation," in *Information Technology in Health Care 2007*, J.I. Westbrook, E.W. Coiera, J.L. Callen, and J. Aarts, Eds., Amsterdam Lancaster Fairfax: IOS, 2007, pp.37-44.
- [10] P. Schmücker, "Elektronik patient file and digital archive in hospitals," in *MEDNET Workbook integrated health care*, U. Eissing, N. Kuhr, and G. Noelle, Eds., ORT UND VERLAG, 2003, pp.103-115.
- [11] K. A. Wagner, F. W. Lee, and J. P. Glaser, "Health Care Information Systems: A Practical Approach for Health Care Management," San Francisco: Wiley, 2005, pp. 3-42.
- [12] O. Galani and A. Nikiforou, "Electronic health record," in *Handbook of research on informatics in healthcare and biomedicine*, INITIAL Lazakidou, Ed., Hersey London: Idea, 2006, pp.1-8.
- [13] M.A. Montero and S. Prado, "Electronic health record as a knowledge management tool in the scope of health," in *Knowledge Management for Health Care Procedures: ECAI 2008 Workshop K4HelP 2008*, D. Riano, Ed., Berlin Heidelberg: Springer, 2008, pp. 152-166.
- [14] B. Fong, A. C. M. Fong, and C. K. Li, "Telemedicine technologies: information technologies in medicine and telehealth," West Sussex: Wiley, 2011, pp. 67-105.
- [15] M. L. Owoc, "Knowledgebases: a management context and development determinants," in Proc. of 2003 Informing Science and Information Technology Education Conference, Pori, 2003, pp. 1193-1199, <http://informing-science.org/proceedings/IS2003Proceedings/docs/147Owoc.pdf> (20.05.2011).



- [16] J. van der Zwaan, E. T. K. Sang, and M. de Rijke, "An experiment in automatic classification of pathological reports," in *Artificial Intelligence in Medicine, AIME 2007 Amsterdam, July 2007, Proceedings*, R. Bellazzi, A. Abu-Hanna, and J. Hunter, Eds., Berlin Heidelberg: Springer, 2007, pp. 207-216.
- [17] M. Gruninger and J. Lee, "Ontology: applications and design," *Communication of the ACM*, 45(2), 2002, pp. 39-41.
- [18] C. Romero-Tris, D. Riano, and F. Real, "Ontology-based retrospective and prospective diagnosis and medical knowledge personalization," in *Knowledge Representation for Health-Care, ECAI 2010 Workshop KR4HC 2010, Lisbon, Portugal, August 2010*, D. Riano, A. Teije, S. Miksch, and M. Peleg, Eds., Berlin Heidelberg: Springer, 2011, pp. 1-15.
- [19] Yo. Wang, D. Niu, and Ya. Wang, "Power load forecasting using data mining and knowledge discovery technology," in *Intelligent Information and Database Systems: Second International Conference, ACI-IDS, March 2010*, N.T. Nguyen, M.T. Le, and J. Swiatek, Eds., Berlin Heidelberg New York: Springer, 2010, pp. 319-328.
- [20] S. K. Wasan, V. Bhatnagar, H. Kaur, "The impacts of data mining techniques on medical diagnostics," *Data Science Journal*, Volume 5, 19 October 2006, [http://www.jstage.jst.go.jp/article/dsj/5/0/119/\\_pdf](http://www.jstage.jst.go.jp/article/dsj/5/0/119/_pdf) (24.05.2011).
- [21] S. Tangsripiroj and M. H. Samadzadeh, "A taxonomy of data mining applications supporting software reuse," in *Intelligent System Design and Applications*, A. Abraham, K. Franke, and M. Köppen M, Eds., Berlin Heidelberg: Springer, 2003, pp. 303-311.
- [22] M. Kwiatkowska, M. S. Atkins, L. Matthews, N. T. Ayas, and C. F. Ryan, "Knowledge-based induction of clinical rediction rules," in *Data Mining and Medical Knowledge Management: Cases and Applications*, P. Berka, J. Rauch, and D.A. Zighed, Eds., Hershey London: Idea, 2009, pp. 350-375.
- [23] S. N. S. Saad, A. M. Razali, A. A. Bakar, and N. R. Suradi, "Developing treatment plan support in outpatient health care delivery with decision trees technique", in *Advanced Data Mining and Applications*, L.Cao, Y. Feng, and J. Zhong, Eds., Berlin Heidelberg: Springer, 2010, pp. 475-482.
- [24] R. Schmidt, S. Montani, R. Bellazzi, L. Portinale, and L. Gierl, "Case-based reasoning for medical knowledge-based systems," *Intl J Med Inf* 64(2-3), 2001, pp. 355-367.
- [25] I. Bichindaritz and C. Marling C, "Case-based reasoning in health science: foundations and research directions," in *Computational Intelligence in Healthcare 4: Advanced Methodologies*, I. Bichindaritz, S. Vaidya, A. Jain, and L. C. Jain, Eds., Berlin Heidelberg: Springer, 2010, pp. 127-157.
- [26] J. L. Kolodner and R. M. Kolodner, "Using experience in clinical problem solving: introduction and framework," *IEEE Transactions on Systems, Man and Cybernetics*, 17(3), 1987, pp. 420-431.
- [27] S. Cox, M. Oakes, S. Wermter, and M. Hawthorne, "AudioMine: medical data mining in heterogeneous audiology records. world academy of science", *Engineering and Technology*, 1. 2005, <http://www.waset.org/journals/waset/v1/v1-2.pdf> (25.5.2011).
- [28] R. Schmidt, B. Pollwein, and L. Gierl, "Case-based reasoning for antibiotics therapy advice," in *ICCB 1999. LNCS (LNAI)*, K.-D. Althoff, R. Bergmann, and L.K. Branting, Eds., vol. 1650, Heidelberg: Springer, 1999, pp. 550-559.
- [29] M. d'Aquin, J. Lieber, and A. Napoli, "Adaptation knowledge acquisition: a case study for case-based decision support in oncology," *Computational Intelligence*, 22(3-4), 2006, pp. 161-176.
- [30] P. Funk and N. Xiong, "Case-based reasoning and knowledge discovery in medical applications with time series," *Computational Intelligence*, 22(3-4), 2006, pp. 238-253.
- [31] T. S. Yoo, "3D medical informatics," in *Medical Informatics: Knowledge Management and Data Mining in Biomedicine*, H. Chen, S. S. Fuller, C. Friedmann, and W. Hersch, Eds., New York: Springer Science + Business Media, 2005, pp. 333-355.
- [32] C. Friedmann, "Semantic text parsing for patient records," in *Medical Informatics: Knowledge Management and Data Mining in Biomedicine*, H. Chen, S. S. Fuller, C. Friedmann, and W. Hersch W, Eds., New York: Springer Science + Business Media, 2005, pp. 423-448.
- [33] M. Sordo, S. Vaidya, and L.C. Jain, "An introduction to computational intelligence in healthcare: New Directions," in *Advanced Computational Intelligence Paradigms in Healthcare*, M. Sordo, S. Vaidya, and L. C. Jain, Eds., 3rd ed., Berlin Heidelberg: Springer, 2010, pp. 1-26.
- [34] A. Klose, "Extracting fuzzy classification rules from partially labeled data," in *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, vol. 8, Berlin Heidelberg: Springer, 2004, pp. 417-427.
- [35] S. Eggers, Z. Huang, H. Chen, L. Yan, C. Larson, A. Rashid, M. Chau, and C. Lin C, "Mapping medical informatics research," in *Medical Informatics: Knowledge Management and Data Mining in Biomedicine*, H. Chen, S.S. Fuller, C. Friedmann, and W. Hersch W, Eds., New York: Springer Science + Business Media, 2005, pp. 35-58.
- [36] I. Ben-Gal, "Bayesian networks," in *Encyclopedia of Statistics in Quality & Reliability*, F. Ruggeri, F. Faltin, and R. Kenett, Eds., Wiley, 2008, <http://onlinelibrary.wiley.com/doi/10.1002/9780470061572.eqr089/full> (27.05.2011).



# A Bezier Curve Approximation of the Speech Signal in the Classification Process of Laryngopathies

Jarosław Szkoła, Krzysztof Pancerz

Institute of Biomedical Informatics  
University of Information Technology and Management  
Rzeszów, Poland

Email: jszkola@wsiz.rzeszow.pl, kpancerz@wsiz.rzeszow.pl

Jan Warchol

Department of Biophysics  
Medical University of Lublin  
Lublin, Poland

Email: jan.warchol@am.lublin.pl

**Abstract**—The research concerns a computer-based clinical decision support for laryngopathies. The classification process is based on a speech signal analysis in the time domain using recurrent neural networks. In our experiments, we use the modified Elman-Jordan neural network. In the preprocessing step, an original signal is approximated using Bezier curves and next the neural network is trained. Bezier curve approximation reduces the amount of data to be learned as well as removes a noise from the original signal.

**Index Terms**—computer-based clinical decision support; recurrent neural networks; laryngopathies; Bezier curves; approximation

## I. INTRODUCTION

COMPUTER-BASED clinical decision support (CDS) is defined as the use of a computer to bring relevant knowledge to bear on the health care and well being of a patient [1]. Our research concerns designing methods for CDS in a non-invasive diagnosis of selected larynx diseases. Two diseases are taken into consideration: Reinke's edema and laryngeal polyp. In general, the diagnosis is based on an intelligent analysis of selected parameters of a patient's speech signal (phonation). The proposed approach is non-invasive. Comparing it to direct methods shows that it has several advantages. It is convenient for a patient because a measurement instrument is located outside the voice organ. This enables free articulation. Moreover, different physiological and psychological patient factors impede making a diagnosis using direct methods.

The majority of methods proposed to date are based only on the statistical analysis of the speech spectrum (e.g. [2]) as well as the wavelet analysis. In our research, we are going to propose a hybrid approach, which is additionally based on a signal analysis in the time domain. Preliminary observations of signal samples for patients from a control group and patients with a confirmed pathology clearly indicate deformations of standard articulation in precise time intervals. In our previous papers (see [3], [4], and [5]), we have taken into consideration the usage of recurrent neural networks (RNNs), especially, the Elman and Jordan networks [6], [7] also known as "simple

recurrent networks." RNNs can be used for pattern recognition in time series data due to their ability of memorizing some information from the past. The Elman networks (ENs) are a classical representative of RNNs. To improve learning ability of ENs we have modified and combined them with the Jordan networks. Such networks manifest a faster and more exact achievement of the target pattern. Moreover, for the time domain analysis, RNNs have the capability of extracting the phoneme articulation pattern for a given patient (articulation is an individual patient feature) and the capability of assessment of its replication in the whole examined signal.

In contrast to approaches shown in [3], [4], and [5], in the approach presented in this paper, we do not learn the neural network using samples of a speech signal directly. Now, we introduce a preprocessing step. In this step an original signal is approximated using Bezier curves and next the neural network is trained. Bezier curve approximation reduces the amount of data to be learned (by one order of magnitude), as well as, removes a noise from the original signal.

Our paper is organized as follows. After introduction, we shortly describe the medical background related to larynx diseases (Section II). In Section III, we show the procedure used to find approximation of a speech signal using Bezier's curves. Section IV describes the use of the modified Elman-Jordan neural network in finding disturbances in a speech signal. In Section V, we present results obtained by experiments done on real-life data. Some conclusions and final remarks are given in Section VI.

## II. MEDICAL BACKGROUND

A model of speech generation is based on the "source-filter" combination. The source is larynx stimulation, i.e., passive vibration of the vocal folds as a result of an increased subglottis pressure. Such a phenomenon of making speech sonorous in the glottis space is called phonation. The filter is the remaining articulators of the speech canal creating resonance spaces. A signal of larynx stimulation is shaped and modulated in these spaces. A final product of this process is called speech.

Pathological changes appearing in the glottis space entail a bigger or smaller impairment of the phonation functions of the larynx. The subject matter of presented research concerns diseases, which appear on the vocal folds, i.e., they have a direct influence on phonation [8].

We are interested in two diseases: Reinke's edema (*Oedema Reinke*) and laryngeal polyp (*Polypus laryngis*).

#### A. Reinke's Edema

Reinke's edema appears often bilaterally, and usually asymmetrically, on the vocal folds. It is created by transudation in a slotted epithelial space of folds devoid of lymphatic vessels and glands, called the Reinke's space. In the pathogenesis of disease, a big role is played by irritation of the laryngeal mucosa by different factors like smoking, excessive vocal effort, inhalatory toxins or allergens. The main symptoms are the following: hoarseness resulting from disturbance of vocal fold vibration or, in the case of large edemas, inspiratory dyspnea. In the case of Reinke's edemas, conservative therapy is not applied. They are microsurgically removed by decortication with saving the vocal muscle.

#### B. Laryngeal Polyp

Laryngeal polyp is a benign tumor arising as a result of gentle hyperplasia of fibrous tissue in mucous membrane of the vocal folds. In the pathogenesis, a big role is played by factors causing chronic larynx inflammation and irritation of the mucous membranes of the vocal folds: smoking, excessive vocal effort, reflux, etc. The main symptoms are the following: hoarseness, aphonia, cough, tickling in the larynx. In case of very big polyps, dyspnea may appear. However, not big polyps may be confused with vocal tumors especially when there is a factor of the load of the patient voice. The polyp may be pedunculated or may be placed on the wide base. If it is necessary, polyps are microsurgically removed with saving a free edge of vocal fold and vocal muscle.

### III. PHONEME APPROXIMATION USING THE BEZIER CURVES

In order to approximate a speech signal (phoneme), we propose to use 4-point Bezier curves. A Bezier curve is a parametric curve very popular in different applications of computer graphics and related fields. A shape of the 4-point Bezier curve is determined by four control points  $P_0 = [P_0^x, P_0^y]$ ,  $P_1 = [P_1^x, P_1^y]$ ,  $P_2 = [P_2^x, P_2^y]$ , and  $P_3 = [P_3^x, P_3^y]$ . The curve interpolates points  $P_0$  and  $P_3$  and approximates points  $P_1$  and  $P_2$ . Two parametric equations determine the shape of the curve:

$$\begin{aligned} x(t) &= (1-t)^3 P_0^x + 3(1-t)^2 t P_1^x + 3(1-t)t^2 P_2^x + t^3 P_3^x, \\ y(t) &= (1-t)^3 P_0^y + 3(1-t)^2 t P_1^y + 3(1-t)t^2 P_2^y + t^3 P_3^y, \end{aligned}$$

where  $t \in [0, 1]$ . Application of the Bezier curves has the following advantages:

- a signal is encoded using a smaller number of values than a number of samples (by one order of magnitude), this can accelerate a learning process of neural networks,

- some noise from the original signal can be removed,
- due to transformations, signals with different magnitudes and gradients can be compared.

The main disadvantage is the complicated process of finding approximation of a signal in the form of a family of Bezier curves.

In this section, we propose an iterative algorithm for finding a family of Bezier curves best approximating a given signal curve. The algorithm inches forward (from the beginning) along the approximated signal curve (corresponding to a given phoneme) trying to find the best Bezier curve approximating the part of the signal curve. At each stage of searching, if a better curve cannot be found, the current Bezier curve is recorded for a covered part of the curve and new searching for the remaining part is started. In this algorithm, a number of Bezier curves approximating a given signal curve is not given in advance. The presented algorithm can be called the stepping algorithm. Algorithm 1 shows formally our procedure. An error  $\varepsilon$  between the Bezier curve and the signal curve in a given interval of samples (from *start* to *end*) is calculated separately for coordinates  $x$  and  $y$  according to the following formulas:

$$\begin{aligned} \varepsilon^x &= \sum_{i=start}^{end} |t_B^i - t_F^i|, \\ \varepsilon^y &= \sum_{i=start}^{end} |v_B^i - v_F^i|, \end{aligned}$$

where  $\{t_B^i, v_B^i\}_i$ ,  $\{t_F^i, v_F^i\}_i$  are sequences of samples of the Bezier curve  $B$  and the signal curve  $F$ , respectively.

The presented algorithm has a polynomial time complexity.

Algorithm 1 uses Algorithm 2 for finding the best moving of a given control point. This algorithm checks an error between the Bezier curve and the original curve. If the error  $\varepsilon$  is worsened (with respect to the previous one  $\varepsilon_p$ ) after moving a control point, then moving is canceled and its direction for the next step is changed to opposite. A number of searching steps is limited (in our experiments to 100).

### IV. RECURRENT NEURAL NETWORKS FOR LEARNING PHONEME PATTERNS

For each Bezier curve represented by four control points  $P_0 = [P_0^x, P_0^y]$ ,  $P_1 = [P_1^x, P_1^y]$ ,  $P_2 = [P_2^x, P_2^y]$ , and  $P_3 = [P_3^x, P_3^y]$ , we calculate three Euclidean distances:

$$\begin{aligned} D_1 &= \sqrt{(P_1^x - P_0^x)^2 + (P_1^y - P_0^y)^2}, \\ D_2 &= \sqrt{(P_2^x - P_0^x)^2 + (P_2^y - P_0^y)^2}, \\ D_3 &= \sqrt{(P_3^x - P_0^x)^2 + (P_3^y - P_0^y)^2}. \end{aligned}$$

Let  $\mathcal{B}$  be a family of Bezier curves approximating a given phoneme. The phoneme is represented by sequences of distances calculated for each Bezier curve. Such sequences are used to learn the modified Elman-Jordan network. The modified Elman-Jordan network has been proposed in [4].

---

**Algorithm 1:** Algorithm for determining a family of Bezier curves approximating a phoneme.

---

**Input** :  $F = [(t_0, v_0), (t_1, v_1), \dots, (t_{n-1}, v_{n-1})]$  - a phoneme (a vector of speech signal samples),  $\tau$  - error threshold,  $c_{max}$  - a maximal number of attempts of searching,  $[s_1^x, s_1^y], [s_2^x, s_2^y]$  - moving vectors, where  $s_1^x, s_1^y, s_2^x, s_2^y \in [0, 1]$

**Output:**  $\mathcal{B}$  - the family of 4-point Bezier curves approximating  $F$ .

```

 $\mathcal{B} \leftarrow \emptyset;$ 
 $start \leftarrow 0; end \leftarrow 1;$ 
 $P_1 \leftarrow null; P_2 \leftarrow null;$ 
 $\varepsilon_{p1} \leftarrow null; \varepsilon_{p2} \leftarrow null;$ 
 $c \leftarrow 0; stop \leftarrow false;$ 
while  $stop = false$  do
   $P_0 \leftarrow (t_{start}, v_{start});$ 
  if  $P_1 = null$  then
     $P_1 \leftarrow (t_{start}, v_{start});$ 
  end
  if  $P_2 = null$  then
     $P_2 \leftarrow (t_{start}, v_{start});$ 
  end
   $P_3 \leftarrow (t_{end}, v_{end});$ 
  Create a curve  $B$  on the basis of  $P_0, P_1, P_2, P_3$ ;
  Calculate an error  $\varepsilon_1$  between  $B$  and  $F$  in the interval  $[t_{start}, t_{end}]$ 
  Move point  $P_1$  using Algorithm 2 by vector  $[s_1^x, s_1^y]$ ;
  Calculate an error  $\varepsilon_2$  between  $B$  and  $F$  in the interval  $[t_{start}, t_{end}]$ ;
  Move point  $P_2$  using Algorithm 2 by vector  $[s_2^x, s_2^y]$ ;
  if  $(\varepsilon_{p1} < \tau \text{ and } \varepsilon_{p2} < \tau)$  or  $(\varepsilon_{p1} < \tau \text{ and } \varepsilon_{p2} < \tau)$  then
    if  $1 + end < n$  then
       $end \leftarrow end + 1;$ 
    else
       $stop = true;$ 
    end
     $\varepsilon_{p1} \leftarrow null; \varepsilon_{p2} \leftarrow null;$ 
     $c \leftarrow 0;$ 
  else
     $c \leftarrow c + 1;$ 
  end
  if  $c > c_{max}$  then
    if  $1 + end < n$  then
       $\mathcal{B} \leftarrow \mathcal{B} \cup \{B\};$ 
       $start \leftarrow end;$ 
       $end \leftarrow start + 1;$ 
       $P_1 \leftarrow null; P_2 \leftarrow null;$ 
       $\varepsilon_{p1} \leftarrow null; \varepsilon_{p2} \leftarrow null;$ 
       $c \leftarrow 0;$ 
    else
       $stop \leftarrow true;$ 
    end
  end
end
Return  $\mathcal{B}$ ;
```

---



---

**Algorithm 2:** Algorithm for moving a control point.

---

**Input** :  $P$  - a control point to be moved,  $\varepsilon_p$  - a last error,  $\varepsilon$  - a current error,  $[s^x, s^y]$  - a moving vector.

**Output:**  $P$  - a moved control point

```

if  $\varepsilon_p = null$  then
   $\varepsilon_p \leftarrow \varepsilon;$ 
end
if  $\varepsilon^x \geq \varepsilon_p^x$  then
   $P^x \leftarrow P^x - s^x;$ 
   $s^x \leftarrow -s^x;$ 
end
if  $\varepsilon^y \geq \varepsilon_p^y$  then
   $P^y \leftarrow P^y - s^y;$ 
   $s^y \leftarrow -s^y;$ 
end
if  $\varepsilon^x < \varepsilon_p^x$  then
   $\varepsilon_p \leftarrow \varepsilon;$ 
end
if  $\varepsilon^y < \varepsilon_p^y$  then
   $\varepsilon_p \leftarrow \varepsilon;$ 
end
 $P^x \leftarrow P^x + s^x;$ 
 $P^y \leftarrow P^y + s^y;$ 
Return  $P$ ;
```

---

Moreover, some learning abilities were tested in [5]. The modified Elman-Jordan network consists of (see Figure 1):

- an input layer,
- a hidden layer,
- a context layer,
- an output layer,
- feedbacks for a hidden layer through the context layer, such feedbacks are used in the Elman networks,
- feedback between an output layer and a hidden layer through the context layer, such feedback is used in the Jordan networks,
- feedback for an output layer.

Output feedback accelerates a learning process and causes seamless modification of weights. Generally, the modified Elman-Jordan network needs a smaller number of epochs (sometimes by 50 per cent) for learning a given pattern (see [5]).

In the approach presented in this paper, we use similar procedure to that presented in [3], [4], and [5]. A difference is that we use as the input for the neural network sequences of distances calculated for Bezier curves approximating an original speech signal instead of samples of that signal. Articulation is an individual patient feature. Therefore, we cannot train a neural network on the independent patterns of phonation of individual vowels. For each patient, parameters of Bezier curves of a speech signal are used for both training and testing of a neural network. The procedure is as follows. We divide the speech signal of an examined patient into time windows corresponding to phonemes. The next step is

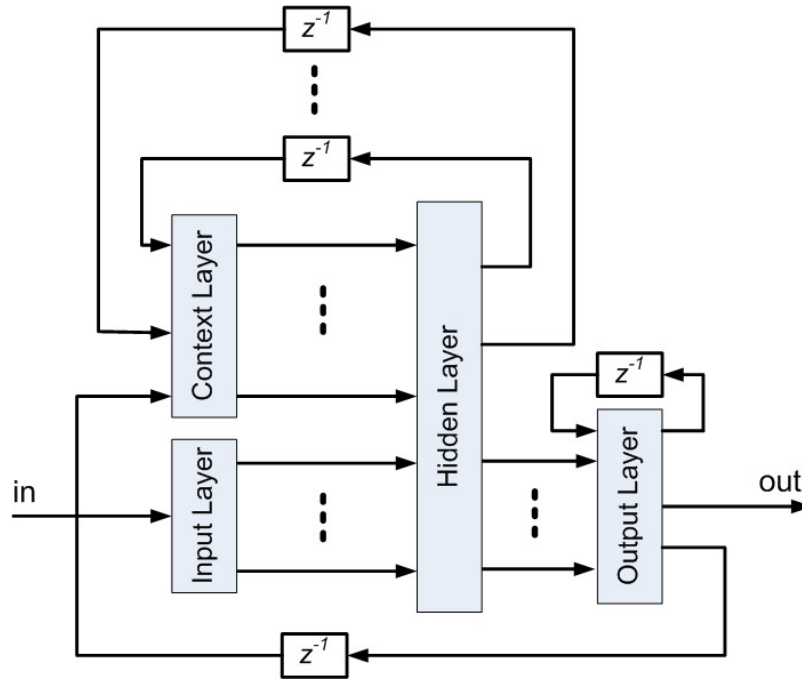


Fig. 1. A structure of the trained Elman-Jordan neural network.

**Algorithm 3:** Algorithm for calculating an average mean squared error corresponding to deformations in a speech signal.

**Input :**  $S$  - a speech signal of a given patient (a vector of samples),  $N$  - a neural network.

**Output:**  $\bar{E}_N$  - an average mean squared error corresponding to deformations in  $S$ .

$W_{all} \leftarrow Div2Win(S)$ ;

$W_{sel} \leftarrow SelWin(W_{all})$ ;

**for each window**  $w \in W_{sel}$  **do**

$B \leftarrow Bezier(w)$ ;

$Train(N, B)$ ;

**for each window**  $w^* \in W_{sel}$  **do**

**if**  $w^* \neq w$  **then**

$B^* \leftarrow Bezier(w^*)$ ;

$E[w^*] \leftarrow MSE(Test(N, B^*))$ ;

**end**

**end**

$\bar{E}[w] \leftarrow Avg(E)$ ;

**end**

$\bar{E}_N \leftarrow Avg(\bar{E})$ ;

**Return**  $\bar{E}_N$ ;

determine a family of Bezier curves approximating it (see Algorithms 1 and 2) and next calculate distances for each curve according to formulas shown at the beginning of this section. Bezier curve parameters of one time window are taken for training the neural network, whereas Bezier curve parameters of the remaining ones, for testing of the neural network. The network learns parameters of a selected time window. If parameters of the remaining windows are similar to the selected one in terms of the time patterns, then for such windows an error generated by the network in a testing stage is small. If significant replication disturbances in time appear for patients with the larynx disease, then an error generated by the network is greater. In this case, the time pattern is not preserved in the whole signal. Therefore, the error generated by the network reflects non-natural disturbances in the patient phonation. Our approach can be expressed formally as it is shown in Algorithm 3. In the algorithm we use the following functions (procedures):

- $Div2Win(S)$  - dividing the speech signal  $S$  into time windows corresponding to phonemes,
- $SelWin(W)$  - selecting randomly a number of time windows from the whole set  $W$ ,
- $Bezier(w)$  - calculating a set of parameters of Bezier curves approximating  $w$ ,
- $Train(N, B)$  - training a neural network  $N$  on a given set  $B$  of parameters of Bezier curves,
- $Test(N, B)$  - testing a neural network  $N$  on a given set  $B$  of parameters of Bezier curves,
- $MSE(E)$  - calculating a mean squared error for the

random selection of a number of time windows. This set of selected windows is used for determining some coefficient characterizing deformations in the speech signal. This coefficient is constituted by an error obtained during testing of the neural network. We propose to use the approach similar to the cross-validation strategy. For each time window we

TABLE I

SELECTED RESULTS OF EXPERIMENTS FOR WOMEN FROM THE CONTROL GROUP OBTAINED USING THE MODIFIED ELMAN-JORDAN NETWORK.

ID	$\overline{E}_{EJN}$	$\overline{n}_{EJN}$
$w_{CG1}$	0.0106	104
$w_{CG2}$	0.0108	103
$w_{CG3}$	0.0120	110
$w_{CG4}$	0.0017	111
$w_{CG5}$	0.0055	99
$w_{CG6}$	0.0159	113
$w_{CG7}$	0.0064	131
$w_{CG8}$	0.0128	113
$w_{CG9}$	0.0128	105
$w_{CG10}$	0.0186	104

TABLE II

SELECTED RESULTS OF EXPERIMENTS FOR WOMEN WITH LARYNGEAL POLYP OBTAINED USING THE MODIFIED ELMAN-JORDAN NETWORK.

ID	$\overline{E}_{EJN}$	$\overline{n}_{EJN}$
$w_{P1}$	0.2184	98
$w_{P2}$	0.0429	71
$w_{P3}$	0.0139	87
$w_{P4}$	0.0201	120
$w_{P5}$	0.0155	132
$w_{P6}$	0.0375	80
$w_{P7}$	0.0148	210
$w_{P8}$	0.0184	94
$w_{P9}$	0.0229	88
$w_{P10}$	0.0462	109

absolute error vector  $E$ :

$$MSE(E) = \frac{1}{n} \sum_{i=1}^n (E_i)^2,$$

where  $n$  is a number of elements in the vector  $E$ ,  $E_i = y(x_i) - z(x_i)$  and  $y(x_i)$  is the obtained output for  $x_i$  whereas  $z(x_i)$  is the desired output for  $x_i$ .

- $Avg(E)$  - calculating an arithmetic average for the vector  $E$  of errors.

## V. EXPERIMENTS

In the experiments, sound samples were analyzed. Samples were recorded for two groups of patients [2]. The first group included patients without disturbances of phonation. They were confirmed by phoniatriest opinion. The second group included patients of Otolaryngology Clinic of the Medical University of Lublin in Poland. They had clinically confirmed dysphonia as a result of Reinke's edema or laryngeal polyp. The information about diseases was received from patients' documentations. Each recording was preceded by a course of breathing exercises with an instruction about a way of articulation. The task of all examined patients was to utter separately Polish vowels: "A", "I", and "U" with extended articulation as long as possible, without intonation, and each on separate expiration. Samples were normalized to the interval [0.0, 1.0] before providing them to the next block. After normalization, samples (as double numbers) were provided to the block calculating the Bezier curve parameters.

In Tables I and II, we present results of experiments carried out using the modified Elman-Jordan network described in

Section IV. Table I includes results for women from the control group as well as Table II includes results for women with laryngeal polyp. Both tables include results for women uttering vowel "A". We give consecutively the average mean squared error  $\overline{E}_{EJN}$  and an average number  $\overline{n}_{EJN}$  of epochs in the training process.

It is easy to see that the modified Elman-Jordan network trained by parameters of Bezier curves approximating the speech signal has some ability to distinct between normal and disease states. The distinction ability presented here would be comparable with abilities obtained if the neural networks were trained using the original speech signals (cf. [4], and [5]). In the approach presented in this paper, we significantly reduce the amount of data to be learned by the neural network. Such observations are very important for further research, especially in the context of a created computer tool for diagnosis of larynx diseases.

## VI. CONCLUSIONS

In the paper, we have shown the classification process of laryngopathies based on a speech signal analysis in the time domain using recurrent neural networks. In our experiments, we have used the modified Elman-Jordan neural network presented in our earlier papers. In the procedure, we have introduced the preprocessing step. In this step, an original signal is approximated using Bezier curves and next the neural network is trained. Bezier curve approximation reduces the amount of data to be learned as well as removes a noise from the original signal. The quality of the proposed method in terms of differentiating normal and pathological categories is not entirely satisfactory, but it shows the direction of further research. In the future, we will concentrate on two directions. The first one is the optimization of the process of finding a family of Bezier curves approximating the speech signal. The second one is an improvement (tuning) of the proposed method for better differentiating between cases belonging to different categories.

## ACKNOWLEDGMENT

This research has been supported by the grant No. N N516 423938 from the Polish Ministry of Science and Higher Education.

## REFERENCES

- [1] R. Greenes, *Clinical Decision Support. The Road Ahead*. Elsevier Inc., 2007.
- [2] J. Warchoł, "Speech examination with correct and pathological phonation using the SVAN 912AE analyser (in Polish)," Ph.D. dissertation, Medical University of Lublin, 2006.
- [3] J. Szkoła, K. Pancerz, and J. Warchoł, "Computer diagnosis of laryngopathies based on temporal pattern recognition in speech signal," *Bio-Algorithms and Med-Systems*, vol. 6, no. 12, pp. 75–80, 2010.
- [4] —, "Computer-based clinical decision support for laryngopathies using recurrent neural networks," in *Proc. of the 10th International Conference on Intelligent Systems Design and Applications (ISDA'2010)*, A. Hassaniien, A. Abraham, F. Marcelloni, H. Hagra, M. Antonelli, and T.-P. Hong, Eds., Cairo, Egypt, 2010, pp. 627–632.

- [5] —, “Improving learning ability of recurrent neural networks: Experiments on speech signals of patients with laryngopathies,” in *Proc. of the International Conference on Bio-inspired Systems and Signal Processing (BIOSIGNALS'2011)*, F. Babiloni, A. Fred, J. Filipe, and H. Gamboa, Eds., Rome, Italy, 2011, pp. 360–364.
- [6] J. Elman, “Finding structure in time,” *Cognitive Science*, vol. 14, pp. 179–211, 1990.
- [7] M. Jordan, “Serial order: A parallel distributed processing approach,” University of California, San Diego, Institute for Cognitive Science, Tech. Rep. 8604, 1986.
- [8] A. Lalvani, *Current Diagnosis and Treatment in Otolaryngology - Head and Neck Surgery*. McGraw-Hill, 2008.
- [9] R. Orlikoff, R. Baken, and D. Kraus, “Acoustic and physiologic characteristics of inspiratory phonation,” *Journal of the Acoustical Society of America*, vol. 102, no. 3, pp. 1838–1845, 1997.
- [10] J. Warchoła, J. Szkoła, and K. Pancerz, “Towards computer diagnosis of laryngopathies based on speech spectrum analysis: A preliminary approach,” in *Proc. of the Third International Conference on Bio-inspired Systems and Signal Processing (BIOSIGNALS'2010)*, A. Fred, J. Filipe, and H. Gamboa, Eds., Valencia, Spain, 2010, pp. 464–467.
- [11] W. Winholtz and I. Titze, “Suitability of minidisc (MD) recordings for voice perturbation analysis,” *Journal of Voice*, vol. 12, no. 2, pp. 138–142, 1998.



# Validation of Data Categorization Using Extensions of Information Systems: Experiments on Melanocytic Skin Lesion Data

Łukasz Piątek, Krzysztof Pancerz, Grzegorz Owsiany  
 Institute of Biomedical Informatics  
 University of Information Technology and Management  
 Rzeszów, Poland

Email: lpiatek@wsiz.rzeszow.pl, kpancerz@wsiz.rzeszow.pl, gowsiany@wsiz.rzeszow.pl

**Abstract**—The purpose of data categorization is to group similar cases (items, examples, objects, etc.) together under a common label so that information can be acted upon in the aggregate form. Sometimes, this process is made arbitrary by an expert. For each case, an expert determines a class (group) to which the case is classified. In the paper, we propose a method for validation of a categorization process. The method is based on extensions of information systems defined in terms of rough sets. Usefulness of the proposed method is shown for the data used in the synthesis of images of melanocytic skin lesions.

**Index Terms**—extensions of information systems, data categorization, rough sets, synthesis of images

## I. INTRODUCTION

**A**N INFORMATION system proposed by Z. Pawlak [1] can represent a finite set of cases described by attributes. Each attribute represents one of features of cases. Besides all cases appearing in the original information system, an extension of it can include new cases, which have not been observed yet, but which are consistent to a certain degree with the knowledge included in the original system. The knowledge can be represented in the form of rules (production, association, etc.). One can consider only the so called consistent extensions of information systems [2], when all new cases are totally consistent with the knowledge included in the original information systems. However, in general case, we can consider partially consistent extensions [3], when some new cases are consistent with the knowledge possessed, only to a certain degree. The important problem is to determine consistency factors of new cases taking into consideration different ways of knowledge representation.

In this paper, we use an approach to computing consistency factors based on rough set theory and proposed in [3]. The algorithm using that approach has been presented in [4]. In Section III we recall it. This algorithm makes use of important results of research on extensions of information systems given in [5]. In the algorithm, we assume that the knowledge included in an original information system  $S$  is expressed by minimal rules true and realizable in  $S$  (see Section III). Computing a consistency factor for a given object is based on determining importance (relevance) of rules extracted from the

system  $S$  which are not satisfied by the new case. We assume that if the importance of these rules is greater the consistency factor of a new object with the knowledge is smaller. The importance of a set of rules not satisfied by the new case is determined by means of a strength factor of this set of rules in  $S$ .

As an example, we consider phenomena related to melanocytic skin lesions. In the process of validation of categorization of cases for respective types of lesions, we can use extensions of information systems. A consistency factor calculated for a given text vector (case) representing combination of colors and structures can be considered in terms of possibility theory proposed by L. Zadeh [6]. This factor enables us to answer a question about possibility with which a given combination of colors and structures appears in a respective melanocytic skin lesion. If we have an information system consisting of information about combinations of colors and diversities of structures appearing for given lesions, it means that we have collected all cases observed until now. In many situations, we have observed only a part of all possible combinations of colors and diversities of structures. In this case, if we take a new combination which has not been observed yet, then the following question  $Q$  arises: "Is it possible that the combination (case) will appear for a given lesion?". In this question, "possible" means "plausible". We wish to answer the question  $Q$  on the basis of the possessed information collected until now in the information system  $S$  (observed combinations for a given lesion). We can determine a possibility distribution on all combinations of colors and diversities of structures.

We can identify consistency factors of cases in the extension of a given information system with a possibility distribution of cases. Possibility theory [6] is a framework for representing vague and incomplete knowledge. Let  $X$  be a set that represents the range of a variable  $x$ . A possibility distribution  $\pi_x$  on  $X$  is a mapping from  $X$  to the unit interval  $[0, 1]$  attached to the variable  $x$ . The variable  $x$  can be treated as some phenomenon  $\mathcal{P}$ . The range of  $x$  represents the set of cases of  $\mathcal{P}$ . The function  $\pi_x$  distinguishes which case of  $\mathcal{P}$  is plausible and which one is less plausible. Let  $u \in X$ ,  $\pi_x(u) = 0$  means

that  $x = u$  is impossible.  $\pi_x(u) = 1$  means that  $x = u$  is totally possible, i.e., plausible. The quantity  $\pi_x(u)$  represents the degree of possibility of the assignment  $x = u$ .

## II. PROBLEM BACKGROUND

In our research, we use a database on melanocytic skin lesions including 548 cases belonging to one of four types of lesions: *Benign nevus*, *Blue nevus*, *Suspicious nevus*, and *Melanoma malignant*. Each case in a database is recorded as a 15-element text vector constituting the input information in the process of synthesis of static images of melanocytic skin lesions (cf. [7], [8], [9]). We have noticed at the current research state that a key role is played by two features of melanocytic skin lesions, namely, *Color* and *Diversity of structure*. These features have a multivalued character and describe the presence or absence of colors and diversities of structures allowed by the ABCD rule [10]. *Color* can have six allowed values: *black*, *blue*, *dark-brown*, *light-brown*, *red*, and *white*, whereas *Diversity of structure* can have five allowed values: *branched streaks*, *pigment dots*, *pigment globules*, *pigment network*, and *structureless area*. Assessment of a tinge of a skin lesion consists in differentiating any number of colors (from the set of six allowed colors). Assessment of structural elements in a skin lesion consists in determining any number of structures (from the set of five allowed diversities of structures). The remaining attributes of the text vector are *Assymetry*, *Border*, and *TDS (Total Dermatoscopy Score)*. In Table I, we present the collation of information about attributes of the text vector used in our experiments. For each allowed *color* and *diversity of structure*, we have one attribute taking one of logical values: 1 (denoting presence) or 0 (denoting absence).

TABLE I  
ATTRIBUTES CORRESPONDING TO FEATURES: *Color* AND *Diversity of structure*

Feature	Attribute	Value set
<i>Color</i>	<i>black</i>	{0, 1}
	<i>blue</i>	{0, 1}
	<i>dark-brown</i>	{0, 1}
	<i>light-brown</i>	{0, 1}
	<i>red</i>	{0, 1}
	<i>white</i>	{0, 1}
<i>Diversity of structure</i>	<i>branched streaks</i>	{0, 1}
	<i>pigment dots</i>	{0, 1}
	<i>pigment globules</i>	{0, 1}
	<i>pigment network</i>	{0, 1}
	<i>structureless area</i>	{0, 1}

Synthesis of colors and structures of a lesion should consider multi-valued character of *Color* and *Diversity of structure* features, capable to create a considerable number of combinations of these parameters, which can simultaneously appear in a given lesion.

According to the ABCD rule, a number of all possible mappings of colors and structures is  $2^{11} - 95$ . 95 is a number of inadmissible combinations. Inadmissible combinations are: combinations without any color (64 cases) and combinations without any structure (32 cases). One of combinations does not have both any color and any structure, hence 95. A real number

of mappings can be greater in view of different mappings of given colors in respective structures. The ABCD rule does not define such mappings. Therefore, a number of all possible combinations is  $2^{30} + 1$  minus 95 inadmissible combinations. Generating over one billion textures is virtually almost impossible. An additional disadvantage of such mapping would be a very frequent repetition of occurrence of selected structures in particular colors, because of its shape, size and place in the generated image. Taking all these circumstances into consideration the synthesis requires a special approach, we should initially (before generating textures) find which colors and structures occur simultaneously in real lesions.

In the process of validation of choices of combinations for respective types of lesions, we use extensions of information systems described in the next section. A consistency factor calculated for a given text vector representing the combination of colors and structures is considered in terms of possibility theory. This factor enables us to answer a question about possibility with which a given combination of colors and structures appears in a respective melanocytic skin lesion.

## III. EXTENSIONS OF INFORMATION SYSTEMS

In this section, we recall crucial notions concerning rough sets, information systems, rules, as well as, extensions of information systems. For more exact description and explanation we refer readers to [5], [11].

An information system is an ordered pair  $S = (U, A)$ , where  $U$  is a non-empty, finite set of objects which is also called universum,  $A$  is a non-empty, finite set of attributes. Each attribute  $a \in A$  is a total function  $a : U \rightarrow V_a$ , where  $V_a$  is a set of values of the attribute  $a$ . Each information system  $S = (U, A)$  can be presented in the form of a data table. Columns are labeled with attributes from  $A$  whereas rows are labeled with objects from  $U$ . Cells of the table include values of appropriate attributes. A decision system is an information system  $S = (U, A)$ , where  $A = C \cup D$  and  $C \cap D = \emptyset$ .  $C$  is a set of condition attributes (in short, conditions) whereas  $D$  is a set of decision attributes (in short, decisions).

Let  $S = (U, A)$  be an information system. Each subset  $B \subseteq A$  of attributes determines an equivalence relation on  $U$ , called an *indiscernibility relation*  $Ind(B)$ , defined as

$$Ind(B) = \{(u, v) \in U \times U : \forall a \in B a(u) = a(v)\}.$$

The equivalence class containing  $u \in U$  will be denoted by  $[u]_B$ .

Let  $X \subseteq U$  and  $B \subseteq A$ . The *B-lower approximation*  $\underline{B}X$  of  $X$  and the *B-upper approximation*  $\overline{B}X$  of  $X$  are defined as

$$\underline{B}X = \{u \in U : [u]_B \subseteq X\}$$

and

$$\overline{B}X = \{u \in U : [u]_B \cap X \neq \emptyset\},$$

respectively.

Dependencies among values of attributes in an information system may be expressed by means of the so-called rules. Each rule  $\rho$  considered by us in the information system  $S = (U, A)$

has the form  $(a_{i_1}, v_{i_1}) \wedge (a_{i_2}, v_{i_2}) \wedge \dots \wedge (a_{i_r}, v_{i_r}) \Rightarrow (a_d, v_d)$ , where  $a_d \in A$  and  $v_d \in V_{a_d}$ , while  $a_{i_j} \in B \subseteq A - \{a_d\}$  and  $v_{i_j} \in V_{a_{i_j}}$  for  $j = 1, 2, \dots, r$ . The rule  $\rho$  is a satisfiable (true) rule if for each object  $u \in U$ : if  $a_{i_1}(u) = v_{i_1} \wedge a_{i_2}(u) = v_{i_2} \wedge \dots \wedge a_{i_r}(u) = v_{i_r}$ , then  $a_d(u) = v_d$ . The rule  $\rho$  is a minimal rule if removing any atomic formula  $(a_{i_j}, v_{i_j})$ , where  $j = 1, 2, \dots, r$ , from the predecessor of a rule makes this rule not true in  $S$ . The rule  $\rho$  is a realizable rule if there exists any object  $u \in U$  such that  $a_{i_1}(u) = v_{i_1} \wedge a_{i_2}(u) = v_{i_2} \wedge \dots \wedge a_{i_r}(u) = v_{i_r}$ . A set of all minimal rules satisfiable and realizable in the information system  $S$  will be denoted by  $Rul(S)$ . By  $Rul_a(S)$  we will denote the set of all rules from  $Rul(S)$  having an atomic formula containing the attribute  $a$  in their successors.

Let  $S = (U, A)$  be an information system. An information system  $S^* = (U^*, A^*)$  is an extension of  $S$  if and only if:

- $U \subseteq U^*$ ,
- $card(A) = card(A^*)$ ,
- for each  $a \in A$ , there exists  $a^* \in A^*$  such that a function  $a^* : U^* \rightarrow V_{a^*}$  is an extension of a function  $a : U \rightarrow V_a$  to  $U^*$ .

We may admit also situation when  $a^* : U^* \rightarrow V_{a^*}$ , where  $V_{a^*} \subset V_a$ , for any  $a^* \in A^*$ . It means that we can add new objects to a given information system  $S$  that have new values of attributes not existing yet in  $S$ . If  $V_{a^*} = V_a$  for each  $a^* \in A^*$ , then  $S^*$  will be called a proper extension of  $S$ , otherwise  $S^*$  will be called a non-proper extension of  $S$ .

A set  $A^*$  of attributes in the extension  $S^* = (U^*, A^*)$  of an information system  $S = (U, A)$  can be also denoted by  $A$  like in the original system  $S$ . So, we write  $S^* = (U^*, A)$  instead of  $S^* = (U^*, A^*)$ . The same applies to attributes of  $A^*$ , i.e.,  $a_1^*, a_2^*, \dots, a_m^* \in A^*$ . So, we write  $a_1, a_2, \dots, a_m$  instead of  $a_1^*, a_2^*, \dots, a_m^*$ , where  $a_1, a_2, \dots, a_m \in A$ .

For any object  $u$  from the extension  $S^*$  of a given information system  $S$ , we define a coefficient called a consistency factor. This coefficient expresses a degree of consistency of  $u$  with the knowledge (expressed by  $Rul(S)$ ) included in the original system  $S$ . The procedure for computing a consistency factor is described here.

For each attribute  $a \in A$  of a given information system  $S$  and a new object  $u^*$  added to  $S$  we can translate an information system  $S$  into the information system  $S_{a,u^*} = (U_a, C_a \cup \{a\})$  with irrelevant values of attributes. Such a system will be called the  $a$ - $u^*$  match of  $S$ .

Let  $S = (U, A)$  be an information system,  $S^* = (U^*, A)$  its extension, and  $u^* \in U^*$  a new object from the extension  $S^*$ . The  $a$ - $u^*$  match of  $S$  is an information system  $S_{a,u^*} = (U_a, C_a \cup \{a\})$  with irrelevant values of attributes created in the following way. Each attribute  $c' \in C_a$  corresponds exactly to one attribute  $c \in A - \{a\}$ . Each object  $u' \in U_a$  corresponds exactly to one object  $u \in U$  and moreover:

$$c'(u') = \begin{cases} c(u) & \text{if } c(u) = c(u^*) \\ * & \text{otherwise} \end{cases}$$

for each  $c' \in C_a$ , and  $a(u') = a(u)$ .

If we create the  $a$ - $u^*$  match of  $S$ , then we create a new information system for which appropriate sets of attribute values are extended by the value  $*$ . The symbol  $*$  means that a given value of the attribute is not relevant.

For simplicity, the attribute  $c \in A - \{a\}$  in  $S$  and the attribute  $c'$  in  $S_{a,u^*}$  corresponding to  $c$  will be marked with the same symbol, i.e.,  $c'$  will be marked in  $S_{a,u^*}$  with  $c$ .

The system  $S_{a,u^*}$  can be treated as a decision system with condition attributes constituting the set  $C_a$  and the decision attribute  $a$ .

For the information system  $S_{a,u^*} = (U_a, C_a \cup \{a\})$ , we define a characteristic relation  $R(C_a)$  similarly to the definition of a characteristic relation in information systems with missing attribute values (cf. [12]).  $R(C_a)$  is a binary relation on  $U_a$  defined as follows  $R(C_a) = \{(u, v) \in U_a \times U_a : \exists c \in C_a c(u) \neq * \text{ and } \forall c \in C_a (c(u) \neq *) \Rightarrow (c(u) = c(v))\}$ . For each  $u \in U_a$ , a characteristic set  $K_{C_a}(u)$  has the form  $K_{C_a}(u) = \{v \in U_a : (u, v) \in R(C_a)\}$ . Let  $X \subseteq U_a$ . The  $C_a$ -lower approximation of  $X$  is determined as  $\underline{C_a}X = \{u \in U_a : K_{C_a}(u) \neq \emptyset \text{ and } K_{C_a}(u) \subseteq X\}$ . Let  $S = (\overline{U}, A)$  be an information system,  $a \in A$ , and  $v_a \in V_a$ . By  $X_a^{v_a}$  we denote the subset of  $U$  such that  $X_a^{v_a} = \{u \in U : a(u) = v_a\}$ .

Let  $S = (U, A)$  be an information system,  $S^* = (U^*, A)$  its extension,  $u^* \in U^*$  a new object from the extension  $S^*$ , and  $a \in A$ . The object  $u^*$  satisfies a rule  $\rho \in Rul_a(S)$  if and only if for each  $v_a \in V_a$  if  $\underline{C_a}X_a^{v_a} \neq \emptyset$ , then  $a(u^*) = v_a$ . A proof can be found in [4].

An approach recalled here does not involve computing any rules from an original information system. The algorithm presented here allows us to determine a set of objects from an original information system  $S$  supporting minimal rules from  $Rul(S)$ , but not satisfied by the object  $u^*$ . A consistency factor is computed as a complement to 1 of the strength of the set of rules not satisfied. Let  $S = (U, A)$  be an information system,  $\{V_a\}_{a \in A}$  is a family of sets of attribute of values in  $S$ ,  $Rul(S)$  a set of all minimal rules true and realizable in  $S$ ,  $S^* = (U^*, A)$  an extension of  $S$ , and  $u^* \in U^*$ . The consistency factor  $\xi_S(u^*)$  of  $u^*$  with the knowledge (expressed by  $Rul(S)$ ) is defined as follows:

$$\xi_S(u^*) = \xi'_S(u^*) \omega_S(u^*),$$

where:

- $\xi'_S(u^*) = 1 - \frac{card(\tilde{U})}{card(U)}$  is a proper consistency,
- $\omega_S(u^*) = \frac{card(\{a \in A : a(u^*) \in V_a\})}{card(A)}$  is a resemblance factor determining some affinity between the object  $u^*$  and objects from  $S$  with respect to values of attributes,

and  $\tilde{U} = \bigcup_{a \in A} \bigcup_{v_a \in V_a} \{\underline{C_a}X_a^{v_a} : \underline{C_a}X_a^{v_a} \neq \emptyset \wedge a(u^*) \neq v_a\}$ .

The estimated time complexity of the algorithm has the form:

$$\Theta(|A|^2|U| + |A||V_a||U|^2),$$

where  $|A|$  is the cardinality of a set of attributes,  $|U|$  is the cardinality of a set of cases,  $|V_a|$  is the maximal cardinality of a set of attribute values.

---

**Algorithm 1:** Algorithm for computing a consistency factor  $\xi_S(u^*)$  of  $u^*$  with the knowledge expressed by  $Rul(S)$

---

**Input :** An information system  $S = (U, A)$ , a new object  $u^*$  added to  $S$ .

**Output:** A consistency factor  $\xi_S(u^*)$  of  $u^*$  with the knowledge expressed by  $Rul(S)$ .

$\tilde{U} \leftarrow \emptyset;$

$i \leftarrow 0;$

**for each**  $a \in A$  **do**

**if**  $a(u^*) \notin V_a$  **then**

$i \leftarrow i + 1;$

**end**

  Create the match  $S_{a,u^*} = (U_a, C_a \cup \{a\})$  of  $S$ ;

**for each**  $v_a \in V_a$  **do**

$X_a^{v_a} \leftarrow \{u \in U : a(u) = v_a\};$

**if**  $C_a X_a^{v_a} \neq \emptyset$  **then**

**if**  $a(u^*) \neq v_a$  **then**

$\tilde{U} \leftarrow \tilde{U} \cup C_a X_a^{v_a};$

**end**

**end**

**end**

$\xi'_S(u^*) \leftarrow 1 - \frac{\text{card}(\tilde{U})}{\text{card}(U)};$

$\omega_S(u^*) \leftarrow \frac{i}{\text{card}(A)};$

$\xi_S(u^*) \leftarrow \xi'_S(u^*) \omega_S(u^*);$

**end**

**return**  $\xi_S(u^*);$

---

#### IV. EXPERIMENTS

In our experiments, we have used a database on melanocytic skin lesions including 548 cases, each belonging to one of four types of lesions:

- *Benign nevus* - 248 cases,
- *Blue nevus* - 78 cases,
- *Suspicious nevus* - 108 cases,
- *Melanoma malignant* - 114 cases.

Categories have been assigned to cases by clinicians.

The experiments have been carried out according to Procedure 2. We have distinguished two tests:

- 1) One information system containing cases belonging to the same category (melanocytic skin lesion) was an original information systems  $S$ , another one containing cases belonging to another category was treated as an extension of  $S$  (possibly non-proper).
- 2) An information system containing cases belonging to the same category (melanocytic skin lesion) was split randomly into two disjoint subsystems. The first part (greater) constituted an original system, the second one - its extension (possibly non-proper).

For each case from the extension of the original information system  $S$ , we have calculated a consistency factor with the knowledge included in  $S$  according to Algorithm 1. In the

---

#### Procedure for experiments

---

**Input :** An original (training) information system  $S_{train} = (U_{train}, A)$ , a testing information system  $S_{test} = (U_{test}, A)$ .

**Output:** A testing information system  $S_{test}$  with consistency factors assigned to objects in  $S_{test}$  with the knowledge expressed by  $Rul(S_{train})$  using Algorithm 1.

**for each**  $u \in U_{train}$  **do**

  Calculate a consistency factor  $\xi_S(u)$  of  $u$  with the knowledge expressed by  $Rul(S_{train})$  according to Algorithm 1;

  Assign  $\xi_S(u)$  to  $u$ ;

**end**

**return**  $S_{test};$

---

approach presented in this paper, we expect the following situations:

- 1) If a given case belongs to a different category from the category of cases in the original system, then the consistency factor should be smaller.
- 2) If a given case belongs to the same category as cases in the original system, then the consistency factor should be greater.

In Table II and III, we present aggregated results (average consistency factors) of our experiments. Results have partly confirmed our expectations. Some exceptions can indicate two directions for further research:

- 1) Some combinations (cases) of colors and diversities of structures used in the synthesis of images of melanocytic skin lesions are incorrectly categorized. It is indication that an informational data base should be verified.
- 2) The proposed method needs some improvement (tuning) for better differentiating between cases belonging to different categories.

#### V. CONCLUSION

In the paper, we have proposed a method for validation of a categorization process. The method is based on extensions of information systems defined in terms of rough sets. Usefulness of the proposed method has been shown for the data used in the synthesis of images of melanocytic skin lesions. Obtained results have indicated directions for further research. The first direction is a verification of an informational data base of combinations of colors and diversities of structures used in the synthesis of images of melanocytic skin lesions. The second one concerns further developing of the proposed methodology.

#### ACKNOWLEDGMENTS

This paper has been partially supported by the grant No. N 516 482640 from the National Science Centre in Poland.

TABLE II  
RESULTS OF EXPERIMENTS CARRIED OUT FOR CASES BELONGING TO DIFFERENT CATEGORIES

Original information system	Tested information system	Average coefficient factor
<i>Benign nevus</i>	<i>Blue nevus</i>	0.28
<i>Benign nevus</i>	<i>Suspicious nevus</i>	0.87
<i>Benign nevus</i>	<i>Melanoma malignant</i>	0.74
<i>Blue nevus</i>	<i>Benign nevus</i>	0.30
<i>Blue nevus</i>	<i>Suspicious nevus</i>	0.39
<i>Blue nevus</i>	<i>Melanoma malignant</i>	0.41
<i>Suspicious nevus</i>	<i>Benign nevus</i>	0.64
<i>Suspicious nevus</i>	<i>Blue nevus</i>	0.27
<i>Suspicious nevus</i>	<i>Melanoma malignant</i>	0.72
<i>Melanoma malignant</i>	<i>Benign nevus</i>	0.61
<i>Melanoma malignant</i>	<i>Blue nevus</i>	0.36
<i>Melanoma malignant</i>	<i>Suspicious nevus</i>	0.81

TABLE III  
RESULTS OF EXPERIMENTS CARRIED OUT FOR CASES BELONGING TO THE SAME CATEGORIES

Original information system	Tested information system	Average coefficient factor
<i>Benign nevus</i>	<i>Benign nevus</i>	0.89
<i>Blue nevus</i>	<i>Blue nevus</i>	0.74
<i>Suspicious nevus</i>	<i>Suspicious nevus</i>	0.82
<i>Melanoma malignant</i>	<i>Melanoma malignant</i>	0.84

#### REFERENCES

- [1] Z. Pawlak, *Rough Sets - Theoretical Aspects of Reasoning About Data*. Dordrecht: Kluwer Academic Publishers, 1991.
- [2] Z. Suraj, "Some remarks on extensions and restrictions of information systems," in *Rough Sets and Current Trends in Computing*, ser. Lecture Notes in Artificial Intelligence, W. Ziarko and Y. Yao, Eds. Berlin Heidelberg: Springer Verlag, 2001, vol. 2005, pp. 204–211.
- [3] Z. Suraj, K. Pancerz, and G. Owsiany, "On consistent and partially consistent extensions of information systems," in *Proc. of the RSFD-GrC'2005*, ser. Lecture Notes in Artificial Intelligence, D. Ślęzak et al., Eds. Berlin Heidelberg: Springer Verlag, 2005, vol. 3641, pp. 224–233.
- [4] K. Pancerz, "Extensions of information systems: The rough set perspective," ser. Lecture Notes in Computer Science, J. Peters, A. Skowron, M. Chakraborty, W.-Z. Wu, and M. Wolski, Eds. Berlin Heidelberg: Springer-Verlag, 2009, vol. 5656, pp. 157–168.
- [5] M. Moshkov, A. Skowron, and Z. Suraj, "On testing membership to maximal consistent extensions of information systems," in *Rough Sets and Current Trends in Computing*, ser. Lecture Notes in Artificial Intelligence, S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H. S. Nguyen, and R. Slowinski, Eds. Berlin Heidelberg: Springer-Verlag, 2006, vol. 4259, pp. 85–90.
- [6] L. Zadeh, "Fuzzy sets as the basis for a theory of possibility," *Fuzzy Sets and Systems*, vol. 1, pp. 3–28, 1978.
- [7] Z. Hippe and Ł. Piątek, "Synthesis of static medical images an example of melanocytic skin lesions," in *Computer Recognition Systems 2*, M. Kurzyński, E. Puchała, M. Woźniak, and A. Żolnierek, Eds. Berlin Heidelberg: Springer-Verlag, 2007, pp. 503–509.
- [8] Z. Hippe, J. Grzymala-Busse, and Ł. Piątek, "Synthesis of medical images in the domain of melanocytic skin lesions," in *Information Technologies in Biomedicine*, E. Piętka and J. Kawa, Eds. Berlin Heidelberg: Springer-Verlag, 2008, pp. 225–231.
- [9] —, "From the research on synthesis of static images of melanocytic skin lesions," in *Computers in Medical Activity*, E. Kącki, M. Rudnicki, and J. Stempczyńska, Eds. Berlin Heidelberg: Springer-Verlag, 2009, pp. 223–229.
- [10] Z. Hippe, "Computer database NEVI on endangerment by melanoma," *TASK Quarterly*, vol. 3, no. 4, pp. 483–488, 1999.
- [11] K. Pancerz, "Some issues on extensions of information and dynamic information systems," in *Foundations of Computational Intelligence, Volume 5: Function Approximation and Classification*, ser. Studies in Computational Intelligence, A. Abraham and S. V. Hassanien, A.E., Eds. Berlin Heidelberg: Springer-Verlag, 2009, vol. 205, pp. 79–106.
- [12] J. W. Grzymala-Busse, "Data with missing attribute values: Generalization of indiscernibility relation and rule induction," in *Transactions on Rough Sets I*, J. Peters, J. Grzymala-Busse, B. Kostek, R. Swiniarski, and M. Szczuka, Eds. Berlin: Springer-Verlag, 2004, pp. 78–95.



# Interval-based Attribute Evaluation Algorithm

Mostafa A. Salama<sup>1</sup>, Nashwa El-Bendary<sup>2</sup> Aboul Ella Hassanien<sup>3</sup>, Kenneth Revett<sup>1</sup>, Aly A. Fahmy<sup>3</sup>

<sup>1</sup>*Department of Computer Science, British University in Egypt, Cairo, Egypt*  
*Email: mostafa.salama, ken.revett@bue.edu.eg*

<sup>2</sup>*Arab Academy for Science, Technology, and Maritime Transport, Cairo, Egypt*  
*Email: nashwa m@aast.edu*

<sup>3</sup>*Cairo University, Faculty of Computers and Information, Cairo, Egypt*  
*Email: aboitcairo, aly.fahmy@gmail.com*

**Abstract**—Attribute values may be either discrete or continuous. Attribute selection methods for continuous attributes had to be preceded by a discretization method to act properly. The resulted accuracy or correctness has a great dependance on the discretization method. However, this paper proposes an attribute selection and ranking method without introducing such technique. The proposed algorithm depends on a hypothesis that the decrease of the overlapped interval of values for every class label indicates the increase of the importance of such attribute. Such hypothesis were proved by comparing the results of the proposed algorithm to other attribute selection algorithms. The comparison between different attribute selection algorithms is based on the characteristics of relevant and irrelevant attributes and their effect on the classification performance. The results shows that the proposed attribute selection algorithm leads to a better classification performance than other methods. The test is applied on medical data sets that represent a real life continuous data sets.

**Index Terms**—Attribute selection; Classification, ChiMerge.

## I. INTRODUCTION

ONE OF the major problems in data mining tools is the curse of dimensionality, several attribute reduction algorithms have been developed to solve such problem. The high number of attributes may contain irrelevant or redundant attributes to the classification methods [1]. Attribute reduction algorithms are either attribute selection or attribute extraction algorithms. Attribute selection algorithms determine the importance of the attributes according to the class labels. The first selected attribute that got the highest rank is the most relevant attribute to the class labels, then the relevance degree decreases until the least ranked attribute. If the classifier is applied only on attributes of the highest ranked attributes, the accuracy of the classifier should be better than being applied on all attributes. The reason of the decrease in accuracy when using all attributes is that the attributes with the lowest ranks have a negative impact on the classification result. An interesting observation that appears in [2], [3] that the trend of the classification accuracy of the classifier applied after the attribute selection algorithm increases until a certain peak where the most relevant attributes are used attributes. Then the classification accuracy starts to decrease which shows the effect of the irrelevant attributes, attributes with the lowest ranks, on the classifier.

The input data sets are either contain attributes of continuous values, discrete values or both types. For continuous data sets, attribute selection algorithms like chi-square, gain ration and information gain have to be preceded by a discretization method [4]. The correctness of the selected attributes has a great dependence on such discretization method. The proposed method here is an attribute selection and ranking method of continuous attributes that does not need to be preceded by a discretization method. It depends on a hypothesis that as the non-overlapped interval of values between the classes labels of an attribute increases as the importance of this attribute increases. This algorithm calculates the number of values in these non-overlapped interval for each attribute and accordingly creates a ranking vector. The proposed algorithm will be compared to other algorithms through checking which attribute selection algorithm would lead to the maximum classification accuracy with the least number of attributes. And hence, two numbers will be used in the comparison which are the number of attributes of highest classification accuracy and the value of this accuracy.

The proposed algorithm will be applied on two different medical real life data sets which are the Indian diabetes and HCV data sets. The classification methods used are the Multi-layered Perceptron and Support vector machine implemented by Weka software.

The rest of this paper is organized as follows: Section II shows the proposed interval-based attribute selection algorithm. Classification results and comparisons with different attribute selection algorithms illustrated in section III. Conclusion and future work is discussed in section IV.

## II. THE INTERVAL-BASED ATTRIBUTE SELECTION ALGORITHM

The interval-based attribute selection algorithm depends on a hypothesis that as the intersection between attribute value ranges of different class labels decreases as the importance of this attribute increases. The reason of this hypothesis is that if an attribute has a certain continuous range of values appears only in the case of a certain class label, then this attribute can help as an indication to this class label. Moreover, as the length of this kind of ranges increases as the importance of this

attribute increases. Figure (1) shows an attribute that contains an interval for each class.

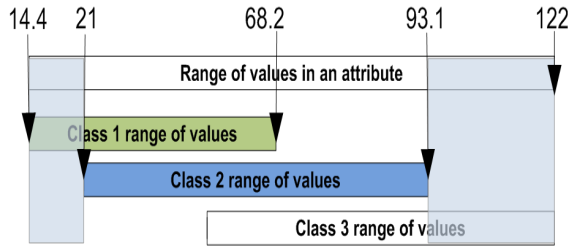


Fig. 1. Non overlapping intervals

The dashed areas show the ranges of values that are not overlapped between multiple class, only a single class label is assigned to this label. In order to evaluate the importance of such attribute, the number of values in ranges that falls in a single class will be calculated for every class and summed. i.e. as shown in figure 1, the number of values that falls in the dashed areas are counted. Then the resulted value, after refinement of this count as shown in the equation 5, will be considered as the attribute rank among other attributes.

$$\mu_a = \frac{1}{n} * \sum_{c \in C} \frac{n_{ci}}{n_c} \quad (1)$$

$\mu_a$  represents the rank of attribute  $a$ ,  $n$  is the number of objects in the data set,  $n_c$  is the number of values where the corresponding objects are of class label  $c$ , and  $n_{ci}$  is the number of values in a rang that is completely falls in class  $c$  where this range is not overlapped with other class labels. For a two class data set, algorithm(1) can be used to calculate the interval-based ranking value which is  $\mu_a$ . The algorithm detects, for every class, the range of values for an attribute that are not in the class and hence counts the number of objects in that range.

The removal of misleading values in Algorithm(2) is an optional step as it depends on the collection methodologies whether it is accurate or not. This step decrease the sensitivity to outliers by removing the values that are most far away from the average of the attribute values. This step should remove only a small percentage of the values in the attribute in order not to affect the accuracy of the results.

### III. EXPERIMENTAL RESULTS AND DISCUSSIONS

#### A. Comparison to other attribute evaluation algorithms

A comparison is applied according to such behavior between different attribute selection algorithms and the proposed interval-based algorithm. The best attribute selection algorithm should follow the following criteria:

- The maximum classification accuracy (peak) is reached with the smallest number of attributes.
- This peak should have the highest value among other attribute evaluation algorithms.

The comparison will be applied through the train and test of the data set multiple times, where in the first time, the data set

---

#### Algorithm 1 Calculate Interval-based rank $\mu_a$ of attribute $a$

---

$\mu_a$  : Attribute  $a$ 's rank, initial value is 0  
*AttributeLength* : Number of objects  
 $x_a$  and  $n_a$  : max and min values of attribute  $a$   
**for** Each Class label  $c$  **do**  
  Remove misleading values.  
  Determine the interval that represent the range of values of the attribute in that class label.  
  *IntervalLength* : Number of objects in class  $c$   
   $x_{ac}$  and  $n_{ac}$  : The max and min values of this interval .  
  //Calculate the number of values outside the interval range.  
   $\mu_c$  : Initial value is 0  
  **for** Each value  $v$  in attribute  $a$  **do**  
    **if**  $v < n_{ac}$  **or**  $v > x_{ac}$  **then**  
       $\mu_c = \mu_c + 1$ .  
    **end if**  
  **end for**  
   $\mu_c = \mu_c / \text{IntervalLength}$   
   $\mu_a = \mu_a + \mu_c$   
**end for**  
 $\mu_a = \mu_a / \text{AttributeLength}$

---



---

#### Algorithm 2 Remove percentage $x$ of misleading

---

Input : *Interval* values of an attribute  $a$  for objects lies in class  $c$   
Output : *avg* average of the values of an attribute  $a$  in a class  $c$   
**for**  $x * \text{IntervalLength}$  values **do**  
  Remove the value of max difference from the average *avg* .  
**end for**

---

will contain only the most single relevant attribute, then the number of attributes will be incrementally increasing until the all the attributes are used. This algorithm could be considered as a semi-wrapper method as the evaluation will be applied only on a certain subset of attributes, where the number of this subsets is equal to the number of attributes. The wrapper-based approaches employ induction classifier as a black box using cross-validation or bootstrap techniques. A method has been previously proposed to compare between different attribute selection algorithms through applying genetic algorithm to evaluate the feature subset candidates suggested by different attribute selection algorithms [11]. This algorithm had solved the problem of the high computation but the problem of dealing with the data sets of continuous attributes still a problem in the used attribute selection algorithms.

In the test, two different classifiers will be applied which are the support vector machine (SVM) and multi-layer perceptron. The SVM classifier uses the Gaussian Radial Basis Function kernel as it shows the best classification accuracy. Both have been extensively used as classification tools with a great deal of success from object recognition. The attribute selection



algorithms used are Chi-merge, gain ratio and information gain attribute selection algorithms.

Another test is applied on these two data sets used, where the classification test is applied on all the possible combination of attributes. The combination that shows the best classification accuracy is the one generated by the proposed interval-based attribute selection algorithm.

*B. Data sets used in classification*

The selection of the data sets used are based on the need of a data set of continuous attributes and discrete class label. A medical data sets have used which are considered as real life data sets that have no specific distribution of values and may contain misleading values due to an error in calibrations or collection of data. The first data set used is pima-indians-diabetes data set which is obtained from UCI machine learning repository [12]. The percentage of error in this data set will be considered zero. It consists of 536 objects and 8 attributes. 90% of the input data used for training while the rest of 10% is used in testing. The second data set used is a data about HCV therapy where it is classified according to the response of some patients to the interferon therapy whether they cured or not. It consists of 66 objects and 13 attributes. There is a percentage of error that may occur in this data set, where experts indicate that it falls between 2 to 3%. Due to the low number of objects, 70% of the input data only are used for training while the rest of 30% is used in testing. Both data sets are adjusted such that the number objects in every class is equal in both stages of training and testing, the classification accuracy will be measured by dividing the number of correctly classified objects by the total number of objects in the testing data set.

*C. Classification results*

1) *pima-indians-diabetes data set:* When different attribute selection algorithms are used like information Gain (IG), Chi-Merge (CM) and Gain Ratio (GR), these algorithms shows the same order of ranked attributes. This is because these algorithms are all entropy based attribute selection algorithms. The comparison between the order of features ranks of entropy based attribute selection algorithms and the interval-based attribute selection algorithm is shown in table (I). In this table another SVM-based feature selection method (SVMB) [10] is used, where it shows nearly the same results as Information gain algorithm.

TABLE I

THE ORDER OF ATTRIBUTES ACCORDING TO INFORMATION GAIN IG AND INTERVAL-BASED IB FEATURE SELECTION ALGORITHMS

IG	2	8	6	5	1	7	3	4
SVMB	2	6	1	7	8	3	4	5
IB	2	5	7	1	4	6	8	3

Table (II) shows the results when perform classification using Support vector machine (SVM) and Multi-layer perceptron

(MLP). The first row in table (II) shows the classification results when using the input data set contains only attribute 2 in the case of using IG and attribute 2 in the case of using IB. The second row the input data set contains attributes 2, 8 in the case of using IG and attributes 2, 5 in the case of IB. The attributes are incrementally increased based on the attribute selection algorithm used until all attributes are used in the input data set.

In the case using MLP classifier, the peak of accuracy has reached with 81.4% when the input data set contain only the first three selected attributes by the IB algorithm which are 2, 5, 7. While the peak when using the other feature selection algorithms is 75.9% and the number of selected attributes is five attributes which are 2, 8, 6, 5, 1. In the case of using SVM both attribute selection algorithms, the proposed IB and the IG attribute selection algorithms, have the same accuracy percentage peak when the first attribute only is selected. It is noticed that both algorithms have selected the same attribute 2 as the most relevant attribute. On the other hand, All the

TABLE II  
CLASSIFICATION RESULTS OF THE PIMA-INDIANS-DIABETES

SVM	SVM	SVM	MLP	MLP	MLP
IG	SVMB	IB	IG	SVMB	IB
<b>64.81</b>	64.81	<b>64.81</b>	74.07	74.07	74.07
64.81	61.11	61.11	74.07	74.07	75.92
64.81	<b>66.66</b>	62.96	74.07407	70.37	<b>81.48148</b>
53.70	66.66	51.85	72.22222	72.22	79.62963
57.40	68.51	57.40	<b>75.92593</b>	75.92	75.92593
57.47	62.96	57.40	74.07407	<b>77.77</b>	74.07407
55.55	55.55	53.70	72.22222	77.77	68.51852
51.85	51.85	51.85	72.22222	70.37	72.22222

possible combination of attributes are tested in the same way as above using MLP classifier, where 90% of the input data is for training while the rest is for testing. the combination that shows the best results is the set of attributes 2, 5, 7 which is the same set and of the same order generated by the proposed IB algorithm.

2) *HCV data set:* Again in the case Information Gain (IG), Chi-Merge (CM) and Gain Ratio (GR) attribute selection algorithms, The attributes are ranked as follows in table (III). The SVM-Based attribute selection shows the same results as the previous algorithms so it is not useful to maintain them in table (III).

TABLE III

THE ORDER OF ATTRIBUTES ACCORDING TO INFORMATION GAIN IG AND INTERVAL-BASED IB FEATURE SELECTION ALGORITHMS

IG	12	13	4	5	1	3	2	9	11	10
IB	3	4	6	13	9	12	1	5	8	7
IG	6	8	7							
IB	11	10	2							

Table (IV) shows the results when perform classification using SVM and MLP after using both entropy based attribute selection algorithm like the IG and the proposed IB algorithms. It shows that in the case of the selected attributes using the proposed IB algorithm, the classification accuracy has the maximum value when using the first four attributes only. Also the peak was 75 % in the case of using MLP, and 65 % in the case of using SVM. So in both classifiers, the selected attributes by IB has a higher peak than those selected by other attribute selection algorithms.

TABLE IV  
CLASSIFICATION RESULTS OF THE HCV

SVM	SVM	MLP	MLP
IG	IB	IG	IB
25.0	50.0	37.5	50.0
37.5	37.5	37.5	37.5
37.5	37.5	37.5	37.5
37.5	<b>75.0</b>	37.5	<b>62.5</b>
<b>62.5</b>	62.5	37.5	50.0
37.5	37.5	37.5	37.5
50.0	37.5	37.5	37.5
50.0	25.0	37.5	37.5
50.0	62.5	37.5	37.5
50.0	62.5	37.5	50.0
50.0	62.5	37.5	37.5
50.0	50.0	37.5	37.5
50.0	50.0	37.5	37.5

#### IV. CONCLUSIONS

The problem of selecting the features that are relevant to the classifier and remove the irrelevant features has been solved using different attribute selection algorithms. The results demonstrate that the proposed interval-based algorithm encouragingly outperforms most of the popular attribute selection algorithms. The proposed algorithm has two phases for feature selection, the first one is to rank all features according

the discussed algorithm, then select the subset of features of the highest classification accuracy. the proposed algorithm depends on two conditions to determine the selected subset of attributes, where the conditions are based on how high is the classification accuracy and how less is the number of selected attributes. The algorithm has been applied on a real life data sets, where it leads to the best classification accuracy with the least number of features.

#### REFERENCES

- [1] C. Shang and Q. Shen, "Aiding classification of gene expression data with feature selection: a comparative study," *Computational Intelligence Research*, vol. 1, pp 68–76, 2006.
- [2] A. G. K. Janeczek, W. N. Gansterer, M. Demel and G. F. Ecker, "On the relationship between feature selection and classification accuracy," *JMLR: Workshop and Conference Proceedings*, vol. 4, pp. 90–105, 2008.
- [3] Mostafa A. Salama, Aboul Ella Hassanien and Aly A. Fahmy, "Pattern-based Subspace Classification Model," *Second World Congress on Nature and Biologically Inspired Computing (NaBIC)*, Kitakyushu, Japan, pp. 357–362, Dec. 2010.
- [4] M. Prasad, A. Sowmya and I. Koch, "Designing relevant features for continuous data sets using ICA," *Int. J. Comput. Intell. Appl.*, vol. 7, p.447 , 2008.
- [5] Yvan Saeys, Inaki Inza and Pedro Larranaga, "A review of feature selection techniques in bioinformatics," *bioinformatics*, Vol. 23, pp. 2507-2517, 2007.
- [6] Huan Liu and R. Setiono, "Feature selection via discretization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 9, pp. 642–645, Aug. 1997.
- [7] Manuel Mejía-Lavalle, Eduardo F. Morales and Gustavo Arroyo, "Two simple and effective feature selection methods for continuous attributes with discrete multi-class," *Lecture Notes in Computer Science*, vol. 4827, pp. 452–461, 2007.
- [8] W. Duch, T. Wieczorek, J. Biesiada and M. Blachnik, "Comparison of feature ranking methods based on information entropy," *Proceedings of the IEEE International Joint Conference on Neural Networks*, vol. 2, pp. 1415-1419, 2004.
- [9] Yin-Wen Chang and Chih-Jen Lin, "Feature Ranking Using Linear SVM" , *JMLR: Workshop and Conference Proceedings*, pp. 53-64, 2008.
- [10] J. Weston S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio and V. Vapnik, "Feature Selection for SVMs," in *Proc. Neural Information Processing Systems*, pp. 668–674, 2000.
- [11] Chi-Ho Tsang, Sam Kwong, and Hanli Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," vol. 40, Issue 9, pp. 2373–2391, Sep. 2007.
- [12] UCI Machine Learning Repository, <http://archive.ics.uci.edu/ml/datasets.html>.

# Medical Image Segmentation Using Information Extracted from Deformation

Kai Xiao  
 Shanghai Jiao Tong University  
 800 Dong Chuan Road  
 Shanghai, 200240, China  
 Email: showkey@gmail.com

About Ella Hassanien  
 Cairo University  
 5 Ahmed Zewal St.  
 Orman, Giza, Egypt  
 Email: Aboitcairo@gmail.com

Neveen I. Ghali  
 Faculty of Science  
 Al-Azhar University  
 Cairo, Egypt  
 Email: nev\_ghali@yahoo.com

**Abstract**—Deformation of normal structures in medical images has usually been considered as undesired and even a challenging issue to be tackled in medical image segmentation and registration tasks. With the objective of improving brain tumor segmentation accuracy in human brain magnetic resonance (MR) images, this paper proposes an approach to extract useful information from the correlation between lateral ventricular deformation and tumor. In some cases, comparative experiments show the improved tumor segmentation accuracy when the extracted information is added as an additional feature.

## I. INTRODUCTION

**B**RAIN tumor segmentation in magnetic resonance (MR) image is an important image processing step for both medical practitioners and scientific researchers. Large amounts of research efforts have been made in developing effective segmentation methods in the past years, however, such methods have failed to achieve the accuracy level comparable to analyses performed by human experts [1], [2], [3], [4].

One of the most challenging problems that hinder the development of accurate automatic systems is that MR image lacks strong association between actual anatomical meaning and MR imaging (MRI) intensity, especially for pathology such as brain tumor. Therefore, in addition to intensities, other features which are relevant to anatomical meaning are necessary for more accurate tumor segmentation.

This paper brings forward the idea of utilizing brain lateral ventricular deformation, which has been normally considered as undesired and challenging, as an additional information for tumor segmentation, and proposes to quantify the deformation information as a feature with a design and implementation of a feature extraction component. The created feature data is then incorporated for improving tumor segmentation accuracy.

## II. BACKGROUND

### A. Brain Lateral Ventricles and Tumor

A ventricle is an internal cavity of brain. A normal brain contains a connecting system of ventricles, commonly referred to as the ventricular system, which is filled with cerebrospinal fluid (CSF) [5], [6], [7], [8]. Fig. 1 illustrates scans from sequences of T1- and T2-weighted MR images in axial view where lateral ventricles are located in the brain center. Lateral

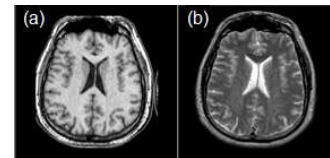


Fig. 1. healthy (a) T1- and (b) T2-weighted brain MR image slices in axial view showing the lateral ventricles, images courtesy of [10].

ventricles are the biggest structures that mainly contain CSF in the axial view of the brain center [9].

As can be seen clearly from Fig. 2(a-i), in cases where brain tumors exist, one or two of the lateral ventricles are compressed; whereas for a healthy brain the two lateral ventricles are nearly symmetrical to each other (see Fig. 1). Fig. 2(j) illustrates the direction of compression from brain tumor and the deformed lateral ventricle. It can be seen that, in some cases, lateral ventricle is compressed in the opposite direction of the location of the brain tumor. This suggests the strong correlation between the two structures. However, due to the fact that brain tumours vary substantially in their location and size, and have diverse effects on lateral ventricles, it is not always the case that brain tumour and deformed lateral ventricles appear in the same image plane. Fig. 3 illustrates how this correlation becomes clear when seen from different planes. The positions of the deformed parts of the lateral ventricle in Fig. 3 (c) and (d) are basically the same as that of the brain tumor in Fig. 3 (b). Fig. 3 (a) illustrates the compression caused by a brain tumour on one lateral ventricle (coronal view). It can be seen that correlations between lateral ventricular deformation and the brain tumor still exist, even though it may not be observable in the same plane or view of MR images.

### B. Brain Tumor Segmentation with Lateral Ventricular Deformation

Automatic MR image segmentation systems are typically designed as a combination of several components of pre-processing, feature extraction, segmentation and classification [1]. Feature extraction component which creates relevant data sets is the key to successful segmentation [1]. It can be assumed that, if the correlation between lateral ventricular

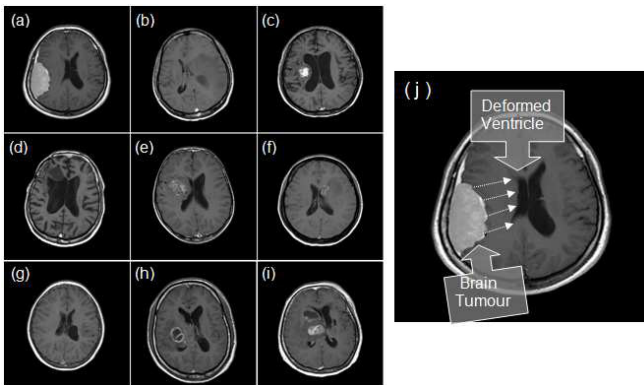


Fig. 2. (a-i): Nine axial view MR image slices from different patients showing brain tumor and lateral ventricles; (j): zoomed illustration of a deformed lateral ventricle and a tumor.

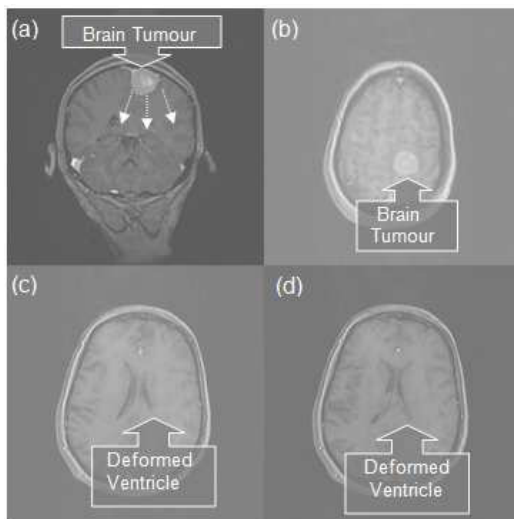


Fig. 3. Some selected MR image slices of one patient showing brain tumour and deformed lateral ventricles not in the same plane of axial view [11]: (a) coronal view in middle of the head; (b) axial view from slice near the top area of the head; (c) axial view from slice in the middle area of the head; (d) axial view from slice next to (c).

deformation and tumor is correctly quantified and used, tumor segmentation accuracy will be accordingly improved.

The fact that lateral ventricles constitute one of the major structures with sharp boundaries in the brain allows for their shapes to be easily delineated from their associated MR images [12], [7], [8]. This makes the feature extraction process based on deformation of this structure relatively more reliable. As a result, lateral ventricular deformation caused by the presence of brain tumors provides an intuition that one could exploit the information derived from the correlations between them. By utilizing this derived information, brain tumor segmentation could be improved.

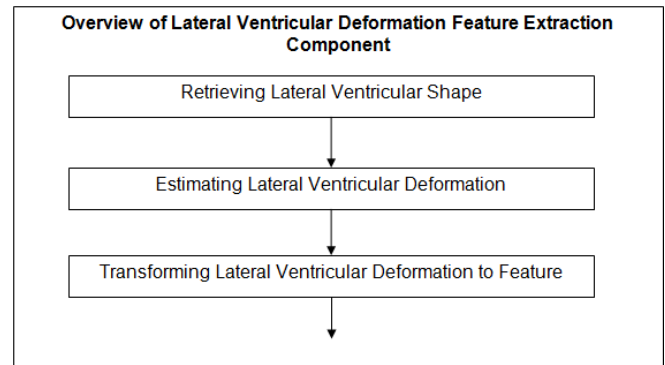


Fig. 4. Overview of the lateral ventricular deformation feature extraction component.

### III. DEFORMATION INFORMATION EXTRACTION

#### A. Overview

The task of extracting the lateral ventricular deformation information should include retrieval of lateral ventricular shape, and transformation of lateral ventricular deformation into feature. To achieve the latter, one process to model and estimate the deformation of the retrieved lateral ventricular shape should be applied.

The step of modeling and estimating lateral ventricular deformation is to associate template and deformed lateral ventricles, and to model, calculate and quantify shape variation. Hence, both healthy and deformed lateral ventricles must be available for shape comparison. However, lateral ventricles of one person are subject to shape variation with age, even with the absence of pathology or abnormality [13]. Furthermore, because of the varieties of size, location and type of brain tumors [14], [15], [16], their compression effects on lateral ventricles are significantly diverse. There are no lateral ventricles that can be used as a general template to perfectly associate the deformed lateral ventricles in all cases. Therefore, an additional step of adjusting feature data is necessary in the lateral ventricular deformation feature extraction component. The design of lateral ventricular deformation feature extraction component actually consists of the three streamlined processes as illustrated in Fig. 4.

#### B. Retrieving Lateral Ventricular Shape

The existing works on brain lateral ventricular shape retrieval generally apply the approach of discriminating tissues of ventricles through the analysis on its intensity and geometric location [17], [18]. In this paper, the lateral ventricular shape retrieval is achieved through three steps: brain MR image tissue segmentation for separating CSF tissue from other tissues, CSF identification for locating the image pixels/regions of which lateral ventricles are composed and lateral ventricles extraction for removing CSF pixels outside the lateral ventricular regions.

In order to separate CSF from other tissues, we can create several clusters by applying a clustering method. In this step, brain tissues are clustered according to the similarity of

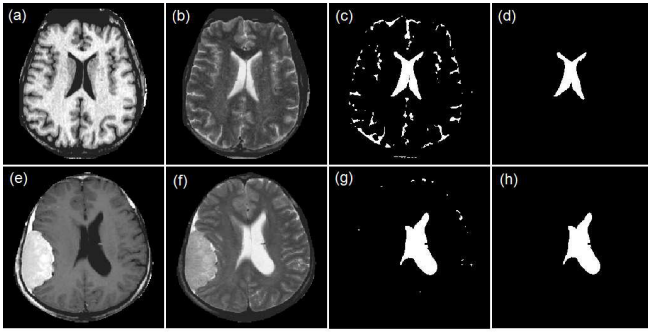


Fig. 5. Lateral ventricular shape retrieval results: (a) healthy T1-weighted MR image data; (b) T2-weighted MR image data of (a); (c) after tissue segmentation on (a) and (b); (d) extracted lateral ventricles from (c); (e) tumor-affected T1-weighted MR image data; (f) T2-weighted MR image data of (e); (g) after tissue segmentation on (e) and (f); (h) extracted lateral ventricles from (g).

MRI intensity of tissue types, and one of the output clusters accommodates CSF tissue. This research applies our previous work of feature-weighted Fuzzy C-Means (fwFCM) [19], [20] which provides higher insensitivity to noise and capability of adjusting feature weights. These properties make fwFCM more suitable for MR images than conventional FCM algorithm.

After brain tissue segmentation, CSF extraction can be conducted by selecting the cluster labeled as CSF from the multiple clusters created by the fwFCM method. It can be seen from Fig. 1 that, under the situation of absence of certain types of lesions, within several major tissue types, CSF is the only one that appears bright in T1-weighted and dark in T2-weighted MR images [6]. As a result, intensity values of CSF in T2-weighted MRI are high while those in T1-weighted MRI are low. This selection process can be expressed as:  $\max_i \{V_{T2,i} - V_{T1-i}\}$ , where  $i$  is the cluster number,  $V_{T2}$  and  $V_{T1}$  represent the centroid values in the T2- and T1-weighted features respectively in the input feature set of the fwFCM clustering.

Although from brain MR images of axial view, lateral ventricles are of large volume, CSF can still fall outside the ventricular system [6]. Furthermore, the proposed brain tissue segmentation step may wrongly assign non-CSF pixels to the cluster of CSF. To remove undesired pixels, a global mask is applied to remove pixels outside the area where pixels of lateral ventricles normally reside, thereby leaves the regions as the extracted lateral ventricles. Fig. 5 demonstrates the lateral ventricles extraction results by applying this approach.

### C. Estimating Lateral Ventricular Deformation

The process of estimating lateral ventricular deformation can be decomposed into two steps: lateral ventricles alignment and lateral ventricular deformation measurement. Lateral ventricles alignment is achieved by two sub-steps of linking lateral ventricles and selecting landmark points. The lateral ventricular deformation measurement step is also achieved by two sub-steps of modeling lateral ventricular deformation and calculating estimated deformation.

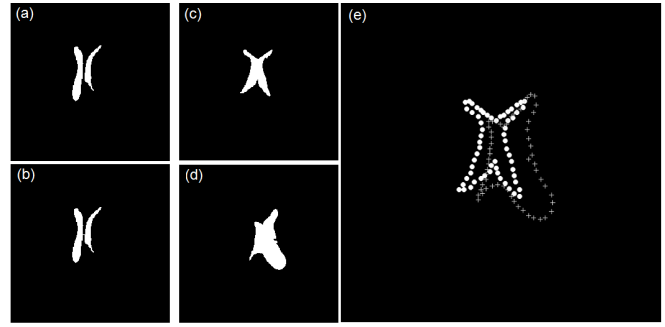


Fig. 6. Lateral ventricles linking and alignment results: (a) separated left and right lateral ventricles; (b) linked left and right lateral ventricles; (c) template; (d) target; (e) aligned template and target. White circles and grey cross represents control points on template and target lateral ventricles, respectively.

In MR images, left and right lateral ventricles are actually separated [7], [8]. Furthermore, lateral ventricles may be broken. To effectively estimate deformation using modeling functions, left and right lateral ventricles are treated as one single object. Therefore left and right ventricles and the disjointed parts of lateral ventricles need to be linked together. Fig. 6(a, b) demonstrate a sample resultant image after the separated lateral ventricles are linked together through a connecting line with the shortest distance.

Based on the study of anatomical properties of brain lateral ventricles, anterior and posterior horns are employed as key landmark points. The process of lateral ventricles alignment is completed by selecting intermediate landmark points based on these key landmark points. Fig. 6(c, d) show the extracted template and target lateral ventricles, respectively. Fig. 6(e) displays the alignment results by selecting the landmark points and overlapping them in one image. However, misalignment effect caused by the applied imperfect template can be easily observed.

Deformation is usually modeled and represented as a transformation function [21]. Generally, in order to model deformation, both linear and non-linear functions can be used. However, with regards to brain MR images, linear transformation functions, cause the images to be globally smoothed, thereby accommodating only very small and simple deformations [22] and making the process of employing them for modeling deformation undesirable. As a nonlinear deformation modeling function, thin plate splines (TPS) function is employed to perform the nonlinear mapping between template and target lateral ventricular boundary image data set [23]. A TPS  $f(x, y)$  is a smoothing function which interpolates a surface that is fixed at landmark points  $P_i$  at a specific height. TPS can be treated as a process of finding a function  $z(x, y)$  which minimizes the bending energy [23], [24]. In the application for 2-dimensional images, instead of assuming that  $f$  corresponds to a displacement orthogonal to the image plane at the landmark points, one can treat it as a displacement in the image plane [24]. By using two separate TPS functions  $f_x$  and  $f_y$  which model the displacement of the selected landmark points in the  $x$  and  $y$  direction, a vector-valued function  $F$  which maps each



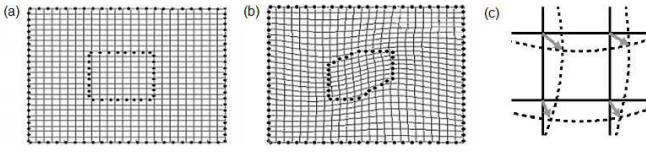


Fig. 7. Example of landmark points with original and deformed TPS meshes formed by  $x$  and  $y$  coordinates: (a) original; (b) deformed. (c) Illustration of the effect of deformation in a zoomed view before and after deformation, solid and dashed lines are segments of the meshes from the original and deformed image, respectively; grey arrows indicate the vector of the displacement due to deformation.

point of the image into a new point in the image plane can be represented using  $(x', y') \rightarrow (f_x(x, y), f_y(x, y))$ , where  $f_x$  and  $f_y$  are the functions causing displacement on  $x$  and  $y$  coordinates respectively.

Once the vector-valued function  $F$  is defined via the selected landmark points with TPS functions, it is then applied to the original coordinates of all pixels in the target image to retrieve new coordinates for all pixels. If one treats some selected horizontal and vertical lines of equal spaces in the original image as a mesh, then a corresponding distorted mesh can be used to describe the displacement of each node of the mesh. This effect is illustrated in Fig. 7(a, b) by visualizing the original and its corresponding deformed meshes after TPS function is applied to all pixels in the image. The effect of the deformation on the lateral ventricles can then be represented by finding out the coordinate displacement value of each pixel in the image. In the deformed image, each pixel is displaced from its original coordinate at specific direction and distance. Therefore, vector can be used for representing the estimated deformation of each point. Fig. 7(c) illustrates an example on a segment of an image using magnitude and direction of vectors to represent the deformation measurement.

#### D. Transforming Lateral Ventricular Deformation to Features

The process of transforming lateral ventricular deformation to features can be decomposed to two indispensable steps: estimated deformation data to feature conversion and lateral ventricular deformation data adjustment.

The estimated deformation data is normalized to the format of image grayscale value of 8 bits, which can be represented as Equation  $I_k = \frac{D_k}{\max\{D\}} \times 255$ , where  $I$  is the normalized intensity value,  $k$  is the index of pixel in the image and 255 is the maximum 8 bits grayscale value of MR image pixels, and  $D$  is the magnitude of displacement vector which can be calculated by using the Euclidean distance as  $D = \sqrt{(P_{o,x} - P_{t,x})^2 + (P_{o,y} - P_{t,y})^2}$ , where  $P_{o,x}$  and  $P_{o,y}$  are the original data point  $x$  and  $y$  coordinate values, while  $P_{t,x}$  and  $P_{t,y}$  are the obtained data point  $x$  and  $y$  coordinate values, respectively.

It can be seen in Fig. 8(b) that that the maximum measured deformation value is not in the area where tumor resides, and the direction of vector denoting the highest displacement value is irrelevant to that of the compression from the brain tumor. This is mainly because of the misalignment between

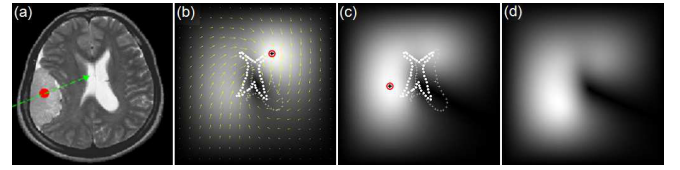


Fig. 8. Example of estimating lateral ventricular deformation: (a) manual selection of tumor, where round dot is selected by user input; (b) visualized deformation estimation, where magnitude of displacement vector is normalized and visualized as image grayscale value, arrows are showing the directions of displacement vectors, and a cross covered by a circle indicates the maximum displacement vector magnitude; (c) visualized deformation estimation; (d) deformation feature extraction final result.

the template and target lateral ventricles caused by using the imperfect lateral ventricles template (as seen in Fig. 6(e)). To address this problem, as shown in Fig. 8(a), a method for adjusting the estimated deformation data is proposed. The method allows user to select one point in the brain tumor. With this reference point, a line can be drawn from the point to the center of the image, at the direction from the border to the center of the image. Deformation estimation values can therefore be adjusted by  $D' = D |\cos(\frac{\theta}{2})|$ , where  $D$  and  $D'$  are the magnitudes of the original and adjusted displacement vector representing the estimated deformation, respectively.  $\theta$  is the angle between the connecting line and the displacement vector. It can be seen that when  $\theta = \pi$  (the connecting line and the displacement vector are in the opposite directions), adjusted displacement vector magnitude is reduced to 0. And when  $\theta = 0$ , magnitude of the adjusted displacement vector is 1, which means the estimated data is kept without any change.

Fig. 8(c) visualizes the estimated deformation data by normalizing it into grayscale values and Fig. 8(d) illustrates the final feature data of lateral ventricular deformation. It can be seen that, although not perfect, the bright area which indicates the high deformation values is approximately in the same location as brain tumor in the original image. The deformation estimation values can then be used as an additional feature in the feature set the brain MR image tumor segmentation methods that support input data of multiple features.

## IV. BRAIN TUMOR SEGMENTATION WITH LATERAL VENTRICULAR DEFORMATION FEATURE

### A. System Implementation

Structure of the brain tumor segmentation system in this research follows the common MR image segmentation system [1] shown in Fig. 9(a). To investigate the effect of brain tumor segmentation caused by the feature of lateral ventricular deformation, special considerations have to be given to the inclusive components of pre-processing, feature extraction and brain tumor segmentation.

The pre-processing component in this brain tumor segmentation system will need to address the issues of intensity non-standardization, geometrical non-uniformity and redundant data in the image background and skull. These issues are respectively addressed by the four streamlined processes of

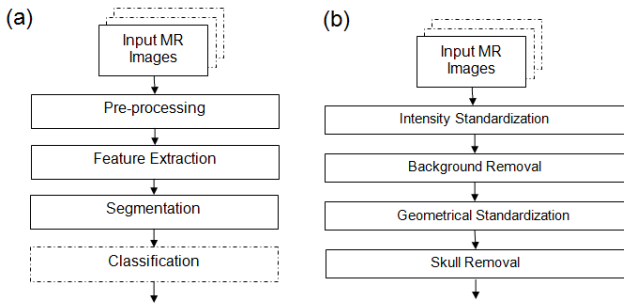


Fig. 9. System structures: (a) structure of a common MR image segmentation system.; (b) overview of the pre-processing component.

intensity standardization, geometrical standardization, background and skull removal processes, as illustrated in Fig. 9(b).

In the segmentation component, selected supervised and unsupervised segmentation methods are used to evaluate the effect of lateral ventricular deformation feature on brain tumor segmentation. In order to achieve that, this research uses two feature sets, one includes the extracted lateral ventricular deformation feature, and the other does not. By comparing the segmentation results using the same segmentation method, i.e., supervised or unsupervised, effectiveness of the feature of brain lateral ventricular deformation can be examined. In this paper, the most frequently used  $k$ -nearest neighbors ( $k$ -NN) [25], [26], [1] and conventional FCM [27], [19], [20] are selected as supervised and unsupervised methods, respectively. The intention of selecting conventional FCM method is to evaluate the segmentation results by the original segmentation method with no interference or adjustment.

### B. Experimentation and Evaluation

To use the supervised  $k$ -NN algorithm on brain tumor segmentation, tumor segmentation results from medical experts are used as training samples in which tumor areas are marked. Therefore all pixels in the images can be labeled as tumor or non-tumor. After the  $k$ -NN classification, each testing image pixel is categorized as tumor or non-tumor. In the experiments using unsupervised FCM algorithm, the number of clusters is set to 6 denoting six clusters of major brain tissues. The cluster of tumor will be identified manually out from 6 clusters after the clustering process due to the fact that there is no training data for the FCM method.

By using input feature set with or without the extracted deformation feature, pixels segmented as tumor which are in the same class as the corresponding pixels in the segmentation by medical expert are treated as correctly segmented tumor pixels; those segmented as tumor but labeled as non-tumor in the segmentation by medical expert are treated as wrongly segmented tumor pixels.

Statistical measures of sensitivity and specificity [28], [29] are applied for evaluating the segmentation results. By treating correctly segmented tumor, wrongly segmented tumor, correctly segmented non-tumor and wrongly segmented non-tumor pixel number as true positive ( $true^+$ ), false positive

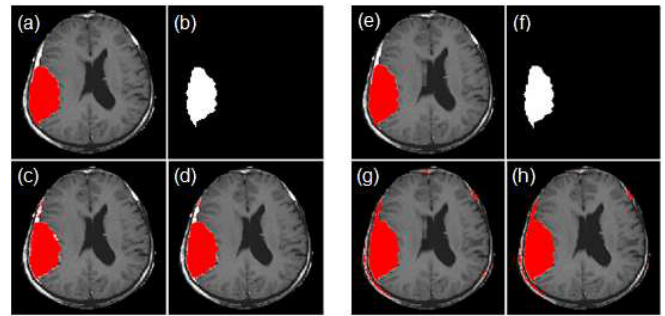


Fig. 10. Training data and tumor segmentation results using  $k$ -NN classifier: (a) segmentation by medical expert; (b) training data converted from (a); (c) segmentation result without deformation feature; (d) segmentation result with deformation feature, and segmentation results using FCM clustering method: (e) segmentation by medical expert; (f) visualized segmentation by medical expert; (g) segmentation result without deformation feature; (h) segmentation result with deformation feature.

( $false^+$ ), true negative ( $true^-$ ) and false negative ( $false^-$ ) number respectively, sensitivity and specificity values can be obtained according to:  $Sensitivity = \frac{true^+}{true^+ + false^-}$  and  $Specificity = \frac{true^-}{true^- + false^+}$ , respectively. A sensitivity of 100% means that the test recognizes all actual positives, i.e., all brain tumor pixels are segmented as tumor. And a specificity of 100% means that the test recognizes all actual negatives, i.e., all non-tumor pixels are segmented as non-tumor [28].

The brain tumor segmentation result of one image case is illustrated in Fig. 10. It can be seen from Fig. 10 (d, h) that, the classification results obtained from the feature set with lateral ventricular deformation feature are respectively closer to the results from medical expert than the result shown in Fig. 10 (c, g), which is created from the feature set without lateral ventricular deformation feature.

With more image cases, sensitivity and specificity values tabulated in Table I and Table II provide further evidence of the positive effect from the additional feature. It can be seen from in Table I that from the experiments using supervised  $k$ NN, with the inclusion of the extracted lateral ventricular deformation feature, specificity values in the increase for all eight cases. Except for the decrease in case number 3, sensitivity values increase in 7 out of 8 cases. In the experiments using unsupervised FCM as illustrated in Table II, with the inclusion of the extracted lateral ventricular deformation feature, specificity values increase for all eight cases. However, sensitivity values decrease in 3 out of 8 cases. This is mainly because there is no specific rule created for distinguishing between tumor and non-tumor pixels in clustering [30]. In short, the clusters of tumor or non-tumor are not well defined concepts if no training process is included in the segmentation.

### V. CONCLUSION

This paper establishes and implements the idea of utilizing deformation of lateral ventricles for brain tumor segmentation. The results show that the extracted lateral ventricular deformation information is relevant to the position of brain tumor.

TABLE I  
SPECIFICITY AND SENSITIVITY VALUES OF BRAIN TUMOR SEGMENTATION USING SUPERISED  $k$ NN METHOD

	Specificity		Sensitivity	
	Feature Set			
	without deformation	with deformation	without deformation	with deformation
Case 1	99.9%	100.0%	94.3%	95.3%
Case 2	99.9%	100.0%	90.9%	98.2%
Case 3	99.8%	99.9%	95.3%	94.3%
Case 4	99.1%	99.4%	79.7%	88.2%
Case 5	99.2%	99.7%	61.6%	82.7%
Case 6	99.8%	99.9%	93.7%	96.0%
Case 7	99.2%	99.8%	87.3%	96.6%
Case 8	99.8%	99.9%	91.0%	95.3%

TABLE II  
SPECIFICITY AND SENSITIVITY VALUES OF BRAIN TUMOR SEGMENTATION USING UNSUPERISED FCM METHOD

	Specificity		Sensitivity	
	Feature Set			
	without deformation	with deformation	without deformation	with deformation
Case 1	100.0%	100.0%	80.6%	81.9%
Case 2	99.2%	99.9%	2.29%	12.47%
Case 3	99.9%	99.9%	22.4%	20.2%
Case 4	98.2%	100.0%	13.8%	18.4%
Case 5	99.9%	99.9%	13.8%	15.7%
Case 6	99.2%	99.5%	11.1%	11.0%
Case 7	97.8%	99.2%	80.3%	84.1%
Case 8	100.0%	100.0%	49.1%	36.5%

By incorporating the relevant lateral ventricular deformation as an additional feature in the feature set for pattern recognition segmentation methods, brain tumor segmentation accuracy increases.

## REFERENCES

- [1] Clarke, L.P., Velthuizen, R.P., Camacho, M.A., Heine, J.J., Vaidyanathan, M., Hall, L.O., Thatcher, R.W., Silbiger, M.L.: MRI Segmentation: Methods and Applications. *Neuroanatomy*. 11(3), 343–368 (1995)
- [2] Clark, M., Hall, L., Goldgof, D., Velthuizen, R., Murtagh, F., Silbiger, M.: Automatic Tumor Segmentation Using Knowledge-Based Techniques. *IEEE Trans. Med. Imaging*. 17, 238–251 (1998)
- [3] Ray, N., Greineer, R., Murtha, A.: Using Symmetry to Detect Abnormalities in Brain MRI. *Computer Society of India Communications*. 31(19), 7–10 (2008)
- [4] Saha, B. N., Rayl, N., Greiner, R., Murtha, R., Murtagh, A., Zhang, H.: Quick Detection of Brain Tumors and Edemas: A Bounding Box Method Using Symmetry. *Computerized Medical Imaging and Graphics*. doi:10.1016/j.compmedimag.2011.06.001 (2011)
- [5] Bear, M.F., Connors, B.W., Paradiso, M.A.: *Neuroscience: Exploring the Brain*. Lippincott Williams Wilkins (1996)
- [6] Gunderman, R. B.: *Essential Radiology: Clinical Presentation, Pathophysiology, Imaging*. Thieme Medical Publishers (1998)
- [7] Goetz, C.G., Pappert, E.J.: *Textbook of Clinical Neurology*. W. B. Saunders Company (1999)
- [8] Fix, J.D.: *Neuroanatomy*. Lippincott Williams Wilkins (2001)
- [9] Brown, M., Semeka, R.: *MRI: Basic Principles and Applications*, 3rd Edition. John Wiley and Sons, Inc. (2003)
- [10] The Whole Brain Atlas, <http://www.med.harvard.edu/AANLIB>
- [11] Internet Brain Segmentation Repository provided by MGH CMA. <http://www.cma.mgh.harvard.edu/ibsr>
- [12] Gaser, C., Nenadic, I., Buchsbaum, B.R., Hazlett, E.A., Buchsbaum, M.S.: Deformation-Based Morphometry and Its Relation to Conventional Volumetry of Brain Lateral Ventricles in MRI. *Neuroanatomy*. 13(6), 1140–1145 (2001)
- [13] Chung, S.C., Tack, G.R., Yi, J.H., Lee, B., Choi, M.H., Lee, B.Y., Lee, S.Y.: Effects of Gender, Age, and Body Parameters on the Ventricular Volume of Korean People. *Neurosci. Lett*. 395, 155–158 (2006)
- [14] Prastawa, M., Bullitt, E., Ho, S., Gerig, G.: A Brain Tumor Segmentation Framework Based on Outlier Detection. *Med. Image Anal.* 395, 155–158 (2004)
- [15] Prastawa, M., Bullitt, E., Gerig, G.: Synthetic Ground Truth for Validation of Brain Tumor MRI Segmentation. In: *Proceedings of Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pp.26–33. LNCS, 3749 (2005)
- [16] Louis, D. N., Ohgaki, H., Wiestler, O.D., Cavenee, W.K., Burger, P.C., Jouvet, A., Scheithauer, B.W., Kleihues, P.: The 2007 WHO Classification of Tumors of the Central Nervous System. 114, 97–109 (2007)
- [17] Worth, A.J., Makris, N., Patti, M.R., Goodman, J.M., Hoge, E.A., Caviness, V.S., Kennedy, D.N.: Precise Segmentation of the Lateral Ventricles and Caudate Nucleus in MR Brain Images Using Anatomically Driven Histograms. *IEEE Trans. Med. Imaging*. 17(2):303–310 (1998)
- [18] Wu, Y., Phol, K. Warfield, S.K., Cuttmann, C.R.G.: Automated Segmentation of Cerebral Ventricular Compartments. In: *International Society for Magnetic Resonance in Medicine Eleventh Scientific Meeting and Exhibition*. ISMRM (2003)
- [19] Xiao, K., Ho, S.H., Hassani, A.E.: Automatic Unsupervised Segmentation Methods for MRI Based on Modified Fuzzy C-Means. *Fundamenta Informaticae*. 87(3-4):465–481 (2008)
- [20] Xiao, K., Ho, S.H., Bargiela, A.: Automatic Brain MRI Segmentation Scheme Based on Feature Weighting Factors Selection on Fuzzy C-Means Clustering Algorithms with Gaussian Smoothing. *Int. J. of Comput. Intell. Bioinfo. Sys. Bio*. 1(3):316–331 (2010)
- [21] Zelditch, M., Swiderski, D., Sheets, D.H., Fink, W.: *Geometric Morphometrics for Biologists*. Academic Press (2004)
- [22] Tittgemeyer, M., Wollny, G., Kruggel F.: Visualising Deformation Fields Computed by Non-linear Image Registration. *Comput. Vis. Sci*. 5(1), 45–51 (2002)
- [23] Bookstein, F.L.: Principal Warps: Thin-Plate Splines and the Decomposition of Deformation. *IEEE Trans. Pattern Anal. Mach. Intell.* 11, 567–585 (1989)
- [24] Hajnal, J.V., Hawkes, D.J., Hill, D.G.: *Medical Image Registration*. CRC Press (2001)
- [25] Cover, T.M., Hart, P.E.: Nearest Neighbor Pattern Classification. *IEEE Trans. Inf. Theory*. 13(1), 21–27 (1967)
- [26] Bezdek, J.C., Hall, L.O., Clarke, L.P.: Review of MR Image Segmentation Using Pattern Recognition. *Med. Phys.* 20(4), 1033–1048 (1993)



- [27] Bezdek, J.C.: A Convergence Theorem for the Fuzzy Isodata Clustering Algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* 2(1), 1–8 (1980)
- [28] Altman, D.G., Bland, J.M.: Statistics Notes: Diagnostic Tests 1: Sensitivity and Specificity. *BMJ.* 308, 1552 (1994)
- [29] Hayes, W.L.: *The Grid: Statistics for the Social Sciences*. Holt, Rhinehart and Winston, New York (1973)
- [30] Everitt, B.S.: Cluster Analysis: a Brief Discussion of Some of the Problems. *The Br. J. Psych.* 147, 143–145 (1972)



## Discovering similarities for the treatments of liver specific parasites

Pınar Yıldırım

Okan University, Faculty of  
Engineering and Architecture,  
Department of Computer  
Engineering, Tuzla Campus,  
34959, Akfırat, Tuzla, Istanbul,  
Turkey

Email: pınar.yildirim@okan.edu.tr

Kagan Ceken

Akdeniz University, Department of  
Radiology, Dumlupınar Bulvarı,  
Antalya, Turkey

Email: kceken@akdeniz.edu.tr

Osman Saka

Akdeniz University, Department of  
Biostatistics and Medical  
Informatics, Dumlupınar Bulvarı,  
Antalya, Turkey

E-mail: saka@akdeniz.edu.tr

**Abstract**—Medline articles are rich resources for discovering hidden knowledge for the treatments of liver specific parasites. Knowledge acquisition from these articles requires complex processes depending on biomedical text mining techniques. In this study, name entity recognition and hierarchical clustering techniques were used for advanced drug analyses. Drugs were extracted from the articles belonging to specific time periods and hierarchical clustering was applied on parasite and drug datasets. Hierarchical clustering results revealed that some parasites have similar in terms of treatment and the others are different. Our results also showed that, there have not been major changes in the treatment of liver specific parasites for the past four decades and there are problems associated with the development of new drugs. Both pharmaceutical initiatives and health-care providers should investigate major drawbacks and develop some strategies to overcome these problems.

**Index Terms**—Biomedical Text Mining, Clustering Analysis, Liver, Parasite.

### I. INTRODUCTION

MEDLINE articles are rich resources for discovering and tracking medical knowledge. Biomedical text mining techniques play an important role to acquire knowledge from these articles and they are applied for numerous studies in biomedical domain so far.

Parasitic diseases affect hundreds of millions of people worldwide and result in significant mortality and devastating social and economic consequences [1]. Parasitic diseases are especially harmful on the liver which supports almost every organ and liver specific parasites also affect many people [2].

In this study, we developed a knowledge discovery on the treatment of parasites affecting liver and we hope that our study makes substantial contributions to all scientists and medical experts working on parasites affecting the efficiency of the liver's mechanism.

### II. METHODS

We used the Medline distribution available through the PubMed Web portal at the National Library of Medicine (NLM)<sup>1</sup> as well as on the in house distribution at the EMBL-

EBI<sup>2</sup>. In the first phase of our article analysis procedure, two clinicians proposed a list of species which induce liver-specific diseases. They also proposed classes of drugs that could be used in the treatment of these diseases.

Medline is a collection of biomedical documents and administered by the National Center for Biotechnology Information (NCBI) of the United States National Library of Medicine (NLM). PubMed web site provides a service of the National Library of Medicine that include over 20 millions bibliographic citations from Medline and other life science journals for biomedical articles back to 1950s. The full text of articles are not stored; rather, links to the provider's site to obtain the full-text articles are given, is available [3-4].

TABLE I  
NUMBER OF ARTICLES FOR SELECTED PARASITES

Parasite	Number of Articles
Clonorchis Sinensis	178
Echinococcus Multilocularis	229
Echinococcus granulosus	400
Entamoeba histolytica	1075
Fasciola Hepatica	917
Schistosoma Japonicum	446
Schistosoma Mansoni	1731
Opisthorchis Viverrini	213
<b>Total</b>	<b>5189</b>

Medline abstracts are in XML format and they contain logical markup to organize meta information such as the journal, author list, affiliations, publication dates and related MeSH headings [5].

We used a complex query for the retrieval of the Medline abstracts that are relevant for the liver specific parasites. The query resulted in a document set of 17,377 articles and all articles were processed with the text mining solution available at

<sup>1</sup> <http://www.ncbi.nlm.nih.gov/pubmed/>

<sup>2</sup> <http://www.ebi.ac.uk/citexplore/>

the EBI (European Bioinformatics Institute) called “Filter Server”. In EBI architecture, a filter server specializes in recognizing the vocabulary of a particular terminology and receives a stream of text and annotates it with XML tags [6].

In our study, the filter servers identified the species mentions and its variants and the mention of drugs in the text. Species in the articles were annotated and parasites’ names affecting liver were selected by two medical doctors. The frequencies of parasites were calculated. The frequency of the parasite provides the number of times a considered parasite appeared in the selected articles. Most ranked eight of them were used for analysis. Relevant articles for each parasite belonging to specific time periods (e.g., 1970-1980, 1980-1990, 1990-2000 and 2000-2009) were collected from PubMed. Table 1 shows the number of articles for selected parasites.

After retrieving the articles in specific time periods, drugs names were found by using drug filter server which tags drugs’ names from DrugBank. The DrugBank database is a bioinformatics and cheminformatics resource that combines detailed drug data with comprehensive drug information<sup>3</sup> [7].

Drugs have some variations such as synonyms and brand names. DrugBank was searched for each drug, synonyms and brand names of drugs were found. After finding variations, these names were mapped to one specific name.

#### A. Clustering Analysis

Clustering analysis is one area of machine learning of particular interest to data mining. It provides the means for the organization of a collection of patterns into clusters based on the similarity between these patterns, where each pattern is represented as a vector in multidimensional space [8].

Hierarchical clustering methods produce a hierarchy of clusters from small clusters of very similar items to large clusters that include more dissimilar items. Hierarchical methods usually produce a graphical output known as a dendrogram or tree that shows this hierarchical cluster structure. Some hierarchical methods are divisive; those progressively divide the one large cluster comprising all of the data into smaller clusters and repeat this process until all clusters have been divided. Other hierarchical methods are agglomerative and work in the opposite direction by first finding the clusters of the most similar items and progressively adding less similar items until all items have been included into a single large cluster [8].

The Euclidean distance is one of the common similarity measures and it is defined as the square root of the squared discrepancies between two entities summed over all variables measured [9].

The pseudo code of agglomerative hierarchical clustering algorithm is as follows:

1. Compute the proximity matrix
2. Merge the closest two clusters
3. Update the proximity matrix to reflect the proximity between the new cluster and the original clusters
4. Repeat step 2 and 3 until only one cluster remains

Figure 1 shows an example of dendrogram for A,B,C,D and E objects [10].

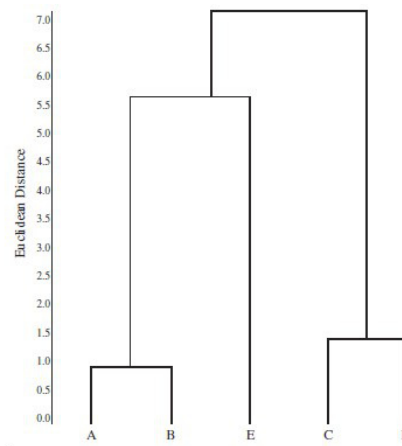


Fig. 1 An Example of Dendrogram for A,B,C,D and E objects

### III. RESULTS

Hierarchical clustering analysis was used to post-process the results from the co-occurrence analysis for the treatment of parasites. R statistical software is used for clustering and heatmap analysis. A heatmap is a graphical way of displaying a table of numbers by using colors to represent the numeric values.

The main categories of drugs that are used for the treatment of parasites comprise anthelmintic, anti-inflammatory and antiprotozoal drugs. Figures 2, 3, 4 and 5 show the heatmaps of the cluster analysis.

According to the observed results, one cluster consists of *Echinococcus Multilocularis*, *Echinococcus Granulosus* and *Fasciola Hepatica*. They share the following commonality:

- Albendazole, mebendazole and praziquantel from the standard treatment for all three parasites. This raises the notion that drugs developed for the treatment of one specie could in principle be exploited for the other two species.

*Clonorchis Sinensis*, *Schistosoma Mansoni* and *Opisthorchis Viverrini* form the second cluster of the analysis. In this cluster, praziquantel is seen as the common treatment. Apart from this drug, these parasites show little treatment with the other antihelmintic drugs. It is possible that these species would still profit from treatment with any of the other drugs.

*Schistosoma Mansoni* forms its own cluster. In the cluster, patients undergo treatment with antiinflammatory drugs similar to patients suffering from *Fasciola Hepatica*, but the type of antiinflammatory treatment differs significantly from the treatment of *Fasciola Hepatica*. Furthermore, patients suffering from *Schistosoma Mansoni* receive additional novel anthelmintic drugs such as levamisole and oxamniquine from 1980 to 2000.

Altogether, the treatment of parasites seems to be fairly stable over the past four decades with regards to the reporting of treatments in the scientific literature.

<sup>3</sup> <http://www.drugbank.ca/>

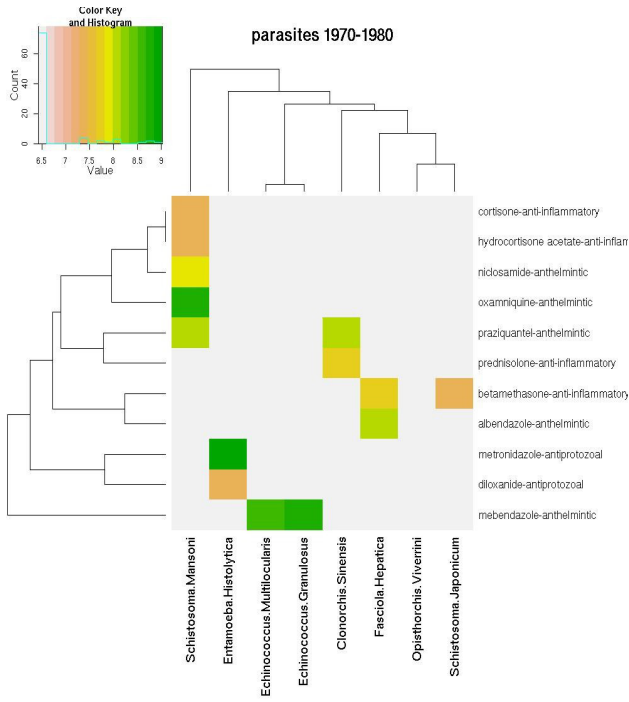


Fig. 2 Drug heatmap of parasites for 1970-1980 time period

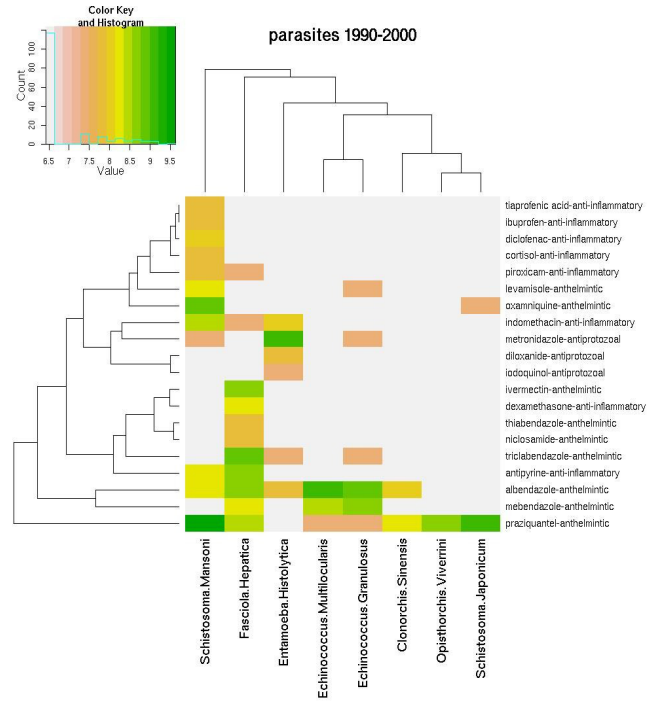


Fig. 4 Drug heatmap of parasites for 1990-2000 time period

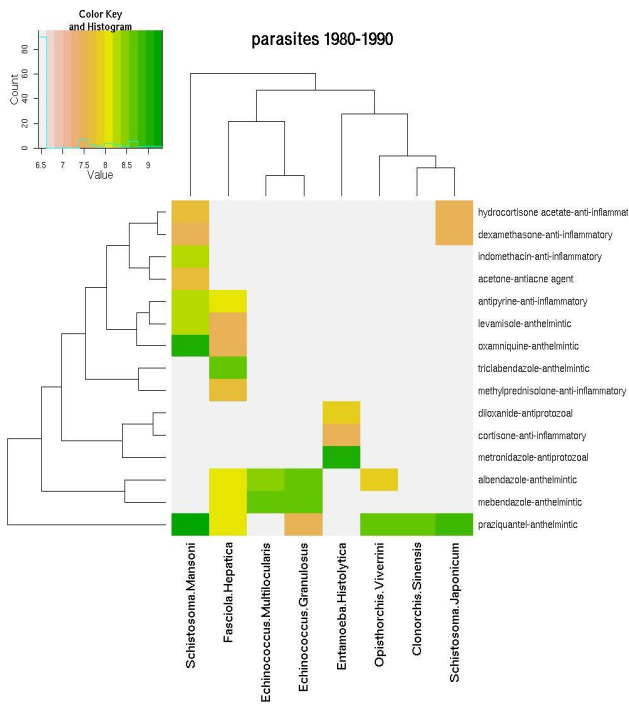


Fig. 3 Drug heatmap of parasites for 1980-1990 time period

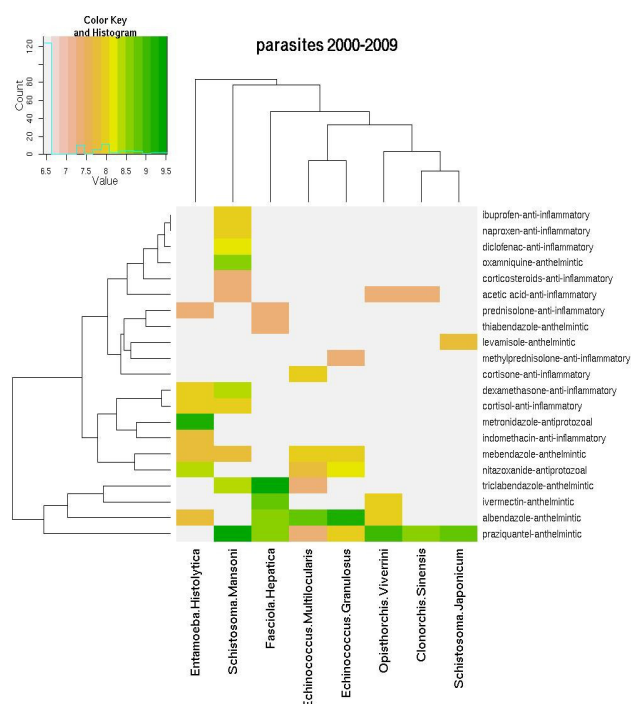


Fig. 5 Drug heatmap of parasites for 2000-2009 time period

#### IV. DISCUSSION

Infections with parasites are important causes of morbidity and mortality [11]. The control of parasitic disease requires a complex interplay of activities in the fields of public health, education, political will and medicine science. There is a need for treatment, and the search for better drugs is a perpetual process. Advances in science, especially in the field of parasite genomics and its attended technology, have opened up possibilities for new drugs.

We analyzed Medline abstracts to extract hidden knowledge and according to our results, there are no big changes among time periods in the treatment of liver specific parasites. Despite the large global burden of parasitic diseases, there has been very little recent effort by the pharmaceutical industry to develop agents to treat human parasitic infections [12].

Parasitic diseases, though globally massive in their impact, mainly affect poor people in poor regions of the world. As such, they would never be viewed as viable target markets for the pharmaceutical industry, particularly in today's post merger climate. In parallel, funding for basic research on these organisms and the pathogenesis of the diseases they produce has been woefully inadequate compared with funding for diseases of much lower prevalence but more direct impact in the developed countries of Europe and North America [1].

The protection of proprietary rights and the return of investments are also important issues for drug makers. With the long payback period associated with these indications, costs often are not recovered when a compound runs off patent and generic products may be introduced [13].

Regulatory requirements are another major concern that has a considerable impact on the length and costs of the drug development process and, hence, on the ultimate market price of the drug product. Paradoxically, increasingly demanding standards favour the larger wealthy companies which are those least interested in tropical diseases. Nevertheless, dossiers do not always undergo the same level of review worldwide, sometimes because of limited health budgets, and sometimes owing to a misconception about the regulatory process [13].

Medline abstracts are generally publicly available and therefore easy to share and distribute, while full text papers are not always available. However, there are some works to make them easily available and they can be public and sharable soon. As future work, it would be of interest to develop an efficient way to analyze full text papers to compare the results of abstracts [14].

#### V. CONCLUSIONS

Biomedical literature provides valuable knowledge for clinical studies and research. Medical experts can not read all the articles in a specific medical problem and discover hidden connections between entities. In this study, we worked with medical doctors and considered their needs and

the point of view of them. Liver specific parasites were selected for the research. The combination of data mining and text mining techniques were used to get some facts hidden in Medline articles. Drugs which are based on specific time periods were extracted from the articles by using named entity recognition techniques and hierarchical clustering techniques were applied on parasite-drug datasets. Hierarchical clustering results revealed that some treatments of parasites are similar and the others are different. Our results also show that there have not been major changes in the treatment of liver specific parasites for the past four decades. We investigated the reasons for the challenge of drug discovery and development for parasitic diseases. We believe that our results will make an important contribution to medical research, clinical studies and pharmaceutical research.

#### ACKNOWLEDGMENT

We would like to thank Antonio Jose Jimeno Yepes, Dietrich Rebholz Schuhmann and Rabin Saba for their help and contributions.

#### REFERENCES

- [1] A. R. Renslo, J. H. Mckerrow, "Drug discovery and development for neglected parasitic diseases, *Nature Chemical Biology*, vol. 2, no.12, 2006.
- [2] L. A. Marcos, A. Terashima, E. Gotuzzo, "Update on hepatobiliary flukes: fascioliasis, opisthorchiasis and clonorchiasis, *Current Opinion in Infectious Diseases*, 2008, pp 523-530.
- [3] N. Uramoto, H. Matsuzawa, T. Nagano, A. Murakami, H. Takeuchi, K. Takedo, "A Text Mining System for Knowledge Discovery from Biomedical Documents", *IBM Systems Journal*, vol. 43, Issue.3, pp 516-533.
- [4] W. Zhou, N. D. Smalheiser., C. Yu, "A tutorial on information retrieval: basic terms and concepts, *Journal of Biomedical Discovery and Collaboration*", 2006, pp 1-8.
- [5] D. Rebholz-Schuhmann, M. Arregui, A.J.J. Yepes, H. Kirsch, G. Neadic, "Automatic Text Analysis Based on Web Services", *Handout for the ISMB 2007 Tutorial*, ISMB, Vienna, 20.07.2007.
- [6] D. Rebholz-Schuhmann, H. Kirsch, S. Gaudan, M. Arregui, G. Neadic, "Annotation and Disambiguation of Semantic Types in Biomedical Text: a Cascaded Approach to Named Entity Recognition", *NLPXML '06 Proceedings of the 5th Workshop on NLP and XML*, 2005.
- [7] D. S. Wishart, C. Knox, A. C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang & J. Woolsey, "DrugBank: a comprehensive resource for in silico drug discovery and exploration", *Nucleic Acids Research*, 34, D668-D672, 2006.
- [8] S. M. Holland, "Cluster Analysis", Department of Geology, University of Georgia, Athens, GA 30602-2501, 2006.
- [9] J. W. Beckstead, "Using Hierarchical Cluster Analysis in Nursing Research", *Western Journal of Nursing Research*, Vol. 24, No. 307, pp-307-319, 2002.
- [10] K. Vipin, "Introduction to Data Mining", Addison-Wesley, 2006.
- [11] A. J. J. Wood, "Drug Therapy", *The England Journal of Medicine*, 1996, pp 1178-1184.
- [12] A. C. White, "Nitazoxanide: a new broad spectrum antiparasitic agent", *Expert Rev. Anti-infect.* 2004, pp 43-49.
- [13] P. Trouiller, P. L. Olliaro, "Drug development output from 1975 to 1996: what proportion for tropical diseases. *International Journal of Infectious Diseases*", 1998; 3: 61-63.
- [14] A. Vlachos, "Evaluating and combining biomedical named entity recognition systems", In *Poster Proceedings of BioNLP at ACL*, Prague, 2007.



## Reliability Analysis of Healthcare System

Elena Zaitseva, Vitaly Levashenko,  
Miroslav Rusin  
University of Zilina,  
ul.Univerzitna 8215/1, 01026 Zilina,  
Slovakia  
Email: {Elena.Zaitseva,  
Vitaly.Levashenko,  
Miroslav.Rusin}@fri.uniza.sk

**Abstract**—Modern system is complex and includes different types of components such as software, hardware, human factor. Reliability is principal property of this system. The importance analysis is one of approaches in reliability engineering. Application of this approach for healthcare system is considered in this paper. The importance reliability analysis allows estimating the influence of every healthcare system component to the system reliability, its functioning and failure.

### I. INTRODUCTION

THE PRINCIPAL goal of IT application in medicine is improvement and conditioning of medical care [1]–[3]. Modern healthcare systems have to reduce problems and difficulties in diagnosing and treatment of diseases, and have to perfect patient care. Therefore the healthcare has to be characterised by high reliability first of all and reliability analysis of such system is important problem.

There are investigations in reliability analysis of healthcare system. This area includes some concepts that can be declared as reliability analysis of medical equipment and devices [3]–[5] and human reliability analysis in medicine [4], [6], [7]. Unfortunately, these concepts develop independently, only in paper [4] problems of reliability analysis of technical part and human factor have been considered but different methods for their analysis have been proposed. It is caused by reliability engineering state, where there are some independent areas of investigation, for example, as software reliability analysis, hardware reliability analysis, human reliability analysis (HRA). Methods of estimation and quantification of these objectives aren't interchangeable. Therefore reliability analysis system with different types of component needs new methods. These methods have to allow estimation of such system based on unified methodology. Declaration of this problem for the healthcare system has been presented in [8].

According to [8] the healthcare system includes four components of different types (Fig.1): hardware, software, human factor and organization factor. In the paper [8] there

have been shown that the hardware and software components unite in one component for the healthcare system. This component is named as technical component, but this component is separated from other two components: special technical component and basic technical component. The first of them includes special equipment, devices and software (for example, magnetic resonance imaging scanners). The second component corresponds with basic equipment and software as personal computer, operating system, database and etc. The human factor and organization factor have been interpreted as two components.

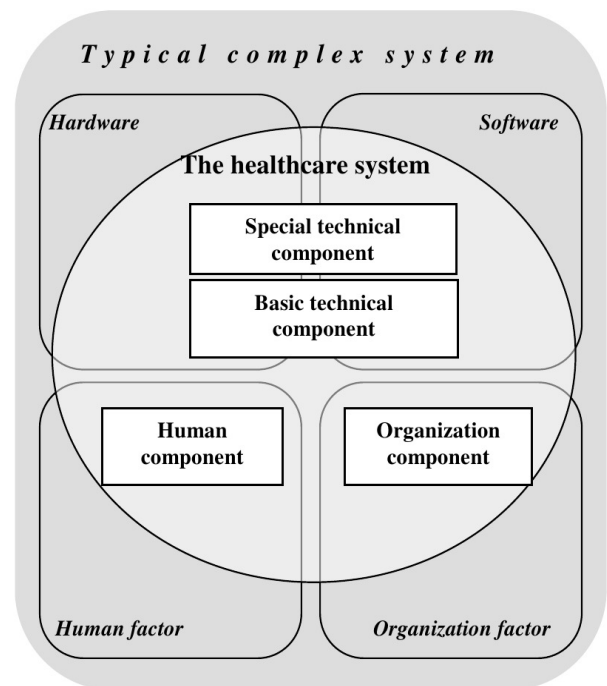


Fig. 1 The healthcare system typical structure for reliability analysis

In this paper reliability analysis of the healthcare system (Fig.1) is developed. The influence of system component states (some levels of functioning and failure) to the system reliability is investigated and quantified based on unified methodology. In other words the probabilities of the healthcare system performance levels are calculated against changes of the system component state changes.

This work was supported by grant VEGA and project "CREATING A NEW DIAGNOSTIC ALGORITHM FOR SELECTED CANCER DISEASES" co-financed from EU sources and European Regional Development Fund.

## II. RELIABILITY ANALYSIS OF HEALTHCARE SYSTEM

### A. Background and mathematical model

The basic reliability concept is defined as the probability that the system will perform its intended function during a period of running time without any failure. A fault is an erroneous state of the system. Although the definitions of fault are different for different systems and in different situations, a fault is always an existing part in the system and it can be removed by correcting the erroneous part of the system. New tendencies in reliability engineering have been defined in [9], some of they are:

- detail analysis of changes of the system reliability states from perfect function to failure;
- priority analysis of causes of the system failure, e.g. discover causes and mechanisms of failure and to identify consequences;
- development of methods for the system reliability analysis in design.

These tendencies have been taken into account in process of any system reliability analysis that includes next steps:

- the quantification of the system model;
- the representation and modelling of the system;
- the quantification of the system reliability (definition of reliability indexes and measures for the system evaluation).

Two steps of the system process analysis considered with definition of the mathematical model. This model has to allow estimation some levels of the system reliability changes. Binary-State System (BSS) and Multi-State System (MSS) are basic mathematical models in reliability analysis. BSS is used for description of initial system as system with two states: reliable and unreliable. But this model doesn't allow quantifying different levels of the system reliability. MSS is mathematical model in reliability analysis that is used for description system with some (more than two) levels of performance (availability, reliability) [9], [10]. MSS allows presenting the analyzable system in more detail than traditional Binary-State System.

The MSS and each of  $n$  components can be in one of  $m$  possible states: from the complete failure (it is 0) to the perfect functioning (it is  $m-1$ ). A structure function is one of typical representations of MSS [10], [11]. This function of a MSS of  $n$  components is denoted as:

$$\phi(x_1, \dots, x_n) = \phi(\mathbf{x}): \{0, \dots, m-1\}^n \rightarrow \{0, \dots, m-1\}, \quad (1)$$

where  $x_i$  is the  $i$ -th component;  $\mathbf{x} = (x_1, \dots, x_n)$  is vector of components states; values of a MSS reliability (structure function  $\phi(\mathbf{x})$ ) and its component state (variables  $x_i, i = 1, \dots, n-1$ ) change from zero to  $(m-1)$ .

Need to say that for the structure function (1) there are next assumptions that will be used in the system reliability estimation [12]:

- the structure function is monotone and  $\phi(\mathbf{s}) = s$  ( $s \in \{0, \dots, m-1\}$ );
- all components are  $s$ -independent and are relevant to the system.

Every system component states  $x_i$  is characterized by probability of the performance rate:

$$p_{i,s} = \Pr\{x_i = s\}, \quad s = 0, \dots, m-1 \quad (2)$$

The principal advantage of the system representation by the structure function (1) is definition of this function for any system with different complexity and structure.

There are different directions for quantification of MSS behaviour. One of them is importance analysis [12]–[14].

Importance analysis is used for MSS reliability estimation depending on the system structure and its components states. Quantification is indicated by importance measure. They have been widely used as tools for identifying system weaknesses, and to prioritise reliability improvement activities. MSS importance measures are probabilities that the system has the reliability level  $h$  ( $h = 1, \dots, m-1$ ) if the  $i$ -th system component states is  $s$  ( $s = 1, \dots, m-1$ ). Different combinations of the system reliability levels  $h$  and components states  $s$  allow investigating boundary system state and system states that take priority of failure.

Note one more significant aspect of the importance analysis. Some types of importance measures can be calculated for the system in design. Therefore this system quantification method can be used for the system reliability estimation in design.

The theoretical aspects of MSS importance analysis have been investigated since first paper in MSS analysis [15]. These investigations were developed in papers [12]–[16]. Importance measures for system with two performance level and multi-state components and their definitions by output performance measure have been considered in [12]. Universal generating function method has been used for importance analysis in [12], [16]. Composite importance measures for MSS estimation have been proposed in [14]. New method based on Logical Differential Calculus for importance analysis of MSS has been considered in paper [11], [17] and new type of importance measures has been proposed. These measures have been named as Dynamic Reliability Indices (DRIs). The importance analysis method based on Logical Differential Calculus is demonstrable, intuitive and is characterized by simple calculation.

Therefore MSS importance analysis is actual approach in reliability engineering because allows:

- to investigate the system behaviour in detail that include the quantification of different level of reliability;
- to examine causes of the system failure;
- to estimate the system reliability analysis in design.

The algorithm for the healthcare system reliability estimation by importance analysis based on typical process of the estimation is in Fig.2.

According to the algorithm in Fig.2 number  $m$  of performance (reliability or availability) levels for the system and its components for estimation of this system is defined firstly. Then the structure function as mathematical model of this system is determined taking into account the number of performance levels. For example, consider the healthcare system for the Decision Support System for Early



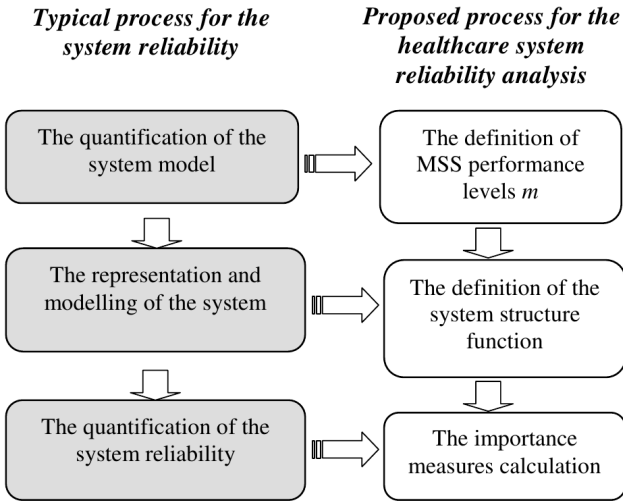


Fig. 2 The healthcare system reliability analysis process

Diagnostics in Oncology (DSSEDO) that have been described in [8]. The system structure can be interpreted as typical for healthcare system in Fig.1. Define for this system number of performance levels as  $m = 3$ . The structure function of this system is defined as:

$$\phi(x) = \text{OR}(\text{AND}(x_1, x_2), \text{AND}(x_1, x_3, x_4)), \quad (3)$$

where  $x_1$  is performance level of the special devices;  $x_2$  is performance level of the basic devices;  $x_3$  and  $x_4$  is performance level of the human and organization components of the system;  $\text{OR}(y, z) = \max(y, z)$ ;  $\text{AND}(y, z) = \min(y, z)$ .

Therefore the structure function (3) is mathematical model for the DSSEDO that is used for estimation and quantification of its performance or reliability (the detail description of this function is in Table I).

*B. Direct Partial Logic Derivative*

The mathematical tool of Multiple-Valued Logic (MVL) as Logical Differential Calculus is used for calculation of importance analysis. The MSS structure function is interpreted as MVL function in this case. The Logical Differential Calculus is mathematical tool that permits to analysis changes in function depending of changes of its variables. Therefore evaluate influence of every system component state change to level of MSS reliability by Direct Partial Logic Derivative (this approach is part of Logical Differential Calculus). Direct Partial Logic Derivative reflects the change in the value of the MVL function when the values of variables change.

A Direct Partial Logic Derivative with respect to  $i$ -th variable for a MSS reliability analysis has been defined in [17] as:

$$\begin{aligned} \partial\phi(j \rightarrow \tilde{j})/\partial x_i(a \rightarrow \tilde{a}) = \\ = \begin{cases} 1, & \text{if } \phi(a_i, x) = j \text{ and } \phi(\tilde{a}_i, x) = \tilde{j} \\ 0, & \text{other} \end{cases} \quad (4) \end{aligned}$$

TABLE I. TRUTH TABLE OF STRUCTURE FUNCTION (3)

$x_1 x_2 x_3 x_4$	$\phi(x)$	$x_1 x_2 x_3 x_4$	$\phi(x)$	$x_1 x_2 x_3 x_4$	$\phi(x)$
0 0 0 0	0	1 0 0 0	0	2 0 0 0	0
0 0 0 1	0	1 0 0 1	0	2 0 0 1	0
0 0 0 2	0	1 0 0 2	0	2 0 0 2	0
0 0 1 0	0	1 0 1 0	0	2 0 1 0	0
0 0 1 1	0	1 0 1 1	1	2 0 1 1	1
0 0 1 2	0	1 0 1 2	1	2 0 1 2	1
0 0 2 0	0	1 0 2 0	0	2 0 2 0	0
0 0 2 1	0	1 0 2 1	1	2 0 2 1	1
0 0 2 2	0	1 0 2 2	1	2 0 2 2	2
0 1 0 0	0	1 1 0 0	1	2 1 0 0	1
0 1 0 1	0	1 1 0 1	1	2 1 0 1	1
0 1 0 2	0	1 1 0 2	1	2 1 0 2	1
0 1 1 0	0	1 1 1 0	1	2 1 1 0	1
0 1 1 1	0	1 1 1 1	1	2 1 1 1	1
0 1 1 2	0	1 1 1 2	1	2 1 1 2	1
0 1 2 0	0	1 1 2 0	1	2 1 2 0	1
0 1 2 1	0	1 1 2 1	1	2 1 2 1	1
0 1 2 2	0	1 1 2 2	1	2 1 2 2	2
0 2 0 0	0	1 2 0 0	1	2 2 0 0	2
0 2 0 1	0	1 2 0 1	1	2 2 0 1	2
0 2 0 2	0	1 2 0 2	1	2 2 0 2	2
0 2 1 0	0	1 2 1 0	1	2 2 1 0	2
0 2 1 1	0	1 2 1 1	1	2 2 1 1	2
0 2 1 2	0	1 2 1 2	1	2 2 1 2	2
0 2 2 0	0	1 2 2 0	1	2 2 2 0	2
0 2 2 1	0	1 2 2 1	1	2 2 2 1	2
0 2 2 2	0	1 2 2 2	1	2 2 2 2	2

where  $\phi(\bullet_i, x) = \phi(x_1, \dots, x_{i-1}, \bullet_i, x_{i+1}, \dots, x_n)$  is value of structure function;  $\tilde{a} \neq a$ ,  $\tilde{j} \neq j$  and  $a, j, \tilde{a}, \tilde{j} \in \{0, \dots, m-1\}$ .

For monotone structure function the changes from  $a$  to  $\tilde{a}$  and from  $j$  to  $\tilde{j}$  can be defined as changes from  $a$  to  $\tilde{a} = (a-1)$  or  $\tilde{a} = (a+1)$  and from  $j$  to  $\tilde{j} = (j-1)$  or  $\tilde{j} = (j+1)$  accordingly. These changes are caused by gradual type of reliability changes without jumps too.

The Direct Partial Logic Derivative allows to calculate the system boundary states for which change the  $i$ -th component state from  $a$  to  $\tilde{a}$  cause changes of the system performance level from  $j$  to  $\tilde{j}$ . These states correspond to the nonzero values of the derivative (4). For example, for the healthcare system with structure function (3) boundary states of the system performance level reduction for the first component are in Table II. These states are computed by Direct Partial Logic Derivative that is indicated in Table II too.

Therefore according to the Table II the first component state change from 2 to 1 doesn't cause the system failure (change from 1 to 0) and the failure of this component doesn't influence to the system performance level change from 2 to 1, because the Direct Partial Logic Derivatives  $\partial\phi(1 \rightarrow 0)/x_1(2 \rightarrow 1)$  and  $\partial\phi(2 \rightarrow 1)/x_1(1 \rightarrow 0)$  have zero value only. But break down of the first component and its deterioration cause failure and degradation of the system

TABLE II.  
BOUNDARY STATES FOR THE FIRST COMPONENT OF HEALTHCARE  
SYSTEM WITH STRUCTURE FUNCTION (3)

$x_2, x_3, x_4$	$\frac{\partial \phi(1 \rightarrow 0)}{\partial x_1(1 \rightarrow 0)}$	$\frac{\partial \phi(1 \rightarrow 0)}{\partial x_1(2 \rightarrow 1)}$	$\frac{\partial \phi(2 \rightarrow 1)}{\partial x_1(1 \rightarrow 0)}$	$\frac{\partial \phi(2 \rightarrow 1)}{\partial x_1(2 \rightarrow 1)}$
0 0 0	0	0	0	0
0 0 1	0	0	0	0
0 0 2	0	0	0	0
0 1 0	0	0	0	0
0 1 1	1	0	0	0
0 1 2	1	0	0	0
0 2 0	0	0	0	0
0 2 1	1	0	0	0
0 2 2	1	0	0	1
1 0 0	1	0	0	0
1 0 1	1	0	0	0
1 0 2	1	0	0	0
1 1 0	1	0	0	0
1 1 1	1	0	0	0
1 1 2	1	0	0	0
1 2 0	1	0	0	0
1 2 1	1	0	0	0
1 2 2	1	0	0	1
2 0 0	1	0	0	1
2 0 1	1	0	0	1
2 0 2	1	0	0	1
2 1 0	1	0	0	1
2 1 1	1	0	0	1
2 1 2	1	0	0	1
2 2 0	1	0	0	1
2 2 1	1	0	0	1
2 2 2	1	0	0	1

accordantly (derivatives  $\frac{\partial \phi(1 \rightarrow 0)}{\partial x_1(1 \rightarrow 0)}$  and  $\frac{\partial \phi(2 \rightarrow 1)}{\partial x_1(2 \rightarrow 1)}$  have nonzero values that correspond to the boundary system states).

Investigation of the boundary states of the system is important problem but set of boundary state has high dimensionality and isn't acceptable well for practical application. Therefore in engineering problem probability measures for the system reliability or performance are used as a rule.

### III. HEALTHCARE SYSTEM RELIABILITY ANALYSIS

#### A. Healthcare system probability state

MSS probability state,  $R(j)$ , is one of the best known MSS reliability measures [12]. It is the probability that system performance level is equal to the level  $j$ :

$$R(j) = \Pr\{\phi(\mathbf{x}) = j\}, j \in \{0, 1, \dots, m-1\}. \quad (5)$$

For example, for the healthcare system with structure function (3) can be computed system state probabilities based on its structure function:

$$R(0) = p_{1,0} + (p_{1,1} + p_{1,2}) \cdot p_{2,0} \cdot (p_{3,0} + (p_{3,1} + p_{3,2}) \cdot p_{4,0}),$$

$$R(1) = (p_{1,1} + p_{1,2}) \cdot p_{2,0} \cdot (p_{3,1} + p_{3,2}) \cdot (p_{4,1} + p_{4,2}) + p_{1,1} \cdot (p_{2,1} + p_{2,2}) + p_{1,2} \cdot p_{2,1} \cdot (p_{3,0} + p_{3,1}) \cdot (p_{4,0} + p_{4,1}),$$

$$R(2) = p_{1,2} \cdot p_{2,1} \cdot p_{3,2} \cdot p_{4,2} + p_{1,2} \cdot p_{2,2}.$$

#### B. MSS Importance Measures

Probability states (5) don't enable the analysis of the change in system reliability that is caused by a change in component states. Importance analysis of the healthcare system allows estimating the influence of every system component state changes to system performance. Consider some of importance measures and their calculation by Direct Partial Logic Derivative.

*Structural Importance* (SI) is one of the simplest measures of component importance and this measure is concentrated on the topological aspects of the system. According to definition in papers [13], [18] this measure determines the proportion of working states of system in which the working of the  $i$ -th component makes the difference between system failure and its working. SI of MSS for the  $i$ -th component state  $s$  is probability of this system performance level  $j$  decrement if the  $i$ -th component state changes from  $s$  to  $s-1$  depending on topological properties of system:

$$I_S(s_i | j) = \frac{\rho_i^{s,j}}{m^{n-1}}, \quad (6)$$

where  $\rho_i^{s,j}$  is number of system states when the change component state from  $s$  to  $s-1$  results the system performance level decrement and this number is calculated as numbers of nonzero values of Direct Partial Logic Derivatives (4).

There is one more definition of SI [11]. It is modified SI that represent of the  $i$ -th system component state change influence to MSS performance level decrement for boundary system state. In terms of Direct Partial Logic Derivatives (4) modified SI is determined as:

$$I_{MS}(s_i | j) = \frac{\rho_i^{s,j}}{\rho_i^{(s_i, j)}} \quad (7)$$

where  $\rho_i^{s,j}$  is defined in (6);  $\rho_i^{(s_i, j)}$  is number of boundary system states when  $\phi(s_i, \mathbf{x}) = j$  (it is computed by structure function (1)).

Modified SI  $I_{MS}$  is probability of MSS performance decrement depending on the  $i$ -th component state change and boundary system states. A system component with maximal value of the SI measure ( $I_S$  and  $I_{MS}$ ) has most influence to MSS and this component failure causes high possibility of MSS failure [11], [13].

SI and modified SI measures don't depend on components state probability (2) and characterize only topological aspects of MSS performance. These measures are used for prevention system analysis or reliability analysis in step of a system design previously.

*Birnbaum Importance* (BI) of a given component is defined as the probability that such component is critical to MSS functioning [13], [14], [19]. This measure has been defined for traditional system with two states firstly as:

$$I_B(x_i) = |\Pr\{\phi(\mathbf{x}) = 1, x_i = 1\} - \Pr\{\phi(\mathbf{x}) = 1, x_i = 0\}|$$

But mathematical and logical generalization of this measure for MSS has some interpretations. So in paper [12] proposed definition of BI for system with two performance level that consists of multi-state components. Authors of the paper [14] considered definition of BI of MSS failure analysis. Than in paper [18], [20] new modifications of BI and algorithms for calculation based on different methodological approach have been proposed. One more interpretation of BI for MSS in terms of Logical Differential Calculus has been presented in paper [11]. According to this definition, BI is probabilistic measure that can be interpreted as rate at which the MSS fails as the  $i$ -th system component state decreases:

$$I_B(s_i | j) = \left| \Pr\{\phi(x) |_{x_i=s} = j\} - \Pr\{\phi(x) |_{x_i=s-1} = j\} \right|, \quad (8)$$

where

$$\Pr\{\phi(x) |_{x_i=s} = j\} = \sum p_{1,a_1} \cdots p_{i-1,a_{i-1}} p_{i+1,a_{i+1}} \cdots p_{n,a_n}$$

if  $\phi(x) = j$  and  $x_i = s$  for  $a_w = \{0, \dots, m-1\}$ ,  $w = 1, \dots, n$  and  $w \neq i$ ;  $s = \{1, \dots, m-1\}$ .

BI measures (8) depend on the structure of the system and states of the other components, but is independent of the actual state of the  $i$ -th component.

Consider the definition of *Criticality Importance* (CI) that is the probability that the  $i$ -th system component is relevant to MSS performance decrement if it has failed or has diminished state. For the system with two performance level this measure is considered in [19] in detail. For MSS this measure can be defined as probability of the MSS performance reduction if the state of the  $i$ -th system component has changed from  $s$  to  $s-1$ :

$$I_C(s_i) = I_B(s_i | j) \cdot \frac{p_{i,s-1}}{R(j)}, \quad (9)$$

where  $I_B(s_i | j)$  is the  $i$ -th system component BI measure (8);  $p_{i,s-1}$  is probability of the  $i$ -th system component state  $s-1$  (2) and  $R(j)$  is probability of system state  $j$  that is defined in accordance with (5);  $s = \{1, \dots, m-1\}$

The CI measure (9) correct BI for unreliability or lower state of the  $i$ -th component relative. This measure is useful, if the component has high BI and low probability of investigated state with respect of MSS performance decrement. In this case the  $i$ -th component CI is low.

*Fussell-Vesely Importance* (FVI) measure quantifying the maximum decrement in system reliability caused by the  $i$ -th system component state deterioration [12], [14]. By other words this measure represents the contribution of each component to the system and for the system with two performance levels is calculated by next equation [21]:

$$I_{FV}(x_i) = \frac{\Pr\{\phi(x)=0\} - \Pr\{\phi(x)=0 | x_i=1\}}{\Pr\{\phi(x)=0\}}.$$

FVI for MSS represents probabilistic measure of the  $i$ -th component state deterioration influence to the system performance level decrement:

$$I_{FV}(s_i | j) = 1 - \frac{\Pr\{\phi(s_i, \mathbf{x}) = j\}}{R(j)} \quad (10)$$

where

$$\Pr\{s_i, \phi(x) = j\} = \sum p_{1,a_1} \cdots p_{i,s_i} \cdots p_{n,a_n} \quad \text{if}$$

$\phi(x) = j$  and  $x_i = s$  for  $a_w = \{0, \dots, m-1\}$ ,  $w = 1, \dots, n$  and  $w \neq i$ ;  $s = \{1, \dots, m-1\}$ .

The calculation of FVI measure is similar to algorithm for computation of BI measure.

*Reliability Achievement Worth* (RAW) and *Reliability Reduction Worth* (RRW) are two importance measures and both represent adjustments of the improvement potential to MSS unreliability. RAW for Binary-State System (system that has only two performance level as function and failure) indicates the increase in the system unreliability when the  $i$ -th component is failed and this measure is defined as [21]:

$$I_{RAW}(x_i) = \frac{\Pr\{\phi(0_i, x) = 0\}}{F}$$

According to the papers [12], [14] RAW for MSS is defined as the ratio of MSS unreliability if the  $i$ -th component state has decrease:

$$I_{RAW}(s_i | j) = \frac{\Pr\{\phi(s_i-1, x) = j\}}{R(j)}, \quad (11)$$

where  $s = \{1, \dots, m-1\}$ .

RRW can be interpreted as opposite importance measure to RAW and for Binary-State System is defined as [21]:

$$I_{RRW}(x_i) = \frac{F}{\Pr\{\phi(1_i, x) = 0\}}.$$

Generalization of this equation and representation of RRW in [12], [14] allows defining RRW for MSS as importance measure quantifies potential damage caused to the MSS by the  $i$ -th system component:

$$I_{RRW}(s_i | j) = \frac{R(j)}{\Pr\{\phi(s_i, x) = j\}}, \quad (12)$$

where  $s = \{1, \dots, m-1\}$ .

There is one more type of importance measures for MSS that are *Dynamic Reliability Indices* (DRIs). These measures have been defined in paper [11], [17]. DRIs allow to estimate component relevant to MSS and to quantify the influence of this component state change to the MSS performance. There are two groups of DRIs: *Component Dynamic Reliability Indices* (CDRIs) and *Dynamic Integrated Reliability Indices* (DIRIs).

CDRI indicates the influence of the  $i$ -th component state change to MSS performance level change [17]. This

definition of CDRI is similar to definition of modified SI, but CDRI for MSS failure take into consideration two probabilities: (a) the probability of MSS performance level decrease caused by the  $i$ -th component state reduction and (b) the probability of this component state:

$$I_{CDRI}(s_i|j) = I_{MS}(s_i|j) \cdot p_{i,s_i-1} \quad (13)$$

where  $I_S(x_i|j)$  is the modified SI (7);  $p_{i,s_i}$  is probability of component (2).

DIRI is the probability of MSS performance level decrement that caused by the one of system components state deterioration. DIRIs allow estimate probability of MSS failure caused by some system component (one of  $n$ ):

$$I_{DIRI}(s|j) = \sum_{\substack{q \neq i \\ \dot{c}}}^n \dot{c} \dot{c} \quad (14)$$

#### IV. EXAMPLE OF HEALTHCARE SYSTEM IMPORTANCE ANALYSIS

Consider the healthcare system in Fig. 1. The structure function of such healthcare system is declared based on an expertise for every real system. This function is defined based on the expert knowledge and influence form the area of the system application. For example, the Decision Support System for Early Diagnostics in Oncology (DSSEDO) has structure function (3) that is described in Table I. The MSS mathematical model of this system has three levels of performance ( $m = 3$ ) and four components ( $n = 4$ ). In Table I the system component state 0 considers to the component failure; the component state 1 is component functioning with some unimportant restriction; the component state 2 is perfect functioning. The component probabilities in Table III have been determined for this system by the expertise.

TABLE III. COMPONENT PROBABILITIES

$i$	$m$	0	1	2
1		0.1	0.2	0.7
2		0.1	0.4	0.5
3		0.2	0.4	0.3
4		0.3	0.5	0.3

So the system state probabilities (5) for this healthcare system are calculated based on component probabilities in Table III:

$$R(0) = p_{1,0} + (p_{1,1} + p_{1,2}) \cdot p_{2,0} \cdot (p_{3,0} + (p_{3,1} + p_{3,2}) \cdot p_{4,0}) = 0.137,$$

$$R(1) = (p_{1,1} + p_{1,2}) \cdot p_{2,0} \cdot (p_{3,1} + p_{3,2}) \cdot (p_{4,1} + p_{4,2}) + p_{1,1} \cdot (p_{2,1} + p_{2,2}) + p_{1,2} \cdot p_{2,1} \cdot (p_{3,0} + p_{3,1}) \cdot (p_{4,0} + p_{4,1}) = 0.488,$$

$$R(2) = p_{1,2} \cdot p_{2,1} \cdot p_{3,2} \cdot p_{4,2} + p_{1,2} \cdot p_{2,2} = 0.375$$

Therefore the performance level 1 of the healthcare system is more probably than system perfect functioning (the

performance level 2) and system failure that have probabilities 0.137 and 0.375 accordingly.

Importance measures of this system are in Table IV. According to the data in Table IV the first system component change has maximal influence to the system reliability. Therefore correct functioning of special devices is important condition for reliability of the healthcare system with structure function (3). But need to say that the modification of the structure function of this system causes change of the importance analysis result. The positive result of importance analysis will be obtained based on investigation of some structure function of this system. The impediment for this analysis is caused by generation of structure function based on expert knowledge only that is subjective.

TABLE II. IMPORTANCE MEASURES FOR THE SYSTEM WITH STRUCTURE FUNCTION (3)

$i$	1	2	3	4
<b>Importance measures</b>				
$I_S(x_i 1)$	0.222	0.123	0.049	0.049
$I_S(x_i 2)$	0.123	0.099	0.025	0.025
$I_S(x_i)$	0.173	0.111	0.027	0.027
$I_{MS}(x_i 1)$	1	0.588	0.308	0.308
$I_{MS}(x_i 2)$	1	0.889	0.400	0.400
$I_{MS}(x_i)$	1	0.739	0.354	0.354
$I_B(x_i 1)$	0.456	0.248	0.063	0.010
$I_B(x_i 2)$	0.800	0.452	0.050	0.105
$I_B(x_i)$	0.628	0.350	0.032	0.058
$I_C(x_i 1)$	0.095	0.052	0.026	0.006
$I_C(x_i 2)$	0.419	0.473	0.053	0.137
$I_C(x_i)$	0.257	0.263	0.040	0.072
$I_{CDRI}(x_i 1)$	0.100	0.059	0.062	0.062
$I_{CDRI}(x_i 2)$	0.200	0.356	0.160	0.200
$I_{CDRI}(x_i)$	0.150	0.208	0.111	0.131

#### V. CONCLUSION

In this paper new algorithms of IM calculation for MSS analysis are considered. These algorithms are implemented based on methods of MVL as Logical Differential Calculus and MDD. But investigated MSS has one principal assumption: system and all its components have  $m-1$  different performance levels and state unreliability. In next investigation we are going to develop this mathematical approach for estimation of MSS without this assumption. Structure function of such MSS is defined as:

$$\phi(x): \{0, \dots, m_i-1\} \times \dots \times \{0, \dots, m_n-1\} \rightarrow \{0, \dots, M-1\}.$$

In this case MSS consists of  $n$  components and has  $M$  levels of the performance rate from complete failure (this level corresponds with 0) to the perfect functioning (this level is interpreted as  $M-1$ ). Each of  $n$  MSS components is characterized by different performance level and the  $i$ -th component has  $m_i$  possible states: from the complete failure (it is 0) to the perfect functioning (it is  $m_i-1$ ).

#### REFERENCES

[1] D. Castro, "Meeting National and International Goals for Improving Health Care: The Role of Information Technology in Medical Research," in Proc. on Atlanta Conference on Science and Innovation Policy, 2009, pp. 1-9.

- [2] Keng Siau, "Health Care Informatics," *IEEE Transactions on Information Technology in Biomedicine*, vol. 7(1), pp.1-7, Jan. 2003.
- [3] T. Cohen, "Medical and Information Technologies Converge," *IEEE Engineering in Medicine and Biology Magazine*, vol. 23(3), pp. 59-65, March 2004.
- [4] B. S. Dhillon, "Human and Medical Device Reliability," in *Handbook of Reliability Engineering*, H. Pham (ed), Springer, 2003.
- [5] A. Taleb-Bendiab, D.England, et al., "A principled approach to the design of healthcare systems: Autonomy vs. governance," *Reliability Engineering and System Safety*, vol. 91(12), pp. 1576-1585, Dec. 2006.
- [6] M. S. Bogner, *Human Error in Medicine*, Lawrence Erlbaum Associates, Hillsdale, N.J. 1994.
- [7] M. Lyons, S. Adams, et al., "Human reliability analysis in healthcare: A review of techniques," *Int. Journal of Risk & Safety in Medicine*, vol. 16(4), pp.223-237, April 2004.
- [8] E. Zaitseva, "Reliability Analysis Methods for Healthcare system," in *Proc. of the IEEE 3rd Int Conf on Human System Interaction*, Rzeszow, Poland, May 13-15, 2010, pp.211-216.
- [9] E. Zio, "Reliability engineering: Old problems and new challenges," *Reliability Engineering and System Safety*, vol. 94(2), pp. 125-141, Feb. 2009
- [10] A. Lisnianski, G.Levitin, *Multi-State System Reliability. Assessment, Optimization and Applications*, World scientific, 2003.
- [11] E. Zaitseva, "Importance Analysis of Multi-State System by tools of Differential Logical Calculus," in *Reliability, Risk and Safety. Theory and Applications*, R. Bris, C. Guedes, S. Martorell (eds), CRC Press, 2010, pp.1579-1584.
- [12] G. Levitin, L. Podofillini, E. Zio, "Generalised Importance Measures for Multi-State Elements Based on Performance Level Restrictions," *Reliability Engineering and System Safety*, vol. 82(3), pp. 287-298, March 2003.
- [13] F. C. Meng, "On Some Structural Importance of System Components," *Journal of Data Science*, No. 7, pp. 277-283, July 2009.
- [14] J. E. Ramirez-Marquez, D. W. Coit, "Composite Importance Measures for Multi-State Systems with Multi-State Components," *IEEE Trans. on Reliability*, vol.54(3), pp.517 – 529, March 2005.
- [15] D. A. Butler, "A complete importance ranking for components of binary coherent systems, with extensions to multi-state systems," *Naval Research Logistics Quarterly* 26, 1979, pp.565-578.
- [16] G. Levitin, A.Lisnianski, "Importance and sensitivity analysis of multi-state systems using the universal generating function method," *Reliability Engineering and System Safety*, vol. 65(3), pp.271-282, March 1999.
- [17] E. Zaitseva, V. Levashenko, "Dynamic Reliability Indices for parallel, series and  $k$ -out-of- $n$  Multi-State System," in *Proc. of the IEEE 52nd Annual Symposium Reliability & Maintainability (RAMS)*, Newport Beach, USA, 23-26 January, 2006, pp.253 – 259
- [18] S. Wu, "Joint importance of multistate systems," *Computers and Industrial Engineering*, vol. 49, pp.63-75, 2005.
- [19] R. M. Fricks, K. S. Trivedi, "Importance Analysis with Markov Chains," in *Proc. of the IEEE 49th Annual Reliability & Maintainability Symposium (RAMS)*, Tampa, USA, 27-30 January, 2003, pp.89 -95.
- [20] E. Zio, M. Marella, L. Podofillini, "Importance measures-based prioritization for improving the performance of multi-state systems: application to the railway industry," *Reliability Engineering and System Safety*, vol. 92(10), pp.1303-1314, Oct. 2007.
- [21] Y.-R. Chang, S. V. Amari, S. Y. Kuo, "Computing System Failure Frequencies and Reliability Importance Measures Using OBDD," *IEEE Transactions on Computers*, vol.53(1), pp.54-68, Jan. 2004.



# 1<sup>st</sup> International Workshop on Advances in Semantic Information Retrieval

**B**AG-OF-WORDS document representation is still the basis for many information retrieval models. Recent advances in semantic technologies have resulted in methods and tools that allow creating and managing domain knowledge. They influence the way and form of representing documents in the memory of computers, approaches to analyze documents, techniques to mine and retrieve, etc. Searching for video, voice and speech raises new challenging problems to retrieval systems.

The aim of this workshop is to discover new challenges in various branches of Internet technologies applied to semantic information retrieval, to foster opportunities for international collaboration between scientists all over the world, and to explore directions for further research. The workshop addresses semantic information retrieval theory, as well as important matters related to Web tools used in practice.

The topics and areas include but are not limited to:

- Models for document representations
- Ontology for semantic information retrieval
- Ontology alignment, mapping and merging
- Search and ranking
- Semantic multimedia retrieval
- Natural language semantic processing
- Evaluation methodologies for semantic search and retrieval
- Query interfaces
- Visualization of the retrieved results
- Domain-specific semantic applications

## PROGRAM COMMITTEE

**Troels Andreasen**, Roskilde University, Denmark

**Shu-Ching Chen**, Florida International University, USA

**Ernesto Damiani**, University of Milan, Italy

**Vladimir Dobrynin**, Saint Petersburg State University, Russia

**Anna Fensel**, Telecommunications Research Center Vienna, Austria

**Adina Magda Florea**, University "Politehnica" of Bucharest, Romania

**Fredric C. Gey**, University of California, Berkeley, USA

**Yannis Haralambous**, Institut Télécom, France

**Enrique Herrera-Viedma**, University of Granada, Spain

**Wladyslaw Homenda**, Warsaw Univ. of Technology, Poland

**Qun Jin**, Waseda University, Japan

**Janusz Kacprzyk**, Systems Research Institute Polish Academy of Sciences, Poland

**Tuomo Kakkonen**, University of Eastern Finland, Finland

**Kamen Kanev**, Shizuoka University, Japan

**Cristian Lai**, Center of Advanced Studies, Research and Development in Sardinia, Italy

**Simone Ludwig**, North Dakota State University, USA

**Robert Meersman**, Free University of Brussels, Belgium

**Nikolay Mirenkov**, University of Aizu, Japan

**Kendall Nygard**, North Dakota State University, USA

**Ryuichi Oka**, University of Aizu, Japan

**Vladimir A. Oleshchuk**, University of Agder, Norway

**Incheon Paik**, University of Aizu, Japan

**Maciej Piasecki**, Wroclaw University of Technology, Poland

**Evgeny Pyshkin**, Saint-Petersburg State Polytechnical University, Russia

**Marek Reformat**, University of Alberta, Canada

**Roman Y. Shtykh**, Rakuten Inc., Japan

**Ryszard Tadeusiewicz**, AGH University of Science and Technology, Poland

**Eloisa Vargiu**, University of Cagliari, Italy

**Alexander Vazhenin**, University of Aizu, Japan

**Haofen Wang**, Shanghai Jiao Tong University, China

**Shih-Hung Wu**, Chaoyang University of Technology, Taiwan

**Slawomir Zadrozny**, Systems Research Institute Polish Academy of Sciences, Poland

## ORGANIZING COMMITTEE

**Vitaly Klyuev**, University of Aizu, Japan

**Maxim Mozgovoy**, University of Aizu, Japan





## Fuzzy Cognitive Map Theory for the Political Domain

Sameera Al Shayji  
Department of Information Systems and Computing, Brunel University, Uxbridge, Middlesex, U.K  
Email: samira@fasttelco.com

Nahla El Zant El Kadhi  
Department of Management Information Systems, Ahlia University, Manama, Kingdom of Bahrain  
Email: znahla@yahoo.fr

Zidong Wang  
Department of Information Systems and Computing, Brunel University, Uxbridge, Middlesex, U.K  
Email: Zidong.Wang@brunel.ac.uk

**Abstract**—An acceleration of regional and international events contributes to the increasing challenges in political decision making, especially the decision to strengthen bilateral economic relationships between friendly nations. Obviously this becomes one of the critical decisions. Typically, such decisions are influenced by certain factors and variables that are based on heterogeneous and vague information. A serious problem that the decision maker faces is the difficulty in building efficient political decision support systems (DSS) with heterogeneous factors. The basic concept is a linguistic variable whose values are words rather than numbers and therefore closer to human intuition. Fuzzy logic is based on natural language and is tolerant of imprecise data. Furthermore, fuzzy cognitive mapping (FCM) is particularly applicable in the soft knowledge domains such as political science. In this paper, a FCM scheme is proposed to demonstrate the causal inter-relationship between certain factors in order to provide insight into better understanding about the interdependencies of these factors. It presents fuzzy causal algebra for governing causal propagation on FCMs.

### I. INTRODUCTION

THE considerable knowledge has been generated, organized, and digitized in various governmental sectors, but it is still not readily accessible at any time or in any convenient place for decision makers. Existing relationships between countries can be described from a variety of perspectives, such as historical, respectful, friendly, neighboring, traditional, religious, political, and economic aspects. Apart from such a variety of relationships, almost all nations seek to build bridges of cooperation with other countries in various ways. One way to build these relationships is to strengthen the economic relationships, wherein the decision maker must take into consideration many factors and variables that influence the promotion of an economic relationship. This information and these factors are diversified and may involve different sectors. From a research viewpoint, the challenges lie in recognizing, finding and extracting these different variables. A conscientious decision maker who takes responsibility for promoting and strengthening bilateral economic relationships needs access to well-structured information relevant to his/her decisions. Unfortunately, in reality, the basic concept of this information is a linguistic variable, that is, a variable whose values are words rather than numbers across different domains including the political and investment domains. This makes it extremely difficult for the decision

maker to understand the concepts, restraints, and facts that exist in these domains. Due to the various factors that influence the decisions intended to strengthen economic relationships with other countries, there is an urgent need to develop a proper system that analyzes the data gathered from different sectors and produces precise and certain outputs that could be useful to the decision makers. In Kuwait, the scattered data mostly lies in various governmental sectors, including the Kuwait Fund for Development, the Kuwait Investment Authority, the Ministry of Foreign Affairs, the Prime Minister's Office, the Embassies of Kuwait, and the Decision Maker's Office. Due to various forms of political data that exist in so many contrasting domains, certain imperfections, such as imprecision, uncertainty and ambiguity, inevitably appear. A popular way to handle the scattered data is to construct the so-called fuzzy ontology as presented in [20]. Ontology is useful for sharing knowledge, building consensus and constructing knowledge-based systems. So far, many ontology systems have been implemented such as the Semantic Web. More recent work in the field of ontology in governments was presented by Ortiz-Rodriguez [15]. The problem fundamental to develop an ontology system is to respect the diversity of languages and concept presentations in the world while encouraging the exchange of information. Despite initial efforts in this area, there has been little literature concerning fuzzy-logic-based ontology especially from a political domain. The purpose of this paper is, therefore, to shorten such a gap by proposing a prototype architecture for generating ontology in order to extract knowledge from various data sources. These sources may take on various forms, such as textual data, knowledge-based data, and regular documents.

### II. METHODOLOGY

Different methodological approaches for building ontology have been proposed in the literature [3, 6, 11]. Until now, there has been no standard method for building ontology. The approach described in this paper is adopted from the ontology modeling approach of Noy and McGuinness [13] and Fernandez-Lopez [11]. The process begins with the extraction of key concepts and relationships between sets of information, and then proceeds to integrate fuzzy logic with ontology. The ontology includes information about important concepts in each domain. For the purposes of the ontology

we refer the readers to [22]. The framework for bilateral trade ontology with semantic or linguistic relations in the investment domain was first presented in [22] as a case study. In this paper, we aim to present a case study that contains clear concepts for the political and investment domains. On the other hand, Fuzzy Cognitive Mapping (FCM) is especially applicable in the soft knowledge domains (e.g., political science, military science, history, international relations, and political election at governmental levels [24]). For this reason, we propose FCM's simulation to demonstrate the causal inter-relationship between certain factors and variables in the political and investment domains that influence top political decision makers so as to strengthen bilateral economic relationships between friendly nations. Note that the FCM simulation provides insight into and better understanding of the interdependencies of these factors, which provides a constructive contribution to the decision making process. Our proposed ontology will cover the two main important government sectors in Kuwait: the Kuwait Investment Authority and the Ministry of Foreign Affairs. In general, it is important to first know how to model these two sectors and present their major trends, actions, norms and principles. It is crucial to describe the domains and the relationship between them, and to understand the complexity involved in making decisions as well as how ontology building can be helpful and beneficial for decision makers. Ontology editors create and manipulate ontology. Examples of such editing tools include Protégé, which is an ontology editor and knowledge-base framework, and Fuzzy Logic Toolbox, which extends the technical computing environment with tools that design systems based on fuzzy logic. We will integrate the fuzzy logic membership as a value that reflects the strength of an inter-concept relationship and is consistently used to represent pairs of concepts across ontology. More work about fuzzy set and membership can be found in [22], where the concept consistency is dealt with by means of a fixed numeric value. Concept consistency is computed as a function of ngth of all the relations associated to the concept. In [22], an object paradigm (OP) ontology was presented for important concepts in order to capture a high level of knowledge to facilitate the work of decision makers in the decision-making process of the political field. The OP ontology approach was used to determine and specify important concepts in the political and investment domains for ontology conceptualization. A more expressive, reusable, and objective object paradigm ontology was presented by Al Asswad, Al-Debei, de Cesare, and Lycett [23]. Accordingly, in this paper, we will present the concept by using the OWL editing tools ontology. The aim of using OWL is to integrate the concept of the political and investment domains. It is worth mentioning that, according to the World Wide Web Consortium (W3C), the most recent development in standard ontology language is OWL. Like Protégé, OWL makes it possible for users to describe concepts, but it also provides new facilities. More justification in regard to use Protégé was presented in [22]. A survey of existing ontology editing tools was done in Islam et al. [14], and the comparison between them was presented in [13].

#### *A. Fuzzy Cognitive Mapping (FCM)*

FCM is a fuzzy-graph structure for representing causal reasoning with a fuzzy relation to a causal concept [26]. Fuzzy cognitive maps are especially applicable in the soft knowledge domain (e.g., political science, military science, history, international relations, and organization theory [24]). Fuzzy logic generated from fuzzy theory and FCM is a collaboration between fuzzy logic and concept mapping. FCM is used to demonstrate knowledge of the causality of concepts to define a system in a domain starting with fuzzy weights quantified by numbers or words [25]. In [24], FCM was used to demonstrate the impact of drug addiction in America. In fact, FCM is an extension of a cognitive way of representing weighted causal links, where an expert's domain knowledge is merged with a collaborative knowledge that helps in the decision-making process. As a soft-system modeling and mapping approach, FCM combines aspects of qualitative methods with the advantages of quantitative (i.e., causal algebra) methods. In a FCM, the positive (+) and the negative (-) signs above each arrowed line provide a causal relationship whereby each fuzzy concept is linked with another one. In this sense, the FCM is a cognitive map of relations between the elements (e.g., concepts, events, project resources) that enables the computation of the impact of these elements on each other, where the theory behind that computation is fuzzy logic. Since FCMs are signed fuzzy non-hierarchical digraphs [25], metrics can be used for further computations, and causal conceptual centrality in cognitive maps can be defined with adjacency-matrix [26]. So far, FCMs have been used to construct a diagram to represent words, ideas, and variables linked and arranged around a central idea, in order to generate and classify ideas to help the decision-making process. In [27], Khoubati, Themistocleous, and Irani developed a FCM based model to evaluate the adoption of Enterprise Application Integration (EAI) in healthcare organization, where the FCM simulation was conducted to demonstrate the causal interrelationships between the EAI adoption factors that influence the EAI adoption in healthcare organization [27].

#### *B. Fuzzy Cognitive Map Model for Evaluation*

An FCM is a method for graphically representing state variables within a dynamic system through links that signify cause and effect relationships using fuzzy weight quantified via numbers or words [25]. Experts can translate such words into numeric values and present them graphically to show which factors are contributory and to what degree they contribute. The main advantage of FCM is its flexibility. It can always accept additional variables, so factors can be included at any time. Nine steps are employed in designing a cognitive map: (1) identification of factors, (2) specification of relationships, (3) levels of all factors, (4) intensities of causal effects, (5) changeable factors versus dependent factors, (6) simulating the fuzzy cognitive map, (7) modifying the fuzzy cognitive map, (8) simulating the modified fuzzy cognitive map, and (9) conclusion. More description about these steps was presented in [24]. These flexible and efficient steps have been extensively used for planning and decision-making in numerous fields, see e.g. political and Middle East crisis

[24]. In our research, when preparing a fuzzy cognitive map, the first step entails the identification for factors (concepts) and the following eighteen factors (concepts) are selected based on several events in the region. The regional and international events have contributed to the increasing challenges actors face in political decision making, particularly the decision to strengthen bilateral economic relationships with friendly nations. Fig. 1 presents the FCM model that provides insight into factors influencing such decisions, where (F1) means the degree of promoting bilateral economic relationships with friendly nations security and stability, (F2) the political stability, (F3) the threat of terrorism, (F4) the threat of nuclear war, (F5) the threat of provocation, (F6) the multiple parties involved, (F7) the multiple ethnic groups involved, (F8) the multiple sects involved, (F9) the loans, (F10) other financial aid, (F11) the nation regional and international attitudes, (F12) the peace in the Middle East, (F13) the status of agreement, and (F14) type of agreement. Table I shows a number of initial rows vectors (connection-matrix) demonstrations to present the interrelation of some factors with other factors. Table I presents the different factors and the relationship between them. For example, table I shows clearly the negative impact of factor (4) that represent the threat of nuclear war on strengthen the economic relationship. The goal set for this hypothetical fuzzy cognitive matrix is to determine how the threat of nuclear war and others factors impacts the strengthen economic bilateral relationship. Hence, F4 is one of the most critical factors in this fuzzy cognitive map. In addition the threat of terrorism (F3) is the major negative cause of political stability. The value -1 represents full negative causal effect, whereas +1 full positive causal effect. Zero denotes no causal effect.

Table I: Connection-matrix presentation of factors.

	F1	F2	F3	F4	F5	F6	F7
F1	(+) 0.475 Usually	(+) 0.475 Usually	(+) 0.475 Usually	(+) 0.475 Usually	(+) 0.475 Usually	(+) 0.475 Usually	(+) 0.475 Usually
F2	(+) 1 Always	(+) 1 Always	(-) 1 Very Much	(-) 0.475 Usually	(-) 0.475 Usually	(-) 0.475 Usually	(-) 0.475 Usually
F3	(-) 1 Always	(-) 1 Always	0	(+) 0.75 Very Much	(+) 0.75 Very Much	(+) 0.75 Very Much	(+) 0.75 Very Much
F4	(-) 1 Always	(-) 1 Always	(+) 0.75 Very Much	(+) 0.75 Very Much	(+) 0.475 Usually	(+) 0.475 Usually	(+) 0.475 Usually
F5	(-) 0.75 Very Much	(-) 0.75 Very Much	(+) 0.75 Very Much	(+) 0.375 Some times	(+) 0.375 Some times	(+) 0.375 Some times	(+) 0.375 Some times
F6	(-) 0.75 Very	(-) 0.75 Very	(+) 0.75 Very	(+) 0.375 Some	(+) 0.375 Some	(+) 0.375 Some	(+) 0.375 Some
F7	(-) 0.75 Very	(-) 0.75 Very	(+) 0.75 Very	(+) 0.375 Some	(+) 0.375 Some	(+) 0.375 Some	(+) 0.375 Some

C. Use of Fuzzy Cosal Algebra to Clarify The Relationships Between Factors

This work seeks to clarify the relationships between concepts, and elucidate the positive or negative effects on each concept while enhancing the knowledge clarification of the relationships. Furthermore a FCM graph structure allows systematic causal propagation, (i.e. forward and backward chaining) and arrows sequentially contribute to the convenient identification of the cause's, effects and affected factors. FCM allows knowledge bases to expand by connecting

additional concepts. Fuzzy causal algebra governs causal propagation and causal combination on within FCM Kosko. Fuzzy logic algebra is created by abstracting operations from multiplication and addition that are defined on a fuzzily partially set P of causal values [26].The algebra that is developed depends only on the partial ordering on P, the range set of the fuzzy causal edge function e, and on general fuzzy-graph properties (connections). Bart Kosko presented the indirect and total causal effects on cognitive maps in [26].Koko explained the causal effect on cognitive node  $C_i$  to concept  $C_j$ , say  $C_i \rightarrow C_{k_1} \rightarrow \dots \rightarrow C_{k_n} \rightarrow C_j$ , which can be denoted with ordered indices as  $(i, k_1, \dots, k_n, j)$ . Then the indirect effect from  $C_i$  to  $C_j$  is the causality  $C_i$  imparts to  $C_j$ . The total effect of  $C_i$  on  $C_j$  is all the indirect effect causality that  $C_i$  imparts to  $C_j$ . The operations of indirect and total effect correspond to multiplication and addition of real numbers and a causal calculus of signs (+ and -). Interpreting the indirect effect operator, I, as some minimum operator and the total effect operator, T, as some maximum operator, these operators depending only on P's partial order and the simplest of these operators are the minimum and the maximum value. Formally, let there be m-many causal paths from  $C_i$  to  $C_j$ :  $(i, k^1_1, k^1_2, \dots, k^1_{n_i}, j)$  for  $1 \leq l \leq m$ , let  $I_l(C_i, C_j)$  denote the indirect effect of concept  $C_i$  on concept  $C_j$  on the  $l$ th causal path. Let  $T(C_i, C_j)$  denote the total effect of  $C_i$  on  $C_j$  over all m causal path. Then

$$I_l(C_i, C_j) = \min \{ e(C_p, C_{p+1}) : (p, p+1) \in (i, k^1_1, \dots, k^1_{n_i}, j) \},$$

$$T(C_i, C_j) = \max_{1 \leq l \leq m} I_l(C_i, C_j)$$

Where p and p+1 are contiguous left-to-right path indices. Hence, the indirect effect amounts specify the weakest causal link in a path and the total effect operation amounts to specifying the strongest of the weakest links. For example the concepts variables are represented by nodes, such as: C1: Threat of nuclear war, C2: Security stability, C3: Nation regional and international attitudes, C4: The type of the agreement, C5: The status of the agreement, C6: Relation type and C7: Strengthen investment indicators. Figure 1 has 7 variables that describe the impact of some conditions on strengthening bilateral economic relationships and causal variables. For example  $(C_1 \rightarrow C_2, C_1)$  that are said to impact  $C_4$ . Such is apparent because  $C_1$  is the causal variable where  $C_4$  is the effect variable. Suppose that the causal values are given by p {none  $\leq$  some  $\leq$  much  $\leq$  a lot}. The FCM appears below

In figure 1, phrases such as "much" and "a lot" denote the causal relationship between concepts. A fuzzy rule, causal link, or connection is defined by each arrow in the figure: a plus (+) represents a causal increase and a negative (-) represents a causal decrease. The causal paths from  $C_1$  to  $C_7$  are nine, the direct effect is (1,7), so the eight indirect effects of  $C_1$  to  $C_7$  are : (1,2,4,5,6,7), (1,2,4,6,7), (1,2,6,7), (1,4,5,6,7), (1,3,4,6,7), (1,3,4,5,6,7), (1,4,6,7), and (1,6,7). The eight indirect effects of  $C_1$  on  $C_7$  can be described as follows:

$$I_1(C_1, C_7) = \min \{ e_{12}, e_{24}, e_{45}, e_{56}, e_{67} \} = \min \{ \text{a lot, much, a lot, some, a lot} \} = \text{some}$$

$$I_2(C_1, C_7) = \min \{ e_{12}, e_{24}, e_{46}, e_{67} \} = \min \{ \text{a lot, much, a lot, a lot} \} = \text{much}$$

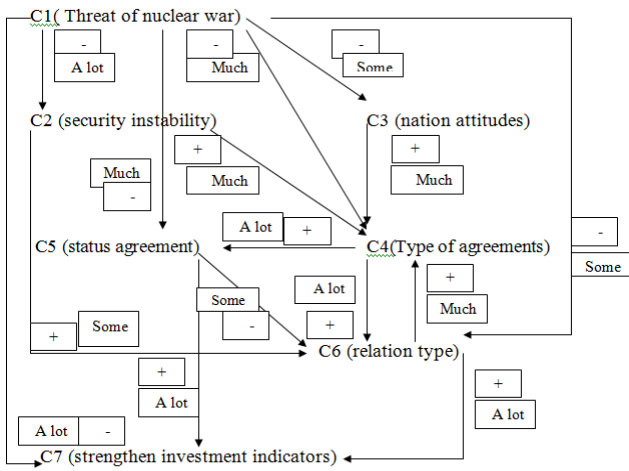


Fig. 1: A fuzzy cognitive map on the impact of strengthening economic bilateral relationship.

$I_3(C_1, C_7) = \min\{e_{12}, e_{26}, e_{67}\} = \min\{\text{a lot, some, a lot}\} = \text{some}$   
 $I_4(C_1, C_7) = \min\{e_{14}, e_{45}, e_{56}, e_{67}\} = \min\{\text{much, a lot, some, a lot}\} = \text{some}$   
 $I_5(C_1, C_7) = \min\{e_{13}, e_{34}, e_{46}, e_{67}\} = \min\{\text{some, much, much, a lot}\} = \text{some}$   
 $I_6(C_1, C_7) = \min\{e_{13}, e_{34}, e_{45}, e_{56}, e_{67}\} = \min\{\text{some, much, much, some, a lot}\} = \text{some}$   
 $I_7(C_1, C_7) = \min\{e_{14}, e_{46}, e_{67}\} = \min\{\text{much, a lot, a lot}\} = \text{much}$   
 $I_8(C_1, C_7) = \min\{e_{16}, e_{67}\} = \min\{\text{some, a lot}\} = \text{some}$   
 Thus the total effect of  $C_1$  on  $C_7$  is  $T(C_1, C_7) = \max\{I_1(C_1, C_7), I_2(C_1, C_7), I_3(C_1, C_7), I_4(C_1, C_7), I_5(C_1, C_7), I_6(C_1, C_7), I_7(C_1, C_7), I_8(C_1, C_7)\} = \max\{\text{some, much, some, some, some, some, much, some}\} = \text{much}$ .

Therefore,  $C_1$  impacts much causality to  $C_7$ . Now that the fuzzy conceptual  $C_i$  has been computed, the advantage is that the causal quality is established.

*D. Fuzzy Logic and Membership*

In recent years, the number and variety of applications of fuzzy logic have increased significantly. The most basic variables underlying fuzzy logic are linguistic variables. A linguistic variable is a variable whose values are words rather than numbers. Although words are inherently less precise than numbers, humans intuit the meaning of words more easily than that of numbers. Furthermore, computing with words exploits the tolerance for imprecision inherent in language. The aim of this section is to present a proposal that integrates fuzzy logic into ontology. Undoubtedly, the success of fuzzy logic applications lies in their ability to handle vague information. Fuzzy logic is especially useful in government applications, since information within governmental sectors is generally vague and requires a common language. In the political domain, one is unlikely to find a document that provides a precise definition for a fuzzy value, but one can usually find a linguistic qualifier. For example, one would not find information in a document numerically char-

acterizing the relation between country x and country y, but one might find the following information in such a document: “country x has a good relation with country y,” “country x has a very good relation with country y,” or “country x has weak relation with country y.” As another example, one might describe “existing bilateral relations” between countries from a variety of perspectives using a set of properties including the following: “historical,” “respectable,” “coalition country,” “antibody state” and “friendly.” Table II presents some examples of such semantic relations. It lists “StrongFriend” as a property of the concept “RelationName” to describe the nature of a relation in the bilateral relation domain. Thus, “StrongFriendRespect,” “WeakRespect,” “Respect,” and “StrongFriend” are properties describing the type of relation between two countries, which require human knowledge for interpretation. Table III presents causal weight to demonstrate FCM model in politic domain.

Table II:

Presentation of some semantic relations in “CountryClassification” and “RelationName” classes

Country Classification	Relation name	Country name
Coalition countries	Strong-Friend-Respect	a b c d e f
sectarian States	Respect	J k l
investment states	Strong-friend	b c d e a g y t
Arab states	Respect-culture	a b x b p k d
EU states	Strong-respect-friend	a b c d r
GCC	History-neighbour- Religion	A b d f c
States voted in favour of the issue of Kuwait	Encourage very strong	A b c e
Crisis States	weak	G w

So a method of making use of this kind of information is needed, especially in the political domain to help decision makers strengthen bilateral economic relationships between friendly nations. In the political domain, associating a numeric membership modifier to many situations is often necessary. Fuzzy logic allows users to model imprecise and vague data, combine different priority functions, and use any value between 1 and 0 as a logic value. It is based on natural languages in order to provide convenient methodologies for representing human knowledge [12]. Fuzzy logic is comprehensible, flexible, and tolerant of imprecise data. A fuzzy ontology describes the relation between the political domain and the investment domain with the semantic relation presented in Fig. 2. More description of this diagram can be seen in [22]. Here, we present the integration of data across different sectors and produce a seamless system permitting valid design support for top political decision makers by employing natural languages. One can convert an ontology into a fuzzy ontology by adding a relation weight to any relation, as discussed in [12, 19].

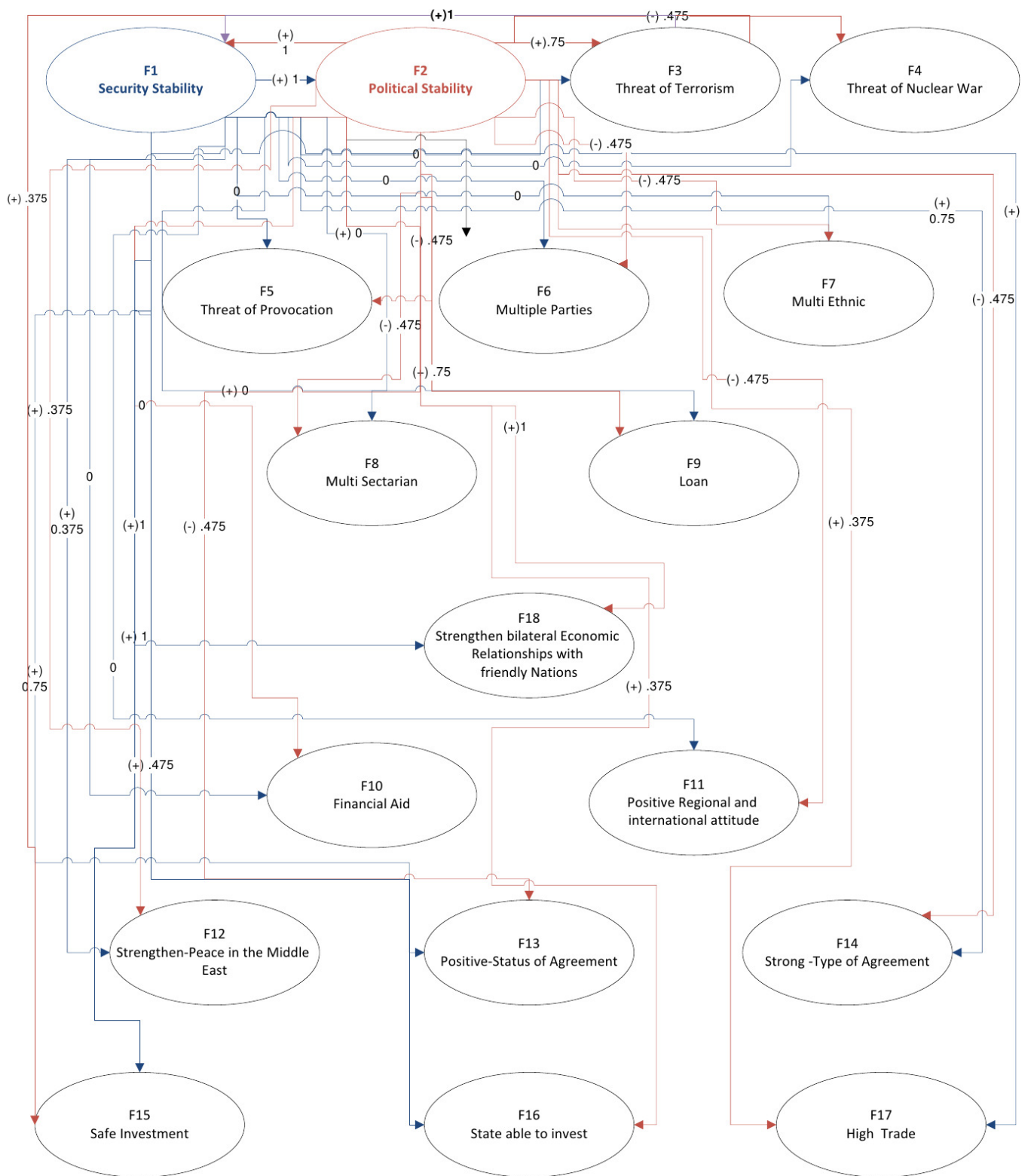


Fig. 1: FCM model presenting certain factors in the political and investment domains.



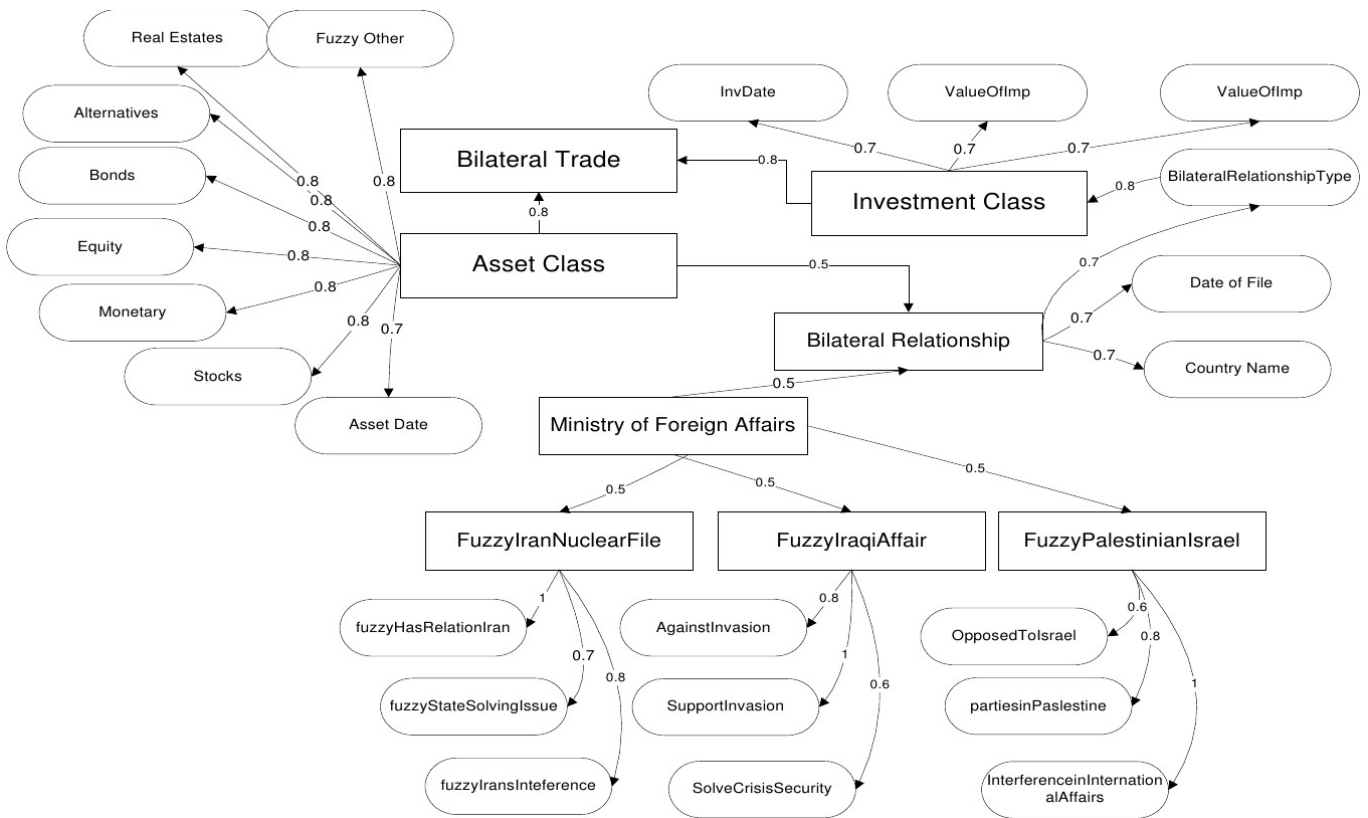


Fig. 2: Fuzzy ontology for the relation between the Ministry for Foreign Affairs and the Kuwait Investment Authority

Table III:  
Causal weight used for FCM model

Des	Weight
Never	0
Not Much	0.125
Sometimes	0.375
Usually	0.475
very much	0.75
Always	1

### III. THE MINISTRY OF FOREIGN AFFAIRS DOMAIN

In the Ministry of Foreign Affairs domain, one might ask questions during the decision-making process when attempting to strengthen bilateral economic relationships with other nations, especially about Iranian affairs, Iraqi affairs, and peace in the Middle East. For example, one might ask the following: Is this country interested in security and stability? Does this country intervene in the affairs of other countries? Will the strengthening of bilateral economic relationships lead to a reactivation of the peace process in the Middle East? The following subsections present each concept in detail.

#### A. Nuclear Affairs

To understand the position of other countries concerning nuclear issues, much information needs to be taken into account. Let us consider the cases that 1) country  $y$  deals with the nuclear power issue; and 2) the investment might be with country  $x$ . For example, does country  $x$  have an interest in the development of nuclear power? Does country  $x$  have a special relationship with who has nuclear power? Does country  $x$  influence actors in country  $y$ ? Does country  $y$  intervene in country  $x$  internal affairs? Does country  $x$  receive benefits from funding sources from  $y$ ? Is country  $x$  keen to solve the nuclear issue with country  $y$  peacefully? Is country  $x$  keen to call on  $y$  to disclose its nuclear reactors to international inspectors? Does country  $x$  agree with country  $y$  about the use of nuclear reactors for military purposes? Does country  $x$  agree with  $y$  about the use of nuclear reactors for peaceful purposes? Does country  $x$  have economic investments in or other relationships with country  $y$ ? Does  $y$  spend indirectly on country  $x$ ? Does country  $x$  agree with  $y$  provocative? Does country  $x$  agree with the positive international resolutions on the nuclear dossier? Does country  $x$  refuse to engage in military action against  $y$ ? Does country  $x$  refuse to participate in an economic blockade against  $y$ ? Does country  $x$  support  $y$  politically? Does country  $x$  have an established relationship with  $y$ ?

To answer these questions, one might use common primitive data types, such as the Boolean “yes,” “no,” “sometimes,” “always,” and “never.” As mentioned above, the Ministry of Foreign Affairs sector includes descriptions of the bilateral relations of other countries over time and information concerning international agreements, with dates and names. One can describe existing bilateral relations from a variety of perspectives using a set of properties including the following: “historical,” “respectful,” “friendly,” “solid,” “common interests,” “excellent,” “very good,” “good,” “acceptable,” “weak,” “diplomatic,” “political,” “economic,” “political and economic,” “strongly supports,” “sometimes supports,” “never supports,” “opponent,” and “unclear.”

#### B. Relation between neighboring countries

To understand the position of a country toward another, one might ask certain questions during the decision-making process (deciding whether to support or help the strengthening of bilateral economic relationships with other nations). For example, the friendly relations between two neighboring countries can turn sour due to several economic and diplomatic reasons. After the Iraqi invasion of Kuwait, for example, If the invasion occurs between two neighbors (x and y) and country z want to make investment with country x, this let country x has to consider many elements before making a decision to strengthen the economic relationship, for example was country z against the invasion? Did country z support the invasion? Does country z undertake efforts to end the crisis over country x or country y security plan? Does country z interfere in country x or country y internal affairs? Does country z endeavor to ensure the unity and the independence of country x or country y or both of them? Did country z vote to resolve the issue concerning country x or country y? Does this country support the withdrawal of U.S. troops from country x or country y under their security plan? Was this country against (or support) the recent invasion?

#### C. Peace in the Middle East

To understand the country’s position regarding the issue of peace in the Middle East, one might ask certain questions during the decision-making process (deciding whether to support or help to strengthen bilateral economic relationships with other nations). For example, does this country support the Arab Peace Initiative? Does this country work to unify Arab stances? How would one describe this country’s position on the reactivation of the peace process? Does this country look forward to seeing stability in the region? Does this country have a positive position regarding the challenges facing the region (yes, no, sometimes, or never)? What is this country’s position on dialogue and negotiation (positive or negative)? How would one describe this country’s position on the European Peace Initiative (positive or negative)? Does this country have interests in common with other countries in the region? Answers to these questions could include “yes,” “no,” “sometimes,” “never,” and “not clear.”

At the same time, one must understand the country’s position toward Palestinian issues. One might ask certain questions during the decision-making process (deciding whether

to support or help to strengthen bilateral economic relationships with other nations). For example, what is this country’s position on the occupation of Palestine? Does this country interfere in Palestine’s internal affairs? Does this country work for the realization of the Palestinian people’s rights? Does this country take the initiative to ensure a peaceful Palestine? Does this country support efforts for Palestinian reconciliation? Does this country seek a comprehensive peaceful solution to help Palestine?

#### IV. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a fuzzy ontology approach and discussed how to conduct this approach in two important governmental sectors in Kuwait: the Kuwait Investment Authority and the Ministry of Foreign Affairs. To build this ontology, we have provided a quantitative way of understanding how these sectors represent major trends by breaking them down into classes and subclasses. This helps to identify the proper ontological concepts for each sector and characterize the properties and elements the different sectors share, the entities in those classes, and the domain and relationships between them. A fuzzy ontology approach has been employed to provide insight into how knowledge can be represented and handled in order to offer decision makers the aid of an intelligent decision-making process. The application of FCM has helped to demonstrate the inter-relationships of influencing factors that political decision makers must take into account when deciding whether to support or help to strengthen bilateral relationships between Kuwait and friendly nations. A detailed analysis of this FCM model has been conducted to provide such decision makers with knowledge and understanding of the factors of successful investment. The present research has contributed to the process of making the decision to strengthen bilateral economic relationships with friendly countries.

#### REFERENCES

- [1] P. Alexopoulos, K. Kafentzis, X. Benetou, T. Tagaris, and P. Georgiolos, “Towards a generic fraud ontology in e-government,” in *Proceedings of the International Conference on Security and Cryptography*, Portugal, 2008, pp. pp. 421-436.
- [2] D. Apostolou, L. Stojanovic, T. P. Lobo, J. C. Miro, and A. Papadakis, “Configuring e-government services using ontologies,” in *Proceedings of the IFIP International Federation for Information Processing Conference*, vol. 189. Boston: Springer, 2005, pp. 141-155.
- [3] H. Beck and H. S. Pinto, “Overview of approach, methodologies, standards, and tools for ontologies,” in *Proceedings from the Third Agricultural Ontology Service (AOS) Workshop*, Gainesville, FL, 2002, p. 58.
- [4] F. Bettahar, C. Moulin, and J. P. Barthes, “Ontologies supporting e-government services,” in *Proceedings of the IEEE Artificial Intelligence Conference*, Portugal, 2005, pp. 1000-1005.
- [5] K. J. Bwalya, “Factors affecting adoption of e-government in Zambia,” *Elec. J. Inform Sys in Dev Count.*, vol. 38, pp. 1-13, 2009.
- [6] C. Calero, F. Ruiz, and M. Piattini, Eds., *Ontologies for Software Engineering and Software Technology*. Berlin: Springer-Verlag, 2006.
- [7] H. Claire, N. S. Jarvis, and W. Cooper, “Infometric and statistical diagnostics to provide artificially-intelligent support for spatial analysis: The example of interpolation,” *International Journal of Geographical Information Science*, Volume 17, Issue 6, 2003, Pages 495 – 516.

- [8] T. Herborn and M. Wimmer, "Process ontologies facilitating interoperability in e-government: A methodological framework," presented at Workshop on Semantics for Business Process Management, the 3rd Semantic Web Conference, Montenegro, June 2006.
- [9] J. Lee and S. Kim, "An intelligent priority decision making algorithm for competitive operators in list-based scheduling," *Int. J. Comp Sci and Net Sec.*, vol. 9, no. 1, , Jan. 2009.
- [10] J. Kaaya, "Implementing e-government services in East Africa: Assessing status through content analysis of government websites," *Elec. J. E-Gov.*, vol. 2, pp. 39-54, 2004.
- [11] M. Fernandez-Lopez, "Overview of methodologies for building ontologies," *J. Data & Know Eng.*, vol. 46, pp. pp. 41 – 64, 2003.
- [12] M. Abulaish and L. Dey, "Interoperability among distributed overlapping ontologies: A fuzzy ontology framework," in *Proceedings of the 2006 IEEE/IWC/ACM International Conference on Web Intelligence*, 2006, pp. insert page range.
- [13] N. Noy and D. McGuinness, "Ontology development 101: A guide to creating your first ontology," Stanford Knowledge Systems Laboratory and Stanford Medical Informatics, Stanford, CA, Rep. KSL-01-05 and SMI-2001-0880, 2001.
- [14] N. Islam, A. Z. Abbasi, and A. Zubair, "Semantic Web: Choosing the right methodologies, tools and standards," in *Proceedings of the International Conference on Information and Emerging Technologies*, Karachi, Pakistan, 2010, pp. 1-5.
- [15] F. Ortiz-Rodriguez, "Mexican e-government ontologies: An adaptation," in the *Proceedings of the Fourth International Latin American and Caribbean Conference for Engineering and Technology*, Mayagez, Puerto Rico, June 2006.
- [16] K. Ralf, "Towards ontology for e-document management in public administration – The case of Schleswig-Holstein," in the *Proceedings of the 36th International Conference on System Sciences*, Hawaii, USA: IEEE Computer Society, January 2006.
- [17] P. Salhofer, B. Stadhofer, and G. Tretter, "Ontology driven e-government," *Electron. J. of E-government*, vol. 7, pp. 415-424, 2008.
- [18] M. Salles, "Supporting public decision making: A progressive approach aided by an ontology," *Int. J. Decision Support Syst. Technology*, vol. 2, no. 1, pp. 21-35, 2010.
- [19] S. Calegari and D. Ciucci, "Integer fuzzy logic in ontologies," *Int. J. Approx. Reasoning* vol. 51, pp. 391-409, 2010.
- [20] U. Inyaem, P. Meesad, and C. D. T. Haruechaiyasak, "Construction of fuzzy ontology-based terrorism event extraction," *3rd Int. Conf. Knowledge Discovery and Data Mining*, 2010, IEEE. doi: 10.1109/WKDD.113.
- [21] S. Shayji, N. Kadhi, and Z. Wong, "On fuzzy-logic-based ontology decision support system for government sector," *12th WSEAS Int. Conf. Fuzzy Systems*, Brasov 2011, 34 .
- [22] S. Shayji, N. Kadhi, and Z. Wong, "Building ontology for political domain," *2011 Int. Conf. Semantic and Web Services*, UAS, 2011.
- [23] M. M. Al Asswad, M. M. Al-Debei, S. de Cesare, and M. Lycett, "Conceptual modeling and the quality of ontologies: A comparison between object-role modeling and the object paradigm," *Proc. 18th European Conf. Information Systems*, Pretoria, 2010.
- [24] G. J. Calais, "Fuzzy cognitive maps theory: Implications for interdisciplinary reading," *Nat. Implication*, vol. 2 no.1, 2008.
- [25] A. M. Sharif and Z. Irani, "Knowledge dependencies in fuzzy information systems evaluation," *Proceeding of the Eleventh Americas Conference on Information Systems*, Omaha, NE, USA, August, 2005.
- [26] B. Kosko, *Fuzzy Cognitive Maps*. London: Academic Press Inc, 65-75, 1986.
- [27] K. Khoumbati, M. Themistocleous, and Z. Irani, "Application of fuzzy simulation for evaluating enterprise application integration in healthcare organizations", European and Mediterranean Conference on Information System(EMCIS),Costa Blanca,Alicante,Spain,July 6-7 2006.



# Building a Model of Disease Symptoms Using Text Processing and Learning from Examples

Marek Jaszuk<sup>\*†</sup>, Grażyna Szostek<sup>†</sup>, Andrzej Walczak<sup>†\*</sup> and Leszek Puzio<sup>\*†</sup>

<sup>\*</sup>University of Information Technology and Management Rzeszów, Poland

<sup>†</sup>Military University of Technology Warsaw, Poland

Email: marek.jaszuk@gmail.com, grazyna.szostek@gmail.com

awalczak@wat.edu.pl, lpuzio@wsiz.rzeszow.pl

**Abstract**—The paper describes a methodology of building a semantic model of disease symptoms. The fundamental techniques used for creating the model are text analysis and learning from examples. The text analyser is used for extracting a set of symptom descriptions. The descriptions are a foundation for delivering a user interface, necessary for collecting patient cases. Given the cases a semantic model is built, which is achieved through clusterisation and statistical analysis of cases. The approach to creating the model eliminates the need of direct model manipulation, because the meaning is retrieved from association to diseases instead of purely linguistic interpretation of symptom descriptions. Detection of synonyms is also completely automatized.

## I. INTRODUCTION

**B**UILDING models of knowledge is a very important topic in the domain of artificial intelligence and knowledge management systems. The most common approach is associated with the intensively developed Semantic Web technology. This technology requires representing knowledge in the form of an ontology. Such models are typically built for some specific domains such as biomedical sciences, or various branches of business and industry. The fundamental element of every ontology is a set of concepts from the particular domain. The concepts are combined using a set of semantic relations defined within the ontology. As a result we get a hierarchic structure in the form of a directed graph with nodes being the ontology concepts (semantic classes), and the edges being the semantic relations.

Unfortunately the task of ontology building requires a lot of effort, and engagement of experts from the given field. One of the most important obstacles, that the ontology builders have to overcome, is the proper identification of concepts which should be used as the ontology nodes. It is assumed, that the nodes have to represent particular meanings instead of their possible verbal descriptions. Thus the ontology building process requires identifying synonyms among the verbal descriptions. To represent the meaning, usually one of the possible descriptions is chosen. The problem is, however, that particular verbal expressions can represent multiple meanings depending on the context of their use. As a consequence they can be classified as representatives of completely different meanings. Another difficulty are the subtle differences in meaning between the particular expressions. In consequence it is frequently very difficult to decide, whether the particular

expressions should be considered representatives of the same concept or their meaning is distinctive enough to separate them. All such decisions are left to the person constructing the ontology, and are the reason of frequent hesitations, which slow down the whole process.

The high labour consumption is not the only consequence of the difficulties mentioned above. The drawback of most ontologies constructed today is their highly subjective character. All individual decisions about defining particular classes and the possible relations between them influence the final shape of the constructed ontology. As a result a given domain can be described by many different models. This is obviously not what is desired. The domain knowledge is only one, and the properly constructed model should be independent of the particular persons building it.

Another obstacle against efficient ontology building is the large size of the models that need to be developed for real world problems. The size is a simple consequence of the huge number of concepts and the possible relations between them. Considering that the domain knowledge is usually contained in resources like books, technical articles, or the Internet, the ontology building process can be supported by extracting the important information from text. This approach is founded on a number of techniques coming from the natural language processing field (NLP). The purpose of using such methodologies is identification of concepts important for the domain and the possible relations between the terms. This approach resulted in a number of ontology learning systems that have already been created. Some of the most well known examples are: OntoLearn [1], Text-to-Onto [2], OntoGen [3], ASIUM [4], TextStorm/Clouds [5], SYNDIKATE [6], ISOLDE (Information System for Ontology Learning and Domain Exploration) [7]. There is number of approaches based on utilising clustering algorithms [8]–[10] for building ontologies. A good overview of the current state of the art in the field of ontology learning can be found in [11], [12]. To assess the ontology learning methodologies, several surveys have been made [13]–[15]. According to their findings most of the systems are semi-automated tools for supporting domain experts in creating ontologies. Complete automation and elimination of user involvement is hard and can be applied only in cases where high quality of the knowledge model is not obligatory.

In this paper we demonstrate an approach to building a model of medical symptoms in association with a set of diseases. The obstacles encountered during building this type of a model belong to the categories already mentioned. The first of them is the reach vocabulary used for describing symptoms, which takes effect in a huge size of the model to be created. The particular symptoms can be described by different combinations of words. The additional difficulty results from the fact that the descriptions which could be considered synonymic, do not always represent exactly the same meaning. There are frequently subtle differences in meaning between the possible alternatives. This results in additional difficulties in indentifying the important concepts. In consequence using standard methodologies requires a lot of effort and time to complete the task. Moreover the final result is always influenced by the personal habits, and knowledge of the model creator.

In our approach a different methodology is employed. It allows for avoiding the most important problems. One of the main assumptions is using NLP methods for text analysis and extracting information which could be important for the assumed task. This delivers a huge database of verbal constructions which are potential descriptions of symptoms. The second stage of the work is based on utilizing the database of verbal constructions. This approach is however completely different than the standard methodologies. It does not assume any direct manipulation of the model by human experts. This stage is replaced by collecting patient cases, and completing the model construction by training the system on these cases. The identification of concepts is achieved by applying clustering algorithm in the space of possible descriptions. In this way the clusters represent the particular meanings, which are the building blocks of the model. The advantage of this approach is that no human is responsible for identifying the meaning standing behind the verbal descriptions. The only expectation from the experts entering the cases is that they should describe the symptoms according to their best knowledge. To do that they can use the set of descriptions delivered by the text analyzer, but they are not restricted to it. They do not need to obey any special restrictions on the verbal constructions to be used. It is even advantageous if the cases are entered by different specialists having different habits. In this way the model structure is resistant to the subjectivity of experts taking part in its creation. Also the human effort during construction of the model is lesser than required during direct manipulation of the model. This is a consequence of the fact that conscious analysis of a complex model is replaced with a relatively simple task of describing cases.

It should also be mentioned that the described system is build for the Polish language. To be more precise, the language specific features are implemented in the module used for text processing. This module is responsible for identifying associations between words in text. The main features of the language, which influence the module structure is extensive inflection and free order of words in sentence. Using linguistic rules and sentence schemas, we are able to identify sets of

associated words forming tree structures. These structures are potential symptom descriptions. Of course a tree of words is not a natural knowledge representation for a human user. To increase the readability of the symptom representation, the tree structures are reduced to flat sequences of words, which resemble the original representation of knowledge extracted from sentences. Such descriptions form the initial database of symptoms which is further purified during the process of collecting cases and learning from examples. Except of the method of creating trees of associated words, the remaining part of the system is universal and free of language specific features. Thus it can easily be moved to another language, if an appropriate method of identifying associations between words would be developed.

The paper is organized as follows. Sec. II discusses the methodology of text processing used for extracting symptom descriptions. In Sec. III the process of collecting patient cases is presented. Sec. IV presents how the semantic model of diagnostic knowledge is built through clusterization and statistical analysis of cases. In Sec. V the results of experiments with text processing are presented.

## II. EXTRACTION OF SYMPTOM DESCRIPTIONS FROM TEXT

The text searching mechanism is founded on the observation, that from the perspective of verbal construction, every symptom descriptions have a common structure. This structure has a form of a tree of words with root being a noun in the nominative case. The case of course can be determined for inflective language like Polish. Every symptom description contains at least one noun in nominative. The branches of the symptom description tree are formed of the words associated to the root noun.

The discussed text analysis methodology has some common elements with other known algorithms. First of all it includes gramatical tagging of words. This task can be solved using several different approaches. Some of the examples are the Stanford POS tagger [16] or the MXPOST [17]. For Polish the most well known tagger is the TaKIPI [18]. Another issue is the analysis leading to finding the relations between words in a sentence. An example of a system realizing this task is the Multiparser [19]. Our approach is not directly based on any of the existing solutions, however, it contains some of their elements. The discussed system is strictly task specific, and developing our own solution allowed for introducing the necessary optimizations. Although this does not mean that the solution is not applicable to other domains after some minor modifications. We do not do any comparisons to other algorithms here. This is because the paper is devoted to presenting the general idea of the methodology designed to build the model of medical diagnostic knowledge. Although we realize that the comparison of the text analysis algorithms from the computational linguistics perspective is a very important issue and this will be the subject of separate study.

The process of extracting symptom descriptions from text consists of the following steps:

- 1) decomposition of text into sentences;

- 2) reading individual sentences, and morphological analysis of words;
- 3) disambiguation of morphological tags;
- 4) discovery of morphologically related words;
- 5) discovering relations using sentence schemas;
- 6) identification of nouns in the nominative case and building trees of words associated to every such noun;
- 7) reduction of every tree to a flat sequence of words;

The details of every step will be described in subsequent sections.

#### A. Specifics of the Polish Language

As already mentioned the text analysis strongly depends on the specific features of the language for which the system is developed. There is a number of characteristic elements of the Polish language which were taken into account while building the module. Below are listed the most important of them for the defined task [20]:

- inflection  
This is a language feature meaning, that words are inflected by case, number, person, etc. The Polish language belongs to the group of inflectional languages. Inflectional properties of words influence parsing sentences. The inflection allows for determining roles of words in sentences.
- discontinuity of phrases  
Elements of noun and verb phrases do not have to occur directly next to each other in a sentence.
- free order of words in sentences  
Words of a sentence may appear in different order without affecting the meaning of the whole sentence.
- lexical polysemy  
Lexical polysemy occurs when two or more words have the same form. For example, the form *drogi* (eng. *roads*) has two lexemes:
  - a) *droga* (eng. *the road*) - noun, plural, feminine gender,
  - b) *drogi* (eng. *dear*) - adjective, singular, masculine gender.
- syntactic polysemy  
Syntactic polysemy occurs when several forms of the same lexeme are identical. For example, morphological analysis of the form *okna* (eng. *windows*) will give a number of interpretations, including: *noun:singular:genitive*, *noun:plural:nominative*, *noun:plural:locative*.

#### B. Morphological Analyzer

The morphological analyzer is a very important resource used during text processing. Our system uses the Morfeusz software package [21]. It assigns one or several tags expressing potential morphological interpretations to the analyzed word (lexeme form). The analyzer is based on a system of tags developed for the IPI PAN Corpus [22], [23]. The contents of the tags includes the basic form of the lexeme, information about the part of speech (lexeme class - noun, adjective, verb,

etc.), number (singular or plural), case (nominative, genitive, etc.), gender (feminine, masculine, etc.), and a several other pieces of information. The analyzer data are represented in the form of finite state machines, which makes new word forms analysis impossible. Also analysis of the word context is not done (the program is not a tagger). So the problem of lexical and syntactic polysemy remains to be resolved.

#### C. Disambiguation of Words

Polysemy is an important factor influencing the effectiveness of discovering verbal associations. The morphological analyser generates multiple morphological tags for many of the words found in text, while only one of them is the correct one. Taking the incorrect tag leads to erroneously constructed tree of verbal associations, and as a result incorrectly extracted description. Disambiguation is thus important for reducing the number of errors in the results of text analysis.

Some of the tags can be eliminated *a priori* taking into account the character of the domain to which the text corpus refers. In this way we are able to eliminate all the tags including the vocative case, as this case is not used in medical texts. An example of a lexeme form eliminated in this way is *szybko* (eng. *fast*), which is an adverb, but could also be interpreted as the noun *szybka* (eng. *glass*) in the vocative case. Other examples of tags eliminated in this way include verbs in the imperative mood. An example is the lexeme form *dym* (eng. *smoke*) which can be interpreted as a noun in the nominative case, as well as a verb in the imperative mood. Of course for the medical texts only the first interpretation makes sense. Yet another example are depreciative forms of nouns, which tend to diminish in value the described entity. An example is *inne* (eng. *different*), which in medical texts is used only as an adjective, so the noun interpretation can be eliminated.

The second method of word disambiguation allows for reducing lexical polysemy. It is based on the observation, that the lexeme form interpretation which is inappropriate for the given context is not a subject of declination (by case, by number, by person, or by gender). As a result the number of forms in which a given lexeme appears in the text corpus is very limited (usually to one of the possible forms). As a result we are able to create a disambiguation dictionary consisting of lexemes inappropriate for the given domain. The lexemes are ignored when appropriate word interpretation is searched for. The examples of such lexemes are: *lewy* - the interpretation as an adjective is correct (eng. *left*), while the noun interpretation is ignored (bulgarian currency), *mały* - the adjective interpretation is correct (eng. *small*), the noun interpretation is ignored (eng. *young*), *normalna* - the adjective interpretation is correct (eng. *normal*), the noun interpretation is incorrect (eng. *normal (line)*).

The two presented methods of disambiguation do not guarantee removing the polysemy completely. The remaining ambiguities we try to resolve using linguistic rules, which is discussed in the subsequent section.

#### D. Identification of Word Associations

The objective of this stage is to eliminate the lexical and syntactic polysemy and identify relations between words using linguistic rules. There is a number of such rules which are characteristic for the Polish language, and their use in sentence construction indicates related words. Below are some of the most important rules used for disambiguation:

- linking preposition  
A compound consisting of preposition and a noun is expressed by inflectional noun ending, which is specific for the case acceptable in this link.
- links between nouns and nouns in genitive  
As it can be observed, when two nouns are directly next to each other in a sentence, the last of them is usually in the genitive case. This feature allows to disambiguate the category of the case for the second noun.
- links between nouns and adjectives  
The dependency between a noun and an adjective is expressed by the characteristic inflectional endings. These endings are characteristic for the number, case and gender, which are common for both of the words.

When the linguistic rules are applied we are able to identify the lexeme forms which are related and eliminate the lexemes which do not create relations. Of course this method does not allow for elimination of ambiguities completely. Sometimes there is more than one alternative of word association, which is allowed by the linguistic rules. In such a case all the possible variants of word associations are built, assuming that one of them is the one we are searching for. After applying the linguistic rules also the knowledge about the subject and the predicate of a sentence is collected. This knowledge will be used when applying sentence patterns.

Except the tasks mentioned above using the linguistic rules allows for establishing relations between words. Some of the most important relation types, resulting directly from the rules are listed below:

- noun - adjective  
It is a relation which occurs between a noun and the corresponding adjective, e.g. *pluco prawe* (eng. *right lung*), *wydzielina ropna* (eng. *purulent discharge*), *ciśnienie niskie* (eng. *low pressure*), etc.
- noun - noun in locative  
It is a relation between two nouns, where the second noun is in the locative case. Morphological analysis discovers only the argument in the locative case, which in case of symptoms specifies the place of occurrence, e.g. *w płucach* (eng. *in lungs*), *na powierzchni* (eng. *on surface*), *we krwi* (eng. *in blood*), etc. The argument specifying what occurred in the specified place remains to be found in the sentence. As it could be observed the noun in the locative case has also an associated preposition, which is a result of a separate rule.
- noun - noun in genitive  
This type of relation associates two nouns occurring in the text immediately next to each other, where the second

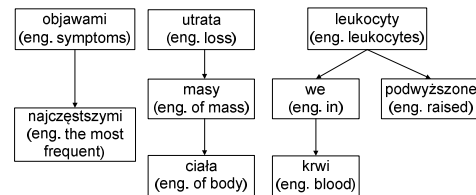


Fig. 1. Word associations extracted from a sample sentence

noun is in the genitive case. For example: *skóra głowy* (eng. *skin of the head*), *masa ciała* (eng. *body weight*), *grzybica stóp* (eng. *mycosis of feet*), etc.

Let us analyze a sample sentence:

*Podwyższone leukocyty we krwi i utrata masy ciała są najczęstszymi objawami.* (eng. *Increased leukocytes in blood and body weight loss are the most frequent symptoms.*)

The linguistic rules allow for generating the following set of relations from the sentence:

- noun - adjective: *objawami najczęstszymi* (eng. *frequent symptoms*);
- noun - noun in genitive: *utrata masy* (eng. *weight loss*);
- noun - noun in genitive: *masy ciała* (eng. *body weight*);
- noun - adjective: *leukocyty podwyższone* (eng. *increased leukocytes*);
- preposition - noun in locative *we krwi* (eng. *in blood*);
- noun - noun in locative: *? we krwi* (eng. *? in blood*).

In the last relation the preposition and the noun in locative were treated as one entity. This is because the relation refers to them as a whole. It can also be observed that the last relation has an unidentified element which is not indicated by any linguistic rule. Assuming that the noun fitting the relation is the closest noun before the noun in locative, we get the missing argument of the relation. The resulting relation is thus: *leukocyty we krwi*. It should be remembered, however, that in general case resolving the missing argument of the rule is not so simple, because of free word ordering.

The obtained word associations are presented in Fig. 1. As we can see the mechanism delivers three separate graph structures. To identify the graphs which are interesting for our purposes, we need to remember, that every symptom description contains at least one noun in the nominative case. This noun is the main element of the description verbal construction. In terms of the graph structures this means that it is the root node of the tree of words. When looking at the structures from Fig. 1 we can see that only two of them contain nouns in nominative. The nouns are: *utrata* and *leukocyty*. As a result only these two trees are considered to be the desired descriptions. The selected trees are then reduced to the flat sequences of words, which originally appeared in the text. As a result two descriptions are extracted from the sentence: *utrata masy ciała* and *podwyższone leukocyty we krwi*.

One element from the example sentence has not been discussed yet. This is the verb *są*. It cannot be associated to the other sentence elements using linguistic rules. It can,

however, be associated using sentence schemas, which allow to identify the sentence subject, predicate and object. We defined a set of the most typical sentence schemas to associate verbs to the rest of the sentences. The discussed sentence contains two subjects, which are the nouns *leukocyty* and *utrata*. The sentence schema detects only the first one, and associates it to the verb. As a result we get the association *leukocyty sq.* Unfortunately the schema assumes a noun in accusative to be the object of the activity expressed by the verb. No such noun could be found in the example sentence. As a result the construction retrieved by the schema is incomplete, and thus ignored.

Another thing which has not been considered yet are the ambiguities resulting from imprecision of the linguistic rules. The ambiguities lead to generating some additional word associations, which for conscious reader are obvious mistakes. Such mistakenly created associations could be the following: *objawami krwi*, *utrata krwi*, and *masy krwi*. This delivers some alternatives to descriptions, which could be extracted from text. If the ambiguities are not possible to be resolved, all the possible variants are generated, assuming that one of them is the correct one.

### III. COLLECTING CASES

The collection of descriptions extracted from text is of course far from perfect. It strongly depends on the actual contents of text corpus. It is obvious, that medical text contains not only symptom descriptions, but a lot of other information, including descriptions of medical procedures, patient treatment, etiology, or pathogenesis of diseases. All that information is unimportant for the diagnostic purposes. Unfortunately, the mechanism extracting information from text is based only on morpho-syntactic rules and is not able to interpret the meaning of extracted information. As a result the collected descriptions include except symptoms, also a lot of other unwanted information. Also some part of the descriptions is incorrect due to ambiguities which we were not able to resolve.

Fortunately the unwanted information is not so huge problem, as it could initially seem. The condition is an efficient search mechanism, which allows for quick finding of the desired description in the database. Given such mechanism, medical experts can quickly describe symptoms observed in patients. The most efficient search mechanism that we are able to deliver is based on suggestions to a typed sequence of characters. This mechanism is well known from the Google search web site. Using this mechanism the user is always able to find the desired description after typing an adequate number of characters. The search mechanism is additionally supported by weights assigned to the descriptions. The weights indicate the descriptions, which are frequently used, and should be moved to the front of the search list. Using the described tool the experts create a database of patient cases, which can be considered training patterns for the system.

Of course we are not able to guarantee that any possible symptom description, that an expert could ever think of,

is available in the collection extracted from text. Thus the description chosen by the user should be open for edition. In this way it is always possible to complete or correct the missing parts of the expression, or even build it from scratch. Every new description is then registered in the system and available for other users.

The training phase of the system is necessary for identifying the descriptions correctly describing symptoms. This is easily learned from the cases. The descriptions which were frequently used are considered to be correct. The descriptions which were not used, or used occasionally, are considered to be incorrect and removed from the system. It is assumed that some level of human errors is possible, and thus the rarely used descriptions are removed, as considered to be erroneous. In this way the system is resistible to occasional human mistakes. Of course we should distinguish the erroneous descriptions, from description of rare symptoms. The descriptions of rare, but important symptoms, are identified given the correlations with diseases. If a rare description is strongly correlated with some disease, it should not be eliminated.

## IV. BUILDING THE SEMANTIC MODEL OF SYMPTOMS

### A. Identification of Concepts

For building any semantic model it is necessary to identify the set of concepts. In our case the concepts of the model are the symptoms and the diseases. The descriptions remaining in the system after collecting a reasonable number of cases, although correct, are not symptoms yet. The reason is that some of them have the same or similar meaning. The meaning of a given description is considered to be a symptom. To discover the symptoms among the set of collected descriptions, it should be noted, that the descriptions with close meaning, have similar statistical distribution with respect to the set of diseases. This distribution is easily retrievable from the set of training cases. For practical reasons the set of diseases, which can be diagnosed is fixed, and *a priori* defined. The considered system has a modular structure, and a single module contains diseases from one domain. Currently we are experimenting with two domains: allergology and pulmonology. As a consequence we collect two sets of cases with diagnosed diseases from one of the two domains. The *a priori* defined set of diseases allows for ignoring the problem of synonymic names of diseases, and makes them immediately the set of the model concepts.

The problem remaining to be resolved is identification of symptoms. As already noted the synonymic descriptions have similar distributions of their occurrence in particular diseases registered in training cases. Identification of sets of such descriptions is possible through clusterisation. As a result of this process we get a set of symptoms, represented by identified clusters. It is of course possible that some descriptions, with different meaning, are closely correlated by occurring in the same diseases. Such descriptions are easily distinguished from synonyms. This is possible by observing, that synonymic descriptions are not used in the same cases, as no one describes the same symptom twice.

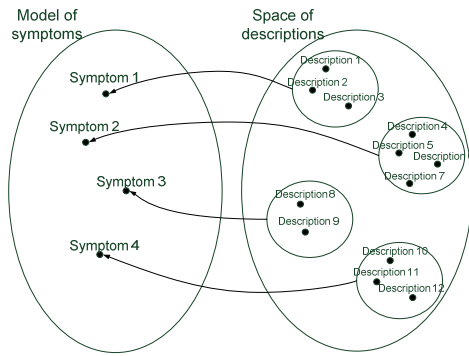


Fig. 2. Schematic mapping between clusters in the space of natural language descriptions, and the model of symptoms

After clusterisation, every description is easily assignable to its respective symptom. It is thus easy to determine the statistical distribution of symptoms with respect to diseases. This is just a result of simple summation of the distributions obtained for the synonymic descriptions. Such distribution can be utilised for construction of a Bayesian network, which can further be used as a diagnostic decision support tool.

### B. Identification of Vertical Relations

The simple clusterisation leads to a model of flat list of symptoms, where no relations between them are taken into account. Such a model is good for building a Bayesian network, but is not the most accurate for more complicated purposes, like semantic reasoning. The basic element of every semantic model is a vertical hierarchy of concepts. Such a hierarchy arranges concepts in the form of a tree, where the concepts with wider meaning are parent nodes of the concepts with more narrow meaning. This type of relation seems to exist also between symptoms. For example the symptom *allergic reaction to the animal fur* could be considered a superclass of the more narrow symptom *allergic reaction to the dog's fur*. This indicates that hierarchic model of symptoms is more accurate than the flat one. The hierarchy of symptoms can be built by applying hierarchic clusterisation algorithm. The clusters obtained in this way are directly transformed into hierarchy of semantic classes.

One should be careful, however, when trying to interpret the meaning of the classes from different levels of the model hierarchy. It should be underlined that the meaning of particular descriptions is resolved not on the foundation of the linguistic interpretations, but on the foundation of statistical association to diseases. It might seem strange at first, as most of the approaches to building semantic models are founded on purely linguistic interpretation of meaning. However, the presented way of meaning determination is much better, if we have in mind, that the model is build for diagnostic purposes. The linguistic interpretation can sometimes be misleading, when one would try to associate it with possible diagnoses. There are many symptoms which could be interpreted as subsymptoms of other symptoms from the linguistic point of view, while at the same time they are associated to significantly

differing diseases. In other words the diagnostic hierarchy of classes does not need to agree with the linguistic hierarchy of classes. In the diagnostic hierarchy of classes the wider meaning refers to occurrence of a symptom in a wider set of diseases, while the subclasses are more specific in the sense that they are associated with a more narrow set of diseases. When considering the mentioned example of the allergic reaction symptoms, we actually do not know if the two symptoms are actually related diagnostically. This is suggested by the linguistic interpretation, because the *dog's fur* is a subclass of the *animal fur*. But cannot be sure if the *allergic reaction to the dog's fur* indicates a subset of the diseases indicated by the *allergic reaction to the animal fur*. As a result, we cannot determine the vertical relations from the purely linguistic interpretations.

The identification of symptoms (i.e. the model concepts) on the foundation of association to diseases also influences the contents of particular synsets (i.e. the sets of synonymic descriptions). The descriptions belonging to individual clusters do not need to be actual synonyms from the linguistic point of view. It is enough if they are used as synonyms from the diagnostic perspective, i.e. their occurrence indicates the same diseases.

To underline the importance of diagnostic meaning determination on the foundation of association to diseases, it should be noted that the structure of a semantic model based purely on a linguistic perspective interpretation is always arguable. This is a result of the fact, that in the standard approach such models are built by choosing one of the possible verbal descriptions of the world. No matter how objective the model constructor tries to be, the final result is always subjective, because he is forced to choose between one of alternatives. The model built according to our approach is objective, in the sense that no individual can directly influence its structure. The structure is an outcome of the cumulated knowledge of all the experts taking part in collecting patient cases. The experts also mutually verify themselves by using or ignoring particular descriptions.

This of course does not mean that models based on linguistic relations are wrong. Everything depends on the purpose of the model. If a system is aimed at doing some kind of linguistic analyses it should be constructed in this way. In our case the aim is doing patient diagnoses, and the system should be constructed in a way which maximizes the accuracy of results.

### C. Adding Restrictions to Semantic Relations for Semantic Reasoning

The foundation for knowledge representation in semantic networks is description logics. This allows for expressing all the dependencies between semantic classes in terms of logical expressions. The logical expressions are then used by semantic reasoning engines. Semantic knowledge representation is thus a powerful tool in decision support systems. Such a tool can also be used for diagnosing patients on the foundation of their symptoms.

The way of constructing the model structure has already been described. What is missing is the set of restrictions of particular semantic relations. Such restrictions are expressed with the aid of a set of logical quantifiers. What is necessary to identify the restrictions, is to find mutual dependencies between symptoms which can easily be transformed onto a set of logical expressions. Such expressions are then associated with particular relations, and in this way the model structure becomes complete and ready for performing semantic reasoning. The input for this process is the set of symptoms observed in a patient, while as a result we get the suggested diseases. Appropriately constructed reasoning could also indicate the missing symptoms, which would significantly improve the diagnosis. In this way the system is able to suggest the medical tests necessary to perform for improving diagnosis.

The construction of the logical restrictions can be determined on the foundation of statistical distribution of symptoms with respect to particular diseases. Again the patient cases are the key resource to identify the model construction elements. Given a set of cases with a particular disease diagnosed it is possible analyse mutual occurrence of the symptoms in particular cases. This data is transformable into a set of logical expressions, such as logical sum, product, or any other statistically relevant dependency. These are the restrictions we are searching for. Such an analysis should be performed for all the diseases from the domain of interest. As a result we get a powerful tool of semantic reasoning.

## V. RESULTS OF EXPERIMENTS

The experiments that we are able to describe at this stage of the work refer to extraction of descriptions from text corpus. The results of clustering and building the semantic model will be described later. Currently we are working on collecting appropriately large collection of patient cases.

The text corpus used for the experiments came from two domains of medicine: allergology and pulmonology. To be more precise the experiments were carried out separately on texts from the two domains. The size of the corpora is rather small. For allergology it is 95kB, and for pulmonology it is 265kB. The main text resources were [24] for allergology and [25] for pulmonology. We selected only the book chapters and paragraphs, which actually describe symptoms. Including any other fragments of texts would deteriorate the results. This results from the fact that the analyser is based only on the foundation of the grammatical construction of the sentences. It is not able to interpret the meaning of the analysed text. The grammatical structure of symptom descriptions is no different than grammatical structure of any other entity described in the text. As a result any text processed by the analyser delivers a set of descriptions, no matter if it refers to symptoms or not. The careful selection of texts is thus important, if we want to avoid getting too many useless descriptions.

As a result of text processing we got 1080 descriptions for allergology and 2810 descriptions for pulmonology. The difference in numbers is the obvious consequence of the

corpora sizes. The average number of words in every description was 5.4 for allergology, and 4.8 for pulmonology. The number of retrieved descriptions is not the only factor which is important. As already mentioned the analyser is not able to distinguish, whether the extracted descriptions refer symptoms or to anything else. It is thus important to assess the rate of the number of symptom descriptions to the number of other descriptions. This rate strongly depends on the specifics of the analysed text. In some texts the symptoms appear rather sparsely, while other are almost entirely devoted to describe symptoms. Thus the observed rate ranges from 10-20% up to 80-90%. The assessed overall rate is about 50%. This amount is huge enough to cover significant part of the possible verbal descriptions of symptoms from the given domain, and be the basis for describing patient cases. The missing element will be completed during collecting cases.

## VI. CONCLUSION

The paper describes a methodology of building a model of diagnostic knowledge. The idea of the system assumes two stages in the model creation process. The first of them is text analysis in order to extract verbal constructions describing symptoms. The second stage aims to collect patient cases and build the model of symptoms on the foundation of the patient cases. As a result of the whole we get a semantic model built of symptoms and diseases. What distinguishes this approach from other solutions typically applied in building semantic models, is that no direct manipulation of the model is required. The system structure is learned from examples. The key tool for extracting the model structure is clusterisation and statistical analysis of particular symptoms occurrence in the cases.

The model is designed for diagnostic purposes. In the simplest case the diagnostic process can be supported by the Bayesian network constructed on the foundation of the data collected during the model construction. The more advanced algorithms lead to construction of a semantic network with hierarchic structure of symptoms, and description logics rules. This allows for performing reasoning based on a semantic inference engine. It should be underlined, that the meaning of the particular concepts forming the model, is not determined on the foundation of their linguistic interpretation. The meaning is a result of associations between the symptoms and the diseases. Such solution is the required when diagnosing a patient is the task. The natural language descriptions are used for human communication only, while the computational model is subordinated the diagnostic purposes. Trying to interpret the model structure in terms of natural language associations could be even misleading, so no one is supposed to do it.

The method of meaning determination not only rises the quality of the model. It also simplifies the model construction process. This is due to eliminating the task of the direct model manipulation by experts. In this way no human needs to care about choosing the most accurate world description method. Carrying about synonyms is also not required. These tasks are completely automatized. The model constructed in the



described way is also resistant to the subjectivity. This is a problem which appears when a model is constructed by a human expert which has to choose among one of possible model structures. In our approach the knowledge is extracted from cases, which are entered by more than one person. In this way the model is a resultant of all the individual habits and knowledge represented by the human users.

The experiments described in the paper refer to extraction of textual descriptions from the text corpus. The lexical analyser is able to extract the required information from text. The criterion for assessing the quality of the set of extracted descriptions is the rate of descriptions actually describing symptoms, to the other descriptions. This parameter does not depend only on the analyser, but also on the quality of the text. By quality of the text we mean the density of symptom descriptions which appear in the text. This factor depends on the authors' writing style. The density of symptom descriptions is important, because the analyzer works only on the foundation of grammatical rules of sentence construction. It is not able to interpret the meaning of the extracted descriptions. As the grammatical construction of a symptom description is no different than description of any other entity, we are not able to filter the undesired descriptions. Such a mechanism would be very helpful, but currently we are not able to deliver it.

The undesired descriptions are, however, not an obstacle which would make impossible delivering the user interface with a collection of symptom descriptions. This interface is necessary for describing the patient cases. The extracted set of descriptions is huge enough, to cover significant range of possible symptom descriptions. The key which allows for efficient entering of cases is an efficient search mechanism and system of weights. Of course we are not able to guarantee that all the required descriptions are extracted from text, thus during collecting the training cases the descriptions are open for edition. In this way it is possible to correct existing descriptions, or create new descriptions from scratch. Such descriptions are immediately available to other users entering the cases. This allows for mutual verification within the team responsible for collecting the cases.

The work on collecting a reasonable number of cases is in progress, so the effects of the second stage of building the model were not described here. This will be a field of experiments with clusterisation, and statistical analysis of cases in order to build the final model, which further will be used as a part of a decision support system.

#### ACKNOWLEDGMENT

This work was financially supported by the European Union from the European Regional Development Fund under the Operational Programme Innovative Economy (Project no. POIG.02.03.03-00-013/08).

#### REFERENCES

- [1] P. Velardi, R. Navigli, A. Cucchiarelli, F. Neri. "Evaluation of ontolearn, a methodology for automatic learning of ontologies," in *Ontology Learning from Text: Methods, Evaluation and Applications*, P. Buitelaar, P. Cimiano, B. Magnini, Eds. IOS Press, 2005, pp. 92-106.
- [2] A. Maedche, S. Staab. "Ontology learning for the Semantic Web." *Intelligent Systems*, vol. 16, pp. 72-79, Mar. 2001.
- [3] "The OntoGen system web site." Internet: <http://ontology-learning.net/wiki/OntoGen>, [Jun. 28, 2011].
- [4] D. Faure, T. Poibeau. "First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX," in *Proc. Ontology Learning ECAI-2000 Workshop*, 2000, pp. 7-12.
- [5] F. Pereira, A. Cardoso. "Clouds: A Module for Automatic Learning of Concept Maps," in *Lecture Notes in Computer Science*, vol. 1889/2000, pp. 468-470, 2000.
- [6] U. Hahn, M. Romacker. "The syndikate text knowledge base generator," in *Proc. of the 1st International Conference on Human Language Technology Research*, San Diego, 2001, pp. 328-333.
- [7] N. Weber, P. Buitelaar. "Web-based ontology learning with ISOLDE," in *Proc. of the ISWC Workshop on Web Content Mining with Human Language Technologies*, Athens, 2006.
- [8] L. Karoui, M.A. Aufaure, N. Bennacer. "Context-based Hierarchical Clustering for the Ontology Learning," in *Proc. of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, Hong Kong, 2006, pp. 420-427.
- [9] S. Sung, S. Chung, and D. McLeod. "Efficient concept clustering for ontology learning using an event life cycle on the web," in *Proc. of the 2008 ACM symposium on Applied computing*, 2008, New York, pp. 2310-2314.
- [10] S. Kok, P. Domingos. "Extracting Semantic Networks from Text Via Relational Clustering," in *Proc. of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer-Verlag Berlin, Heidelberg, 2008, pp. 624-639.
- [11] P. Buitelaar, P. Cimiano. *Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, Amsterdam: IOS Press, 2008.
- [12] W. Wong. "Learning Lightweight Ontologies from Text across Different Domains using the Web as Background Knowledge." Doctor of Philosophy thesis, University of Western Australia, Crawley, 2009.
- [13] A. Gomez-Perez, D. Manzano-Macho. "Deliverable 1.5: A survey of ontology learning methods and techniques." OntoWeb Consortium, Internet: [http://www.csd.uoc.gr/hy566/A\\_survey\\_of\\_ontology\\_learning\\_methods\\_and\\_techniques.pdf](http://www.csd.uoc.gr/hy566/A_survey_of_ontology_learning_methods_and_techniques.pdf), [Jul. 30, 2011]
- [14] M. Shamsfard, A. Barforoush. "The state of the art in ontology learning: A framework for comparison." *Knowledge Engineering Review*, vol. 18, pp. 293-316, Dec. 2003.
- [15] Y. Ding, S. Foo. "Ontology research and development: Part 1 - a review of ontology generation." *Journal of Information Science*, vol. 28, pp. 123-136, Apr. 2002.
- [16] K. Toutanova and C.D. Manning. "Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger." in *Proc. of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 2000, pp. 63-70.
- [17] A. Ratnaparkhi. "A Maximum Entropy Part-of-Speech Tagger." in *Proc. of the First Empirical Methods in Natural Language Processing Conference*, 1996, pp. 250-255.
- [18] M. Piasecki. "Polish Tagger TaKIPI: Rule Based Construction and Optimisation." *Task Quarterly*, vol. 11(1-2), pp. 151-167, 2007.
- [19] J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kubler, S. Marinov and E. Marsi. "MaltParser: A language-independent system for data-driven dependency parsing." *Natural Language Engineering*, vol. 13(2), pp. 95-135. Jun. 2007.
- [20] S. Szpakowicz. "Formalny opis składniowy zdań polskich." Warsaw: Warsaw University Publishing House, 1983. (in Polish)
- [21] M. Woliński. "Morfeusz - a Practical Tool for the Morphological Analysis of Polish Intelligent Information Processing and Web Mining." *Advances in Soft Computing*, vol. 35, pp. 511-520, Jun. 2006.
- [22] A. Przepiórkowski. "The IPI PAN Corpus. Preliminary Version." Warsaw: Institute of Computer Science PAS, 2004.
- [23] M. Woliński. "System znaczników syntaktycznych w korpusie IPI PAN." *Polonica*, vol. XXII/XXIII, pp. 39-55, 2003. (in Polish)
- [24] W.H.C. Burgdorf, G. Plewig, H.H. Wolff, M. Landthaler, "DERMATOLOGIA Braun-Falco," Lublin: Czelej, 2010. (in Polish)
- [25] A. Szczeklik. *Choroby wewnętrzne*. Kraków: Medycyna praktyczna, 2006. (in Polish)



# Query Expansion: Term Selection using the EWC Semantic Relatedness Measure

Vitaly Klyuev  
University of Aizu  
Aizu-Wakamatsu  
Fukushima-ken 965-8580, Japan  
Email: vklyuev@u-aizu.ac.jp

Yannis Haralambous  
Institut Télécom – Télécom Bretagne  
Dép. Informatique, UMR CNRS 3192 Lab-STICC  
Technopôle Brest Iroise, CS 83818  
29238 Brest Cedex 3, France  
Email: yannis.haralambous@telecom-bretagne.eu

**Abstract**—This paper investigates the efficiency of the EWC semantic relatedness measure in an ad-hoc retrieval task. This measure combines the Wikipedia-based Explicit Semantic Analysis measure, the WordNet path measure and the mixed collocation index. In the experiments, the open source search engine Terrier was utilised as a tool to index and retrieve data. The proposed technique was tested on the NTCIR data collection. The experiments demonstrated promising results.

## I. INTRODUCTION

A BAG-of-words representation of documents by information retrieval systems results in queries expressed utilising the language of keywords. Users face a vocabulary problem: A keyword language is not adequate to describe the information needs. Statistical analysis of user behaviors showed that the queries submitted to search engines are short (2 to 3 terms, on average) and ambiguous, and users rarely look beyond the first 10 to 20 links retrieved [20].

To improve user queries, search engines provide many tools.

Manuals and instructions are among them. They help a little, although ordinary users do not like reading.

A query suggestion feature is common for general purpose search engines. Its disadvantage is in the way to generate recommended terms. They are based on the first term typed by the user, which is always the first one in all expanded queries [22].

The relevance feedback feature is another instrument to help users. There are at least two steps in the interaction with a search engine: The first one is the submission of the original query, and the second one is the user reaction on the results retrieved in order to provide the system with the user opinion (marking some documents as relevant). This feature is not popular among users because the mechanisms of changing queries are not clear and users cannot control the process [9].

In addition, query suggestion and relevance feedback are used to modify the queries in order to make them more accurate to express the user information needs.

Query expansion is a well-known and popular technique to reformulate the user query in order to reduce the number of non-relevant pages retrieved by information retrieval systems. Another goal of query expansion is to provide the user with additional relevant documents. Automatic query expansion is an important area of information retrieval: Many sci-

entists are involved in designing new methods, techniques, and approaches.

This paper presents authors' technique to automatically expand the user queries. This technique is based on the EWC semantic relatedness measure [21]. This measure takes into account encyclopedic, ontological, and collocational knowledge about terms. The environment for the experiments includes Terrier as a search engine and NTCIR-1 CLIR data collection for the Japanese-English cross-lingual retrieval task.

The rest of the paper is organised as follows. The next section reviews the approaches to automatic query expansion. Section 3 describe the nature of the measure used. Section 4 provides the necessary details related to this technique to expand the queries. The tools and data utilised in the experiments are presented in Section 5. The results of the experiments are discussed in Section 6 and comments on ongoing experiments are presented in Section 7. Concluding remarks are presented in Section 8.

## II. RELATED WORK

A comprehensive review of the classical approaches to expand queries can be found in [9]. They propose different ways to obtain semantically (topically) related terms, techniques to evaluate importance of the terms found, mechanisms to define the number of terms to add (expand) the user query, and strategies to evaluate the quality of obtained results.

Generally, the semantics of the terms are clear when they are in the sentences because the meanings of the words are fixed and only one is usually selected from the set of all possible variants. However, in the queries, the terms are separate instances, and their semantics are unclear. This is called the polysemy problem. On the other hand, the same things can be described by using different terms. This is the nature of the synonymy problem. The query should be rich enough to include the possible candidates for expressing user information needs.

The general goal of query expansion is to find a solution for these two aforementioned problems. The classical solution for the synonymy problem is to apply thesauri as instruments to obtain the candidates for expansion. WordNet is widely used for this purpose [19]. Modern techniques suggest Wikipedia as a valuable source to find synonyms [12].

Many techniques are used to solve the polysemy problem. Approaches described in [13] and [16] are based on the analysis of the query log files of search engines and clicked URLs. Authors of this study [18] utilised WordNet for a deep analysis of the queries submitted to the information retrieval system in order to find the concepts and then obtain the candidate terms for expansion. The involvement of users is the feature of the approach discussed in [17]. They should select the correct ontology for each query submitted to expand the query. The authors of this study [14] also pointed out that the information exploited by different approaches differs, and combining the different query expansion approaches is more efficient than the use of any of them separately. They investigated techniques to rank the terms extracted from the retrieved documents. One is based on the measures of occurrence of the candidate and query terms in the retrieved documents. The other one utilises the differences between the probability distribution of terms in the collection and in the top ranked documents retrieved by the system. A similar idea is discussed in [15].

The authors of [10] combined the concept-based retrieval, based on explicit semantic analysis (ESA), with keyword-based retrieval. At the first step, they use keyword-based retrieval to obtain the candidates for query expansion. Then, they tune queries applying ESA. After that, they perform the final retrieval in the space of concepts.

It is difficult to compare the aforementioned approaches, because different data sets were used to evaluate them. In many cases, it is not clear wherever the test queries cover a wide range of data set topics. The performance evaluation is done automatically for some approaches, whereas for others, the authors involve the users to judge the quality of retrieval.

### III. MEASURE DESCRIPTION

In study [21], the new measure of words relatedness is introduced. It combines the ESA measure  $\mu_{ESA}$  [10], the ontological WordNet path measure  $\mu_{WNP}$ , and the collocation index  $C_{\xi}$ . This measure is called EWC (ESA plus WordNet, plus collocations) and is defined as follows:

$$\begin{aligned}\mu_{EWC}(w_1, w_2) &= \mu_{ESA}(w_1, w_2) \cdot \alpha \\ \alpha &= (1 + \lambda_{\sigma}(\mu_{WNP}(w_1, w_2))) \cdot \gamma \\ \gamma &= (1 + \lambda'_{\sigma}(C_{\xi}(w_1, w_2)))\end{aligned}$$

where  $\lambda_{\sigma}$  weights the WordNet path measure (WNP) with respect to ESA, and  $\lambda'_{\sigma}$  weights the mixed collocation index with respect to ESA. This index is defined as follows:

$$C_{\xi} = \frac{2 \cdot f(w_1 w_2)}{f(w_1) + f(w_2)} + \xi \frac{2 \cdot f(w_2 w_1)}{f(w_1) + f(w_2)}$$

where  $f(w_1, w_2)$ ,  $f(w_2, w_1)$  are the frequency of the collocations of  $w_1 w_2$  and  $w_2 w_1$  in the corpus,

and  $f(w_i)$  is the frequency of word  $w_i$ . The values for constants  $\lambda_{\sigma}$ ,  $\lambda'_{\sigma}$ , and  $\xi$  were set to 5.16, 48.7, and 0.55, respectively.

Study [21] demonstrated the superiority of this measure over ESA on the WS-353 test set.

### IV. EXPANSION METHOD

Assume that  $Z$  is a pool of term-candidates for query expansion. The formulas below present the method to select terms to expand queries.  $N$  is a number of original query terms, and  $j$  is an index of them. Values for the WordNet component and collocation component should be above zero in order to choose related terms. Thresholds  $t_2$  for EWC values and  $t_1$  are parameters adjusted in the experiments. For every word  $w_i \in Z$ , the weight is calculated. Word  $w_i$  is selected for expansion if its weight is equal to 1.

$$\begin{aligned}\text{weight}(w_i) &= \begin{cases} 1, & \text{if } \sum_{j=0}^N \frac{\text{score}(w_i, w_j)}{N} > t_1, \\ 0, & \text{otherwise} \end{cases} \\ \text{score}(w_i, w_j) &= \begin{cases} 1, & \text{if } \mu_{WNP} > 0; C_{\xi} > 0; \mu_{EWC} > t_2, \\ 0, & \text{otherwise} \end{cases}\end{aligned}$$

This approach can be interpreted as follows: A term is selected from the list of term-candidates, if the similarity score between this term and the majority of original query terms is higher than a given threshold  $t_1$ . The term-candidate should have non-zero values for  $\mu_{WNP}$  and  $C_{\xi}$  components.

### V. TOOLS AND DATA SETS USED

The open source search engine Terrier [1] was used as a tool to index and retrieve data. It provides the different retrieval approaches. TF-IDF and Okapi's B25 schemas [6, 9] are among them. As a data set for experiments, the NTCIR CLIR data collection [2] was used. It consists of 187,000 articles in English. These articles are summaries of papers presented at scientific conferences hosted by Japanese academic societies. The collection covers a variety of topics such as chemistry, electrical engineering, computer science, linguistics, and library science. The size of the collection is approximately 275.5 MB. A total of 83 topics are in Japanese. A structure of the dataset and topics is similar to that of TREC [3]. A Porter Stemmer algorithm was applied to the documents and queries, and a standard stop word list provided by Terrier was also utilised. Only the title fields were considered as a source of the queries. They are relatively short, each query consists of a few keywords. The authors of the study reported in [5] experimented with Terrier applying the

same conditions to the TREC data. To measure the term similarities, an experimental tool described in [21] was utilised.

## VI. RESULTS OF EXPERIMENTS

The authors implemented the proposed technique to expand queries as follows.

A straightforward approach was applied to translate queries into English: Google's translation service [4] generated queries in English. This method was selected because online dictionaries do not work well with terms in katakana and specific terminology [7]. Katakana is one of four sets of characters used in Japanese writing. It is primarily applied for the transcription of foreign language words into Japanese.

To obtain the candidates for query expansion, a query expansion functionality offered by Terrier was adopted. It extracts the most informative terms (in this case 10) of the top-ranked documents (in this case 3) by using a particular DFR (divergence from randomness) term weighting model [8].

Table 1 provides the list of original queries (topics 1, 12, and 24), terms-candidates for expansion (arranged by the decreasing score calculated by Terrier), and the final sets of terms used to expand queries (they are in bold). One to five terms were selected by this method. As one can see from this table, this technique does not usually select the top-ranked terms as candidates for expansion from the Terrier engine point of view.

As mentioned in Section V, a total of 83 topics are available to retrieve documents from the collection. The original goal of topics 0001 to 0030 is to tune the parameters of the retrieval system. Their relevance judgments are known in advance. Topics 0031 to 0083 were used in official runs at the NTCIR 1 Workshop. Organisers found that the number of relevant documents for 13 topics of the 53 contained less than five relevant documents per topic in cross-lingual retrieval. Hence, they discarded these topics from evaluation [25]. The full set was used in these experiments because the goal is to compare the performance of different methods implemented in the same environment. In the evaluations, the partially relevant documents were considered as irrelevant. To archive this, the corresponding file was applied when evaluating the retrieval results.

Table 2 summarises the results of retrieval to tune thresholds  $t_1$  and  $t_2$ . The test queries were generated from topics 0001 to 0030. It is important to note that when the queries are expanded with all the terms proposed by Terrier, the retrieval results drop to zero. The retrieval utilising the TF-IDF schema produced better results for the original queries (without expansion) compared to the BM25 and InL2 models [1]. The line *system* shows this result. The first number in the cells is the value of average precision, and the second one is the value of R-precision. The performance of retrieval with expanded queries utilising the ESA and EWC approaches for the threshold values ( $t_1$  equals 0.67 and

$t_2$  is ranges from 0.08 to 0.15) is better compared to the variant without expansion. For the EWC measure, the maximum of the retrieval performance is reached when the values

of thresholds  $t_1$  and  $t_2$  are set to 0.67 and 0.12. For ESA, the optimal threshold values are 0.67 and 0.08. The performance of ESA is higher than EWC.

Table 3 summarises the results of retrieval for topics 31 to 83. The threshold values were set to the optimal parameters (see Table 2). Six runs were executed. Figure 1 shows an averaged precision/recall graph across 53 queries. The EWC measure demonstrated better performance over ESA in both cases. The line *system* shows the retrieval results without expansion for TF-IDF and BM25 schemes.

To summarize, one can conclude that the EWC measure provides little benefit over ESA, as the results of the retrieval are better. Ontological knowledge combined with collocational knowledge helps in the selection of expansion terms.

TABLE I.  
TOPICS: 1, 12 AND 24: ORIGINAL AND EXPANDED QUERIES

Topic	Original query	Terms for expansion
1	Robot	<b>Robot</b> person <b>human</b> multi comput sice design will confer paper
12	Mining methods	Mine method rule data databas associ discoveri <b>larg</b> tadashi solv amount
24	Machine translation system	Machin translat <b>system</b> exampl <b>base</b> <b>masahiro</b> <b>method</b> nation convert <b>problem</b>

TABLE II.  
THRESHOLDS TUNING: TOPICS 1 TO 30

t1	t2	EWC: Average Precision R-Precision			ESA/ BM25
		InL2	TF-IDF	BM25	
0,5	0,1	0.2940	0.3031	0.3072	<b>0.3101</b>
		0.3216	0.3314	0.3324	<b>0.3347</b>
0,65	0,09	0.2940	0.2936	0.2955	<b>0.2961</b>
		0.3300	<b>0.3332</b>	0.3278	0.3276
0,67	0,07	<b>0.2973</b>	0.2954	0.2960	0.2959
		<b>0.2977</b>	0.2963	0.2916	0.2916
0,67	0,08	0.3101	0.3151	0.3140	<b>0.3172</b>
		0.3277	0.3268	0.3265	<b>0.3315</b>
0,67	0,09	0.3073	0.3105	<b>0.3106</b>	0.3080
		0.3256	<b>0.3373</b>	0.3352	0.3373
0,67	0,1	0.3030	<b>0.3103</b>	0.3099	0.3099
		0.3295	<b>0.3350</b>	0.3349	0.3318
0,67	0,11	0.3049	<b>0.3121</b>	0.3110	0.3102
		0.3239	<b>0.3309</b>	0.3292	0.3302
0,67	0,12	0.3049	<b>0.3121</b>	0.3110	0.3092
		0.3239	<b>0.3309</b>	0.3292	0.3284
0,67	0,13	0.3049	<b>0.3114</b>	0.3099	0.3111
		0.3125	0.3245	0.3248	<b>0.3282</b>
0,67	0,15	0.3033	<b>0.3115</b>	0.3111	0.3110
		0.3143	0.3267	<b>0.3282</b>	0.3282
0,69	0,09	0.3073	0.3105	<b>0.3106</b>	0.3080
		0.3256	<b>0.3373</b>	0.3352	0.3373
0,75	0,1	0.3030	0.3103	<b>0.3099</b>	0.3099
		0.3295	0.3330	<b>0.3349</b>	0.3318
System	system	0.2980	0.3034	0.3017	
		0.2995	0.3166	0.3163	

## VII. ONGOING EXPERIMENTS

The authors are conducting experiments on the translation of queries from Japanese into English. The Mecab system [24] is utilised to segment the queries and extract Japanese terms. An on-line dictionary Space ALC [23] was also applied to retrieve all English variants for the corresponding Japanese terms. The EWC metrics are applied to pairs in order to select the most closely related elements from the sets of the meanings (terms with the highest sum of weights between pairs). For the queries consisting of only one term, the most common English variant is selected. Queries generated in this way will be submitted to the search engine, and comparison will be done with the results of retrieval for queries obtained from Google's translation service.

## VIII. CONCLUSION

This study tested the semantic relatedness measure when selecting the terms to expand queries. Key components of this measure are the ESA measure, the WordNet path measure, and the mixed collocation index. Results produced by the Terrier search engine were a base line in the experiments.

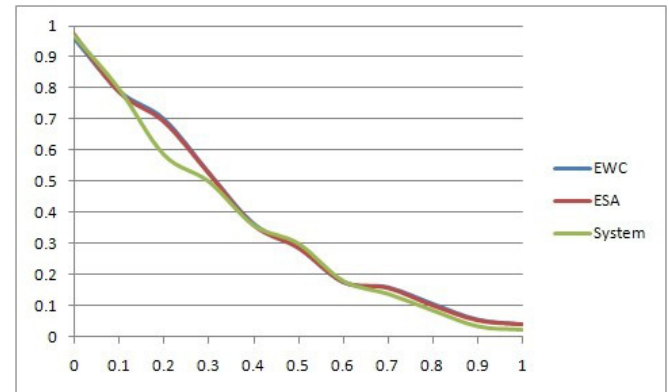


Fig 1. Averaged 11-point precision-recall graph across topics 31 to 83

TABLE III.  
RETRIEVAL RESULTS: TOPICS 31 TO 83

t1	t2	EWC: Average Precision R-Precision		ESA: Average Precision R-Precision	
		BM25	TF-IDF	BM25	TF-IDF
0,67	0,08	<b>0.2363</b>		0.2349	
		<b>0.2416</b>		0.2390	
0,67	0,12		0.2200		0.2161
			0.2317		0.2284
System	system	0.2225	<b>0.2218</b>		
		0.2350	<b>0.2359</b>		

Term candidates for the expansion were also generated by Terrier. The proposed techniques were applied to the ad-hoc retrieval task. As a data set, the NTCIR-1 CLIR Test collection was used. The initial English queries were obtained automatically applying Google translate. The queries were expanded by applying the Wikipedia-based Explicit Semantic Analysis measure, and the DFR mechanism, and the semantic relatedness measure. The retrieval results showed superiority of the last one over ESA and DFR.

## REFERENCES

- [1] Terrier. [On line document], <http://terrier.net>
- [2] NTCIR-1 CLIR data collection. [On line document], <http://research.nii.ac.jp/ntcir/data/data-en.html>
- [3] TREC. [On line document], <http://trec.nist.gov/>
- [4] Google Translate, <http://translate.google.com/>
- [5] Ben He and Iadh Ounis. "Studying Query Expansion Effectiveness," in Proc. *The 31st European Conference on Information Retrieval (ECIR09)*. Toulouse, France, 2009.
- [6] S. E. Robertson, S. Walker, M. M. Beaulieu, M. Gatford, and A. Payne, "Okapi at TREC-4," in Proc. *TREC 4*, 1995.

- [7] Aitao Chen, Fredric C. Gey, Kazuaki Kishida, Hailing Jiang and Qun Liang, "Comparing Multiple Methods for Japanese and Japanese-English Text Retrieval, NTCIR Workshop 1," in Proc. *The First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, August 30 - September 1, 1999.
- [8] G. Amati and C.J. Van Rijsbergen, "Probabilistic models of information retrieval based on measuring the divergence from randomness," *ACM Transactions on Information Systems (TOIS)*, 20(4):357-389, 2002.
- [9] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008.
- [10] Ofer Egozi, Shaul Markovitch, and Evgeniy Gabrilovich, "Concept-Based Information Retrieval using Explicit Semantic Analysis," *ACM Transactions on Information Systems*, 29(2), 2011.
- [11] Philipp Sorg, Philipp Cimiano, "An Experimental Comparison of Explicit Semantic Analysis Implementations for Cross-Language Retrieval," in Proc. *The International Conference on Applications of Natural Language to Information Systems (NLDB)*, Saarbrücken, June 2009.
- [12] Yinghao Li, Wing Pong Robert Luk, Kei Shiu Edward Ho, and Fu Lai Korris Chung, "Improving weak ad-hoc queries using Wikipedia as external corpus," in Proc. *The 30th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM New York, NY, USA, 2007, pp 797 - 798.
- [13] Hamada M.Zahera, Gamal F. El Hady, and Waiel.F Abd El-Wahed, "Query Recommendation for Improving Search Engine Results," in Proc. *The World Congress on Engineering and Computer Science 2010 Vol I, WCECS 2010*, October 20-22, 2010, San Francisco, USA.
- [14] Jose R. Perez-Agueral and Lourdes Araujo, "Comparing and Combining Methods for Automatic Query Expansion," *Advances in Natural Language Processing and Applications Research in Computing Science* 33, 2008, pp. 177-188.
- [15] Ming-hung Hsu, Ming-feng Tsai, and Hsin-hsi Chen, "Combining WordNet and ConceptNet for Automatic Query Expansion: A Learning Approach," in Proc. *Asia Information Retrieval Symposium*, 2008, pp. 213-224.
- [16] Hang Cui1, Ji-Rong Wen, Jian-Yun Nie3, Wei-Ying Ma, "Probabilistic query expansion using query logs," in Proc. *The 11th international conference on World Wide Web*, ACM New York, NY, USA, 2002.
- [17] J. Malecka, and V. Rozinajova, "An Approach to Semantic Query Expansion," in Proc. *Tools for Acquisition, Organization and Presenting of Information and Knowledge, Research Project Workshop*, Bystra dolina, Tatry (2006), pp. 148-153.
- [18] Jiuling Zhang, Beixing Deng, and Xing Li, "Concept Based Query Expansion Using WordNet," in Proc. *the 2009 International e-Conference on Advanced Science and Technology*, 2009, pp. 52-55.
- [19] M. Ellen Voorhees, "Query expansion using lexical-semantic relations," in Proc. *The 17th Annual International ACM SIGIR conference on Research and Development in Information Retrieval*, 1994.
- [20] D. Gayo-Avello Brenes, "Stratifies analysis of AOL query log," *Information Sciences*, 179 (2009), pp. 1844-1858.
- [21] Yannis Haralambous and Vitaly Klyuev, "A Semantic Relatedness Measure Based on Combined Encyclopedic, Ontological and Collocational Knowledge," in the Proc. Of *IJCNLP 2011*, forthcoming publication.
- [22] Vitaly Klyuev, Ai Yokoyama, "Web Query Expansion: A Strategy Utilizing Japanese WordNet," *Journal of Convergence*, V. 1, Number 1, 2010.
- [23] Space ALC. [On line document], <http://www.alc.co.jp/>
- [24] Mecab. [On line document], <http://mecab.sourceforge.net/>
- [25] Noriko Kando, Kazuko Kuriyama, Toshihiko Nozue, Koji Eguchi, Hiroyuki Kato and Soichiro Hidaka, "Overview of IR tasks", in Proc. *The First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, August 30 - September 1, 1999.



# LTIMEX: Representing the Local Semantics of Temporal Expressions

Paweł Mazur<sup>1,2</sup>Robert Dale<sup>2</sup>

<sup>1</sup>Institute of Applied Informatics,  
Wrocław University of Technology  
Wyb. Wyspiańskiego 27,  
50-370 Wrocław, Poland  
Email: pawel@mazur.wroclaw.pl

<sup>2</sup>Centre for Language Technology  
Macquarie University  
NSW 2109, Sydney, Australia  
Email: Robert.Dale@mq.edu.au  
Email: Pawel.Mazur@mq.edu.au

**Abstract**—Semantic information retrieval requires that we have a means of capturing the semantics of documents; and a potentially useful feature of the semantics of many documents is the temporal information they contain. In particular, the *temporal expressions* contained in documents provide important information about the time course of the events those documents describe. Unfortunately, temporal expressions are often context-dependent, requiring the application of information about the context in order to work out their true values. We describe a representational formalism for temporal information that captures what we call the *local semantics* of such expressions; this permits a modularity whereby the context-independent contribution to meaning can be computed independently of the global context of interpretation, which may not be immediately or easily available. Our representation, LTIMEX, is intended as an extension to widely-used TIMEX2 and TimeML representations.

## I. INTRODUCTION

**A**N IMPORTANT part of the meaning of many documents is the temporal information they contain. In particular, many documents present narratives over sequences of events, and specifications of dates and times in the form of *temporal expressions* provide timestamps for these events; these are of significant utility for any application that aims to mine information about events across a large document set. From the information retrieval perspective, time is an important notion that can be used for indexing, organizing, retrieving and, finally, presenting the content of documents; these issues have been the topic of recent studies (see, for example, [1]).

Unfortunately, not all such temporal information is expressed in easy-to-capture and easy-to-interpret expressions. While fully-specified expressions, such as dates like *25th November 2010*, are not uncommon in text, far more common are context-dependent temporal expressions, like *yesterday*, *10th June*, and *the previous summer*. Properly assigning values to such expressions is the aim of *temporal expression tagging*; the TIMEX2 standard (introduced as annotation guidelines in [2]) and TimeML (see [3], [4]) have been developed as forms of representation for such values, and a considerable body of research focuses on developing tools that can accurately annotate such expressions with their absolute values (see, for example, [5]–[10]).

The assumption underlying the processing carried out by existing tools is that there are two distinct stages involved: first the *extent* of the temporal expression must be determined—in other words, it must be properly recognised—and then its value may be computed, taking into account both the lexical content of the expression and the wider context in which it is situated.

This second step conflates knowledge from two sources: information that can be derived from the content of the temporal expression itself, and arbitrary real-world reasoning that takes account of contextual information. As a simple example, the meaning of *yesterday* depends crucially on the time at which is uttered; however, common to all instances of its use is that it means ‘the day before today’. Based on this observation, we can compute the partial meaning of such expressions, thus permitting a modularisation of the process into a part that computes the context-independent *local semantics* of the expression, and a part which uses the wider context to determine the *global semantics*. The first of these can proceed in the absence of the second. However, we then require an interface between the two levels of processing; for this we propose what we call LTIMEX, an extension of the TIMEX2 and TimeML representations that allows us to capture partial meanings.

In this paper, which further develops our earlier work presented in [11], we describe the LTIMEX representation in some detail. Section II explains the relationship between LTIMEX and the existing TIMEX2 and TimeML standards; Section III describes in detail how we represent a wide variety of categories of temporal expressions using LTIMEX; and Section IV draws some conclusions about the utility of the scheme.

## II. BACKGROUND: THE TIMEX2 REPRESENTATION

TIMEX2, developed by the information extraction community, is a widely-used annotation standard for temporal expressions in text; it serves as a target representation for temporal expression taggers. For the purpose of its use in this standard, a temporal expression is defined as a linguistic expression which

TABLE I  
ENCODING OF POINTS AND PERIODS IN TIMEX2.

Attribute Value	Meaning
1	The second millennium AD
19	The 20th century AD
199	The 1990s
1992	Year 1992
1992-06-27	27 June 1992
1992-06-27T18:04	27 June 1992 18:04
1992-06-27T18:04:56	1992-06-27 18:04:56
1992-06-27TMO	morning of 27 June 1992
1992-W04	The fourth week of 1992
1992-SU	Summer of 1992
1992-H1	1st half of 1992
1992-Q3	3rd quarter of 1992
BC0346	Year 346 BCE
MA6	6 million years ago
PAST_REF	vague reference to past
T19:00	7pm (used in a non-specific context)
XXXX-06-XX	a day in June (non-specific context)
P2Y3M	2 years and 3 months
P2DT6H	2 days and 6 hours
PT3.5H	3.5 hours
P2DE	2 decades

references a *point in time* (such as a calendar date or time of day) or a *period* (also called a *duration*).<sup>1</sup>

TIMEX2 defines five attributes to represent the meaning of a temporal expression: `VAL`, `ANCHOR_VAL`, `ANCHOR_DIR`, `MOD`, `SET`. These attributes are used, respectively, to encode the temporal location of a point on a timeline or a duration of a period; to encode the temporal location of one of the period's end-points; to capture the direction of temporal reference from the anchoring point; to express modifications to more basic temporal values; and to flag whether the temporal expression refers to a set of temporal entities.

Values of the `VAL` and `ANCHOR_VAL` attributes use a string representation based on formats defined in the ISO-8601 standard: calendar date (`YYYY-MM-DD`), week date (`YYYY-Www-D`), time of day (`hh:mm:ss`), date and time (`YYYY-MM-DDThh:mm:ss`), and duration (`PnYnMnDTnHnMnS` or `PnW`). The individual character positions in the date and time strings correspond to particular granularities of temporal information, as demonstrated by the examples in Table I. TIMEX2 extends the encodings provided in the ISO standard by introducing tokens representing additional temporal granularities: for example, in place of a month number, TIMEX2 also permits codes for year seasons (e.g. `SU` for summer), half-years (e.g. `H1`), and quarter-years (e.g. `Q3`). It also adds support for BCE years, references to the distant past (i.e. billions, millions, and thousands of years ago) and general references to the past, present and future. For non-specific

<sup>1</sup>The distinction between a point and a period in TIMEX2 is different from the distinction often made in artificial intelligence and work on the philosophy of time. In contrast to usage in those areas, here a point is not a durationless instant, but a point on a timeline of some granularity. For example, a month is annotated as a point, not a period, when referenced as an element of a calendar (e.g. *He graduated in November*).

use of expressions (as in, for example, *a sunny day in June*) TIMEX2 uses an uppercase X to fill the slots at the unspecified granularities (for example, `XXXX-06-XX`). In regard to the encoding of duration, TIMEX2 adds new temporal units for decades, centuries and millennia.

In documents, a temporal expression tagger encodes these values as attributes of inline XML annotations, as in the following example:

(1) I left town on `<TIMEX2 VAL="2010-07-15">15th July 2010</TIMEX2>`.

What is notable about this temporal expression is that it is *fully-specified* or *explicit*: the temporal value can be computed using the lexical content of the string alone, without any reference to context.<sup>2</sup>

Not all temporal expressions are of this kind; rather, many are context-dependent, in that they only partially specify a temporal value, and require incorporation of information available in the context in order to derive their full interpretation. Unfortunately, the nature of the TIMEX2 representation means that there is no way of annotating the value of temporal expressions until this full interpretation has been carried out.

To address this problem, in this paper we describe an extension of TIMEX2 that uses a few simple notational devices to permit the representation of partially-interpreted temporal expressions in a way that is consistent with the original TIMEX2 specification. This has the benefit that it is easy to learn for those already familiar with TIMEX2 annotation; it can also take advantage of existing tools for evaluating the performance of temporal expression taggers. Most importantly, though, it provides us with an interface language that enables a modularisation of the process of temporal expression interpretation, so that we can use distinct components for determining the local semantic interpretation of a temporal expression and for the process of incorporating contextual information.

In TimeML, temporal expressions are annotated inline using TIMEX3 tags. TIMEX3 is a subsequent annotation standard to TIMEX2 which uses a different set of attributes. A detailed description is not appropriate here, but we note that its `value` attribute is used in exactly the same way as TIMEX2 uses the `VAL` attribute, which is the most important and relevant attribute for the development of LTIMEX. TIMEX3 represents the end-points of a period by means of additional TIMEX3 annotations, so its `value` attribute also replaces the `ANCHOR_VAL` attribute in TIMEX2. Thus, although LTIMEX was originally developed as an extension to TIMEX2, it is also compatible with TimeML and TIMEX3.

### III. REPRESENTING PARTIAL INTERPRETATIONS

TIMEX2 was designed to annotate temporal expressions with their global semantics, i.e. the temporal value obtained by interpreting the expression in the context of the content of the document in which it is used. Our experience with the

<sup>2</sup>This is not entirely true, since at least a particular calendar and a particular timezone are assumed, so values will always be relative to these. For most practical purposes, however, this makes the expressions context-free.



TABLE II  
A SUMMARY OF THE LTIMEX ATTRIBUTES.

Attribute	Comments
L-VAL	Encodes the local semantics of expressions concerning the temporal location of points: for underspecified values, the missing slots are filled with x; underspecified time is separated from the date components with t; for offsets, the encodings start with +, -, > or <; ordinally-specified elements are encoded with the pattern \$nu. Encoding of durations is the same as in TIMEX2.
L-ANCHOR_VAL	Local semantics of temporal location of end-points; see the description of L-VAL.
L-ANCHOR_DIR, L-MOD, L-SET	Same values as for the corresponding attributes in TIMEX2.
L-TYPE	Encodes the taxonomical type of an expression. The possible values are: EXPLICIT, UNDERSPECIFIED, OFFSET, OFFSET-DEICTIC, OFFSET-ANAPHORIC, EVENT-BASED, EVENT-BASED-POINT, EVENT-BASED-PERIOD, PERIOD, SET, SET-POINTS, SET-PERIODS.
L-EVENT_ID	Stores an identifier of an event for event-based expressions.
L-ANCHOR_TYPE	For anchored period expressions where the anchor is an offset, indicates the type of the offset; the possible values of the attribute are DEICTIC and ANAPHORIC.

development of a temporal expression tagger (presented in [9]) revealed that it is beneficial for both design and evaluation to explicitly recognize the semantics of the expression with no context involved; we refer to this as the *local semantics*, representing the partial and underspecified context-free meaning of the expression.

LTIMEX extends the set of attributes from TIMEX2 to provide a vocabulary for capturing partially-specified meaning. Some of these local attributes simply mirror the existing attributes; others, however, add new types of information that are intended to be of use to a subsequent processing stage that determines the global semantics of the temporal expression. The attributes provided by LTIMEX are shown in Table II.

A key feature here is the L-TYPE attribute, which stores the type of an expression; this captures the essential distinctions between different kinds of expressions, so that this information can be used to guide subsequent processing. In Table III we present the types of temporal expressions that we distinguish in this work, along with examples of each. These do not represent a flat taxonomy: the major types are expressions referring to single point and duration temporal entities, each of which may have subtypes; but there are also expressions referring to sets of such temporal entities, as well as ordinally-specified, modified and non-specific expressions.

Many of these types are introduced in TIMEX2 at the level of what we refer to as global semantics. The literature on temporal expression tagging identifies subtypes of point expressions that require different interpretation algorithms to derive their global semantics: *explicit*, *underspecified*, and *deictic* and *anaphoric offsets*.<sup>3</sup> We follow these taxonomic distinctions for the purpose of representing the local semantics. For the same reason, we also identify *ordinally-specified* expressions, allowing a further level of distinction for explicit, underspecified and offset expressions.

<sup>3</sup>The terminology used in literature in this regard varies; in this work we use what we believe are the most intuitive terms.

TABLE III  
THE TYPES OF TEMPORAL EXPRESSIONS.

Expression Type	Example Expression
Explicit Point	<i>Friday, 3 April 1998</i>
Underspecified Point	<i>23rd June</i>
Deictic Offset	<i>tomorrow</i>
Anaphoric Offset	<i>the next month</i>
Event-based Point	<i>the day when the last fortress fell</i>
Duration	<i>six months and two days</i>
Event-based Duration	<i>the first two minutes of the meeting</i>
Ordinally-specified	<i>the last Tuesday in 1997</i>
Modified Points	<i>the middle of August</i>
Modified Durations	<i>nearly two decades</i>
Non-specific Point	<i>a sunny day in July</i>
Set	<i>every Tuesday</i>

TABLE IV  
VALUES ASSIGNED TO EXPLICIT DATES AND TIMES IN TIMEX2.

No	Expression	Representation (VAL)
1	3rd January 1987	1987-01-03
2	Friday, 3 April 1998	1998-04-03
3	24/03/1980	1980-03-24
4	03/24/1980	1980-03-24
5	November 1996	1996-11
6	1960s	196
7	12th January 2001 11:59 pm	2001-01-12T23:59

Below, we present the LTIMEX scheme by discussing the representation of local semantics for a wide variety of types of temporal expressions that are found in real texts.

#### A. Explicit Expressions

These expressions are the only context-independent point expressions. For these, the local semantics is always the same as the global semantics, so our L-VAL simply mirrors the VAL in TIMEX2. We present some examples in Table IV.

TABLE V  
EXAMPLES OF UNDERSPECIFIED EXPRESSIONS IN LTIMEX.

No	Expression	Representation (L-VAL)
1	January 3	xxxx-01-03
2	the nineteenth	xxxx-xx-19
3	November	xxxx-11
4	summer	xxxx-SU
5	'63	xx63
6	the '60s	xx6
7	9 pm	xxxx-xx-xxT21
8	11:59 pm	xxxx-xx-xxT23:59
9	eleven in the morning	xxxx-xx-xxT11:00
10	ten minutes to 3	xxxx-xx-xxt02:50
11	15 minutes after the hour	xxxx-xx-xttx:15
12	Friday	xxxx-Wxx-5
13	8:00 p.m. Friday	xxxx-Wxx-5T20:00
14	eight o'clock Friday	xxxx-Wxx-5t08:00

### B. Underspecified Expressions

Underspecified expressions differ from explicit expressions in that they omit elements of information, which then have to be recovered from the context by some process of interpretation. LTIMEX provides for the representation of underspecified expressions by marking those elements of the temporal value that are missing with a special symbol; here we use a lowercase *x*, reminiscent of its common use as a variable.<sup>4</sup> Table V presents examples of a range of underspecified expressions along with their L-VAL attributes using this encoding.

For underspecified expressions referring to times that do not indicate the part of day (either 'am' or 'pm'), such as those in Rows 10, 11 and 14 of the table, we use a lowercase *t* separator (instead of the standard *T* separator) between the date and time parts of the representation. Together, these notational conventions indicate explicitly those parts of a temporal value that remain uninstantiated.

A few other elements of this representation are worthy of mention. The local semantics of bare weekday names, such as *Monday* or *Friday*, can not be represented in the standard month-based format *yyyy-mm-dd*, and therefore must be represented in the week-based format *yyyy-Wnn-d*, where *nn* is the ISO week number and *d* is the number of the weekday within that week (1 denotes Monday and 7 is used for Sunday).

### C. Offset Expressions

*Offset expressions*, as we call them, encode a function which, when applied to a *reference time*, returns the global semantic value denoting the temporal location of the entity referred to by the expression. This temporal function either adds or subtracts a number of units at some granularity: for example, *last*

<sup>4</sup>Underspecified expressions should not be confused with non-specific expressions; these represent two quite independent semantic phenomena. The former omits some information because it is assumed the reader will be able to retrieve it from the context (e.g. *14th June*), while the latter is typically used generically (e.g. 'The dry season starts in *May*'). In the string-based semantic representation, the underspecified expressions we introduce in LTIMEX use lowercase *xs* (e.g. *xxxx-06-14*), while non-specific expressions, already part of TIMEX2, are annotated with uppercase *Xs* (e.g. *XXXX-WXX-7TMO*).

TABLE VI  
LOCAL SEMANTICS OF OFFSET EXPRESSIONS OF DATES.

No	Deictic Expression	Anaphoric Expression	L-VAL
1	today	the same day	+0000-00-00
2	tomorrow	the following day	+0000-00-01
3	yesterday	the previous day	-0000-00-01
4	five days ago	five days earlier	-0000-00-05
5	last month	the previous month	-0000-01
6	last summer	the previous summer	-0001-SU
7	two weeks ago	two weeks earlier	-0000-W02
8	(in) two weeks	two weeks later	+0000-W02
9	this weekend	that weekend	+0000-W00-WE
10	this year	that year	+0000
11	three years ago	three years earlier	-0003
12	next century	the following century	+01

*year* is equivalent to subtracting one year from the year of the reference date, and *three days later* means adding three days.

Based on the reference time used, we further distinguish two kinds of offsets: these may be either *deictic* or *anaphoric*. For deictic expressions the reference time is the time-stamp of the utterance<sup>5</sup> (*S* in the Reichenbachian framework [12, pp. 291–298]) and for anaphoric expressions the reference is to be found somewhere in the context. For example, *yesterday* is deictic, but *the previous day* is an anaphoric expression.

In LTIMEX, both these expressions have the same offset encoded as the value of the L-VAL attribute; the L-TYPE attribute indicates whether the expression is *OFFSET-DEICTIC* or *OFFSET-ANAPHORIC*. The interpretation algorithm can use the value of this attribute to decide whether to apply the offset to the time-stamp of the document or to use a temporal focus tracking mechanism to select the correct reference time. If, for any reason, the annotator or a temporal expression tagger cannot decide on the subtype of the offset, L-TYPE can be specified simply as *OFFSET*, leaving the decision about the subtype to the interpretation module.

Table VI presents pairings of deictic and anaphoric date expressions which share the same value for the L-VAL attribute. A leading + or - indicates whether the operation to be performed is addition or subtraction; for consistency with TIMEX2, we use the ISO-based format to encode the magnitude of the offset. The number of filled slots determines the granularity of the unit of the operation. For example, +0000-00-05 encodes the addition of five days and -0002 encodes the subtraction of two years. Of course, for expressions with zero offset (e.g. *today*) one could use either + or -; by convention we use +.

An offset date expression may be accompanied by unambiguous (e.g. *6 a.m.*) or ambiguous (e.g. *6 o'clock*) information about the time within the referred-to day; see Rows 1–9 of Table VII. In these cases only the date component of the expression (e.g. *today* or *tomorrow*) has the form of an offset; here the *T* and *t* separators combine an offset on their left with an absolute value on their right.

<sup>5</sup>In practical terms, the utterance time may be the time of speaking, the date of publication, the date and time of sending an email, and so on.

TABLE VII  
THE LOCAL SEMANTICS OF OFFSET EXPRESSIONS WITH REFERENCES TO TIMES OF DAY.

No	Expression	Representation (L-VAL)	Type
1	6 a.m. today	+0000-00-00T06:00	deictic date offset + explicit time
2	6 p.m. that day	+0000-00-00T18:00	anaphoric date offset + explicit time
3	6 p.m. two days ago	-0000-00-02T18:00	deictic date offset + explicit time
4	6 o'clock two days ago	-0000-00-02t06:00	deictic date offset + underspecified time
5	tomorrow morning	+0000-00-01TMO	deictic date offset + explicit time
6	morning the day before	-0000-00-01TMO	anaphoric date offset + explicit time
7	last night	-0000-00-01TNI	deictic date offset + explicit time
8	11pm last night	-0000-00-01T23:00	deictic date offset + explicit time
9	2am last night	+0000-00-00T02:00	deictic date offset + explicit time
10	two hours earlier	+0000-00-00T-02	anaphoric time offset
11	an hour and twenty minutes later	+0000-00-00T+01:20	anaphoric time offset
12	in six hours time	+0000-00-00T+06	deictic time offset
13	five minutes ago	+0000-00-00T-00:05	deictic time offset
14	in sixty seconds	+0000-00-00T+00:00:60	deictic time offset
15	sixty seconds later	+0000-00-00T+00:00:60	anaphoric time offset
16	tomorrow two hours later	+0000-00-01T+02	deictic date offset + time offset
17	the next day two hours later	+0000-00-01T+02	anaphoric date offset + time offset
18	8 May 2001, one hour later	2001-05-08T+01	explicit date + time offset
19	17 May, one hour earlier	xxxx-05-17T-01	underspecified date + time offset

Just as there can be date offsets that have no time information, we can also have time offsets with no date information; for example, *five minutes ago*. In such cases we add the operator (+ or -) just after the T separator. Consider the representation of *eighteen hours and fifteen minutes later*: this has the value +0000-00-00T+18:15, making it distinct from the representation of *6:15pm today*, which is +0000-00-00T18:15. More examples are provided in Rows 10–15 of Table VII.

A time offset may also appear together with a date offset, as shown in Rows 16 and 17 in Table VII, or even with an explicit or underspecified date, as shown in Rows 18 and 19. In the first case the representation combines a non-zero date offset with a time offset; in the second case we have a non-offset representation of a date followed by the encoding of the time offset.

Finally, we also need to be able to represent offset expressions built on cycle-based calendar elements (weekday and month names), such as *last Monday* or *next March*. In Table VIII we present examples with the proper encodings. In our representation we only indicate the direction (< for *last*, > for *next*) and the weekday or month name mentioned in the expression ( $D_n$  and  $M_{nn}$ , respectively). It will be the task of the interpretation stage to determine which calendar week and year is intended. Expressions using the determiner *this* (e.g. *this Wednesday* or *this June*) are treated as underspecified expressions unless the determiner is used together with other tokens indicating the direction of interpretation (e.g. *this coming Wednesday*).

#### D. Event-based Point Expressions

An event-based expression identifies a temporal entity by means of a reference to an event. In such expressions, the L-TYPE attribute has the value

TABLE VIII  
THE LOCAL SEMANTICS OF OFFSET EXPRESSIONS INVOLVING ELEMENTS OF CYCLE-BASED CALENDARS.

No	Deictic Expression	Anaphoric Expression	L-VAL
1	last Monday	the previous Monday	<D1
2	next Wednesday	the next Wednesday	>D3
3	this coming Wednesday	that coming Wednesday	>D3
4	this Wednesday	that Wednesday	xxxx-Wxx-3
5	last June	the previous June	<M06
6	next June	the next June	>M06
7	this June	that June	xxxx-06

L-TYPE=EVENT-BASED-POINT, and we provide the identifier of the event in the L-EVENT\_ID attribute. If an application does not perform event recognition, or in a given circumstance is unable to identify the event in question, then the value of this attribute is left empty.

In some cases the temporal value is expressed as an offset to the time of an event, as in Example (2):

- (2) *Ten seconds after the second explosion* the plane hit the ground.

L-VAL=+0000-00-00T+00:10

L-TYPE=EVENT-BASED-POINT L-EVENT\_ID=e

- (3) Jane got a salary raise *the day after Michael lost his job*.

L-VAL=+0000-00-01 L-TYPE=EVENT-BASED-POINT

L-EVENT\_ID=e

Here the L-VAL attribute encodes the offset, just as it does in offset point expressions. The specified type of the expression indicates that the reference time to be used in the interpretation is the time of the event indicated by the  $e$  event variable.

In cases when the time denoted by the expression can be computed from the time-stamp of the event simply by refining its granularity, we use a zero-offset just to indicate the granularity (temporal unit) of the result. Consider the

following example:

- (4) I met my wife *the year when I bought my house*.  
 L-VAL=+0000  
 L-TYPE=EVENT-BASED-POINT L-EVENT\_ID=e

The temporal value of the expression is to be calculated here by adding zero years to the year of the event time-stamp, and discarding any more detailed information that the time-stamp might provide (e.g. the month and day).

In other cases, the time denoted by the expression may be exactly the time of the event, as in the following example:

- (5) *At the time of the peace agreement* the United States agreed to replace equipment on a one-by-one basis.  
 L-VAL=EVENT\_TIME  
 L-TYPE=EVENT-BASED-POINT L-EVENT\_ID=e

Note that the expression does not indicate the granularity. In such cases, the L-VAL attribute contains the EVENT\_TIME token, which means that the temporal value is the time of the underlying event.

For point temporal expressions which refer to a part of an event, as in Example (6), we use the encoding of ordinaly-specified expressions, which we discuss in detail in Section III-G:

- (6) The casualties included 19,240 dead on *the third day of the Battle of the Somme*.  
 L-VAL=3D L-TYPE=EVENT-BASED-POINT  
 L-EVENT\_ID=e

The 3D value tells us that, of the whole time span of the event, the expression refers only to the third day.

#### E. Period Expressions

For expressions that denote periods, L-VAL takes the same values as the corresponding VAL attribute in TIMEX2; this also covers those cases where the duration mixes different units, as in the following example:

- (7) This project will run for *one year and two months*.  
 L-VAL=P1Y2M

The anchoring attributes are to be filled in only if the anchor is mentioned within the extent of the expression. The anchor may be provided in various forms, including an explicit (see Example (8)), underspecified (see Example (9)) or offset (see Examples (10) and (11)) point. In each case, the L-ANCHOR\_VAL attributes encode that anchoring point in one of the formats we have already introduced:

- (8) The accounts are paid in full for *the six months ended 31 March 2009*.  
 L-VAL=P6M  
 L-ANCHOR\_VAL=2009-03-31 L-ANCHOR\_DIR=ENDING
- (9) The accounts are paid in full for *the six months ended March 31*.  
 L-VAL=P6M  
 L-ANCHOR\_VAL=xxxx-03-31 L-ANCHOR\_DIR=ENDING
- (10) The renovations will last *five days starting tomorrow*.  
 L-VAL=P5D L-ANCHOR\_TYPE=DEICTIC  
 L-ANCHOR\_VAL=+0000-00-01  
 L-ANCHOR\_DIR=STARTING

- (11) The movie festival will end on *18 July*, but then we have the theatre workshops that will run for *a whole week starting just the very next day*.  
 L-VAL=P1W L-ANCHOR\_TYPE=ANAPHORIC  
 L-ANCHOR\_VAL=+0000-00-01  
 L-ANCHOR\_DIR=STARTING

In the last example above we also use the L-ANCHOR\_TYPE attribute to encode the type of the offset of the anchor; the possible values here are DEICTIC and ANAPHORIC. The expression may be also anchored implicitly, as in the following example:

- (12) *The next three days* were extremely hot and humid.  
 L-VAL=P3D  
 L-ANCHOR\_VAL=+0000-00-00  
 L-ANCHOR\_DIR=STARTING

In such cases we provide the offset in the L-ANCHOR\_VAL attribute, but leave it to the interpretation algorithm to decide (for example, based on the tense of the sentence) whether the anchor is deictic or anaphoric.

If the expression itself does not state when the period starts or ends, then no anchor-related attributes are specified:

- (13) The Nile Movie Festival lasted *five days*. L-VAL=P5D

If the rest of the document provides such information, the anchor is to be determined in the interpretation stage, when the global semantics is derived; this also means that the final annotation does not have the L-ANCHOR\_VAL and L-ANCHOR\_DIR attributes, it only has ANCHOR\_VAL and ANCHOR\_DIR.

#### F. Event-based Period Expressions

For event-based periods, we encode the duration in the L-VAL attribute just as in the case of other durations discussed in Section III-E, but the type of the expression in the L-TYPE attribute is specified as EVENT-BASED-PERIOD. Similarly to the annotation of event-based point expressions, we provide the identifier of the underlying event in the L-EVENT\_ID attribute. The time of the event, however, does not serve here as the reference time to be used in the following interpretation stage to calculate the value of the VAL attribute; rather, it determines the location of the period, and is used to compute one of the period's anchors. Consider the following examples:

- (14) The rate of US combat deaths in Baghdad nearly doubled in *the first seven weeks of the "surge" in security activity*.  
 L-VAL=P7W  
 L-ANCHOR\_VAL=EVENT\_START  
 L-ANCHOR\_DIR=STARTING  
 L-EVENT\_ID=e L-TYPE=EVENT-BASED-PERIOD
- (15) *The last three days of the battle* were extremely brutal.  
 L-VAL=P3D  
 L-ANCHOR\_VAL=EVENT\_END L-ANCHOR\_DIR=ENDING  
 L-EVENT\_ID=e L-TYPE=EVENT-BASED-PERIOD
- (16) I was so panicked I could not take a single step for *30 minutes after the earth quake*.  
 L-VAL=PT30M  
 L-ANCHOR\_VAL=EVENT\_END

TABLE IX  
THE LOCAL SEMANTICS OF ORDINALLY-SPECIFIED REFERENCES.

No	Expression	Representation (L-VAL)
1	the first Tuesday	1D2
2	the third day	3D
3	the last Tuesday	\$1D2
4	the last day	\$1D
5	the last but one day	\$2D
6	the penultimate day	\$2D
7	the last month	\$1M
8	the last February	\$1M02

L-ANCHOR\_DIR=STARTING

L-EVENT\_ID=e L-TYPE=EVENT-BASED-PERIOD

- (17) There was no terrorist warning in *the three years before the bombing in the underground.*

L-VAL=P3Y

L-ANCHOR\_VAL=EVENT\_START

L-ANCHOR\_DIR=ENDING

L-EVENT\_ID=e L-TYPE=EVENT-BASED-PERIOD

To handle the fact that events themselves span over some periods of time, we introduce two tokens, `EVENT_START` and `EVENT_END`, to be used in the `L-ANCHOR_VAL` attribute. These tokens indicate which end of the period of the event is to be used as the anchor.

### G. Ordinally-specified expressions

Some temporal expressions use what we call *ordinally-specified elements*; for example, the expressions in Examples (18a)–(18c) make reference to a specific day by means of selecting the third day of some coarser temporal unit or an event.

- (18) a. *the third day of the next month*  
 b. *the third day of the previous decade*  
 c. *the third day of the trip*

To encode such ordinally-specified elements we use the format `$nu`, where `n` is a number, `u` indicates the temporal unit to be used, and `$` is an optional marker used when the ordinal is to be counted from the end of some chunk of time (e.g. *last*, *penultimate*). Examples of ordinally-specified elements of expressions and their representations are shown in Table IX.

Expressions using ordinally-specified elements are annotated with multiple `TIMEX2` annotations, as shown in Examples (19)–(21). The ordinally-specified format is recorded only in the outermost annotation; the inner expression, which may be, for example, underspecified or an offset, receives its own proper representation of its local semantics.

- (19) `<TIMEX2 L-VAL="3D">the third day of`  
`<TIMEX2 L-VAL="+0000-01">the next month`  
`</TIMEX2></TIMEX2>`  
 (20) `<TIMEX2 L-VAL="$1D1">the last Monday of <TIMEX2`  
`L-VAL="xxxx-05">May</TIMEX2></TIMEX2>`  
 (21) `<TIMEX2 L-VAL="1D">the first day of`  
`<TIMEX2 L-VAL="2M">the second month of`

`<TIMEX2 L-VAL="+0001">next year</TIMEX2>`  
`</TIMEX2></TIMEX2>`

When deriving the global semantics from the local semantics, the individual values of the nested expressions must be combined together; the process is carried out recursively from the outermost to the innermost, resolving the temporal references while backtracking from the innermost to the outermost.

The type recorded in the `L-TYPE` of an expression whose `L-VAL` is ordinally specified is the same as the type of its innermost expression; Example (19) is an anaphoric offset, Example (20) is underspecified, and Example (21) is deictic.

### H. Non-Specific Expressions

In many cases the decision that a temporal expression is non-specific can be only made when analysing the whole sentence, or even the entire document. For example, consider the generic references to months in the following sentence:

- (22) In the southern hemisphere *days* are much longer in *January* than in *July*.

These are not obviously non-specific when we consider only the extent of the expressions themselves. The local semantic representations are therefore underspecified, i.e. `xxxx-01` and `xxxx-07`. In the interpretation stage, the lowercase `xs` must not be instantiated with a specific year, but must be converted into markers of non-specificity (uppercase `Xs`, for example, if `TIMEX2` is the scheme used for global semantics representation).

Indefinite noun phrases, on the other hand, can already be recognized as non-specific at the level of local semantics:

- (23) a. I was born on *a Sunday*. `L-VAL=XXXX-WXX-7`  
 b. I met my wife on *a sunny day in July*.  
`L-VAL=XXXX-07-XX`

In such cases, we can already mark the relevant slots as non-specific, obtaining the same value as expected in `VAL`.

Periods of indefinite duration, such as *a few days*, can also be recognized as non-specific without reference to the context. The encoding of such durations uses `X` instead of a specific number, e.g. `PXD`.

Similarly, some set expressions can be identified as non-specific already at the stage of local semantic analysis; for example, *every few days* or *some Mondays in 2004*. Unfortunately, `TIMEX2` is unable to represent the semantics of these expressions correctly,<sup>6</sup> and in consequence our representation fails here too.

### I. Set Expressions

The semantic representation of set expressions is complex, because these expressions do not refer to a single entity, but to a set of entities. Neither `TIMEX2` nor `TimeML` express the semantics of these expressions sufficiently well to make these schemes applicable to all set expressions. As an alternative, Pan's [13] first-order logic representation for set expressions, which is formally sound and has much broader coverage, can

<sup>6</sup>For example, *some Mondays in 2004* is represented just in the same way as *all Mondays in 2004*: `VAL=2004-WXX-1`, `SET=YES`.

be encoded in OWL; but the complexity of OWL goes far beyond the goals of TIMEX2 and TimeML.

As indicated earlier, our aim is to provide a representation for local semantics that is compatible with the use of TIMEX2 for representing the global semantics of temporal expressions. Inevitably, this compromises the expressiveness of our representation.

We indicate the set type by assigning the value YES to the L-SET attribute (following the use of the SET attribute in TIMEX2), and we specify any underspecification or offset that might appear, as in the following examples:

- (24) a. *every winter in the 80s* L-VAL=xx8-WI L-SET=YES  
 b. *monthly* L-VAL=xxxx-XX L-SET=YES

In some cases we may be able to obtain a reliable representation by using values or attribute combinations not authorized in TIMEX2. For instance, in Example (25a) the expression is represented by means of any period of two years with its ending anchored on years having zero as their final digit (e.g. 1960, 1990, 2000). In Example (25b) we do something similar, but we anchor the periods on the last day of a month (and in doing so we specify the anchor with the format used for ordinal-specified references).

- (25) a. *the last two years of every decade*  
 L-VAL=P2Y L-SET=YES  
 L-ANCHOR\_VAL=XXX0 L-ANCHOR\_DIR=ENDING  
 b. *the last two days of every month*  
 L-VAL=P2D L-SET=YES  
 L-ANCHOR\_VAL=XXXX-XX-\$1D  
 L-ANCHOR\_DIR=ENDING

This, however, already goes beyond the TIMEX2 rules, which prohibit using the anchor attributes for set expressions [2, p. 42].

#### IV. CONCLUSION

We have developed a string-based representation of the context-independent semantics of temporal expressions, which we call LTIMEX. It can be easily integrated with the existing annotation schemes (specifically, TIMEX2 and TimeML) which currently allow only for the representation of fully-interpreted semantics. We are thus proposing an extension to these schemes that provides a means of support for an additional level of semantic representation; this in turn leads to a modular design of temporal expression tagging, with a well-defined interface between the recognition and interpretation modules, and allows for more detailed evaluation of taggers.

Table II summarises the attributes used in LTIMEX and their values. We use in total eight attributes: three are used in the same way as their TIMEX2 counterparts (L-MOD, L-SET and L-ANCHOR\_DIR); L-VAL represents the partial<sup>7</sup> context-independent meaning of the expression; similarly, L-ANCHOR\_VAL encodes information about the temporal location of an anchor of a period; and three attributes are

<sup>7</sup>It is partial in the sense that it does not capture information about temporal modifiers and anchors, which are encompassed in separate attributes: L-MOD, L-ANCHOR\_DIR, and L-ANCHOR\_VAL.

completely new: L-TYPE, which encodes the taxonomical type of the expression; L-EVENT\_ID, which for event-based expressions stores the identifier of the event; and L-ANCHOR\_TYPE, which, for durations with the anchor expressed by means of an offset, encodes whether it is deictic or anaphoric.

The first obvious task that arises as possible future work is to use LTIMEX for a significant data annotation task; possible candidate corpora already annotated with temporal expressions are WikiWars [14] (TIMEX2) and TimeBank<sup>8</sup> (TimeML).

Another area left for future work is the improvement in the representation of set expressions. This could perhaps be aligned with further development of TimeML, which is to become an ISO standard (see the discussion in [15]); although this goes further than TIMEX2, it still does not have a proper means to represent the global semantics of set expressions.

#### REFERENCES

- [1] O. R. Alonso, "Temporal Information Retrieval," Ph.D. dissertation, University of California, 2008.
- [2] L. Ferro, L. Gerber, I. Mani, B. Sundheim, and G. Wilson, "TIDES 2005 Standard for the Annotation of Temporal Expressions," MITRE, Tech. Rep., September 2005.
- [3] J. Pustejovsky, J. Castaño, R. Ingria, R. Saurí, R. Gaizauskas, A. Setzer, and G. Katz, "TimeML: Robust Specification of Event and Temporal Expressions in Text," in *IWCS-5, Fifth International Workshop on Computational Semantics*, Tilburg, The Netherlands, January 2003.
- [4] R. Saurí, J. Littman, B. Knippen, R. Gaizauskas, A. Setzer, and J. Pustejovsky, "TimeML Annotation Guidelines Version 1.2.1," January 2006. [Online]. Available: <http://www.timeml.org/site/publications/specs.html>
- [5] NIST, "The ACE 2004 Evaluation Plan," 2004, [www.itl.nist.gov/iad/mig/tests/ace/2004/doc/ace04-evalplan-v7.pdf](http://www.itl.nist.gov/iad/mig/tests/ace/2004/doc/ace04-evalplan-v7.pdf).
- [6] I. Mani and G. Wilson, "Robust Temporal Processing of News," in *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (ACL '00)*, Morristown, NJ, USA: Association for Computational Linguistics, October 2000, pp. 69–76.
- [7] F. Schilder, "Extracting Meaning from Temporal Nouns and Temporal Prepositions," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 3, no. 1, pp. 33–50, March 2004.
- [8] D. Ahn, J. van Rantwijk, and M. de Rijke, "A Cascaded Machine Learning Approach to Interpreting Temporal Expressions," in *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2007)*, Rochester, NY, USA, April 2007.
- [9] P. Mazur and R. Dale, "The DANTE Temporal Expression Tagger," in *Proceedings of the 3rd Language And Technology Conference (LTC)*, Z. Vetulani, Ed., Poznan, Poland, October 2007.
- [10] J. Strötgen and M. Gertz, "HeidelTime: High Quality Rule-Based Extraction and Normalization of Temporal Expressions," in *Proceedings of the 5th International Workshop on Semantic Evaluation*. Uppsala, Sweden: ACL, July 2010, pp. 321–324.
- [11] P. Mazur and R. Dale, "An Intermediate Representation for the Interpretation of Temporal Expressions," in *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*. Sydney, Australia: Association for Computational Linguistics, July 2006, pp. 33–36.
- [12] H. Reichenbach, *Elements of Symbolic Logic*. Macmillan, 1947.
- [13] F. Pan, "Representing Complex Temporal Phenomena for the Semantic Web and Natural Language," Ph.D. dissertation, University of Southern California, 2007.
- [14] P. Mazur and R. Dale, "Wikiwars: A new corpus for research on temporal expressions," in *Proceedings of the EMNLP 2010, Conference on Empirical Methods in Natural Language Processing*, 2010.
- [15] J. Pustejovsky, K. Lee, H. Bunt, and L. Romary, "ISO-TimeML: An International Standard for Semantic Annotation," in *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association (ELRA), May 2010.

<sup>8</sup>See the catalogue entry LDC2006T08 at <http://www ldc.upenn.edu>.

# Dependency-Based Rules for Grammar Checking with LanguageTool

Maxim Mozgovoy  
University of Aizu  
Tsuruga, Ikki-machi, Aizu-Wakamatsu,  
Fukushima, 965-8580 Japan  
Email: mozgovoy@u-aizu.ac.jp

**Abstract**—This paper describes a possible extension of well-known open source grammar checking software LanguageTool. The proposed extension allows the developers to write grammar rules that rely on natural language parser-supplied dependency trees. Such rules are indispensable for the analysis of word-word links in order to handle a variety of grammar errors, including improper use of articles, incorrect verb government, and wrong word form agreement.

## I. INTRODUCTION

GRAMMAR checking is a well-recognized problem of natural language processing. Grammar checkers are helpful in a variety of scenarios, such as text authoring and language learning. The purpose of such tools is to find grammatical errors in the input text: incorrect use of person, number, case or gender, improper verb government, wrong word order, and so on. A grammar checker normally works in combination with a spellchecker—a module that detects spelling errors in individual words. As a rule, spell checker cannot correct even basic grammatical flaws, such as erroneous choice of article (like in the expression “an box”).

While a spellchecker is already an essential part of a modern text authoring system, a grammar checking module is still found only in large commercial packages like Microsoft Office or WordPerfect Office. Certain grammar checkers are also available as additional software packages or online services, offered by independent companies [1-3]

This situation is slowly changing nowadays. With the growing popularity of open source software, more natural language processing systems should become available for wider use. Open spellchecking libraries, such as JOrtho and GNU Aspell already exist, and anyone can extend own software with their capabilities. Grammar checking is a more challenging task, and most open projects are still far beyond well-established proofing tools, such as offered in MS Word.

### A. Rule-Based Grammar Checking

Probably, the predominating approach to grammar checking today consists in testing the input text against a set of handcrafted rules [4, 5]. For example, the rule

I + Verb (3<sup>rd</sup> person, singular form)

corresponds to the incorrect verb form usage, as in the phrase “I has a dog”. In order to emphasize the nature of

such rules as erroneous patterns, they are often called “mal-rules”.

This method has several attractive features: (a) rules can be easily added, modified or removed; (b) every rule can have a corresponding extensive explanation, helpful for the end user; (c) the system is easily debuggable, since its decisions can be traced to a particular rule; (d) the rules can be authored by the linguists, possessing limited or no programming skills. An obvious disadvantage of a rule-based system is a large amount of manual work, needed to build an extensive rule set.

An alternative approach is represented with several varieties of statistical systems that analyze existing collections of grammatically correct and incorrect texts, attempting to find word patterns and/or text features that correspond to correct sentences [6, 7]. The simplest statistical grammar algorithm consists in analyzing N-grams—chains of N consecutive words [8]. If a certain word chain is common in the master text corpus, it is considered correct.

Statistical grammar checkers have their own advantages and drawbacks, but their analysis is beyond the scope of this article.

### B. Introducing LanguageTool

The purpose of the present work is to design a possible extension for LanguageTool grammar checker [9]. LanguageTool is a modern rule-based open source grammar checking system, available both as a plug-in for OpenOffice.org and as a downloadable library, which makes it ready for use in any software projects. Currently LanguageTool supports 21 languages, though the number of ready grammar rules ranges from 4 for Lithuanian to 1810 for French (as of April, 2011). The rules can be authored by any interested contributors.

Unfortunately, the syntax of rules in LanguageTool does not allow formulating certain grammatical phenomena. In the subsequent sections, we will consider these limitations and a possible method to reduce them.

## II. BASIC DESIGN PRINCIPLES OF LANGUAGE TOOL

LanguageTool defines an XML-based language for describing mal-rules. In its simplest form, a mal-rule is just a sequence of tokens to be matched in the text:

```
<!-- "all be it" instead of "albeit" -->

<pattern>
  <token>all</token>
  <token>be</token>
  <token>it</token>
</pattern>
<message>Did you mean 'albeit'?</message>
```

The syntax of the rules is flexible and powerful: it is possible to use OR and NOT logic operations (“match token A or token B”; “match any token except C”), skip optional tokens, and, to some extent, use regular expressions.

Several syntactic elements are backed with additional linguistic modules — *sentence splitter* and *part-of-speech tagger*. Sentence splitter determines the boundaries of each sentence, thus allowing the user to find certain tokens exactly at the beginning or at the end of a sentence:

```
<!-- "another words," instead of
      "in other words,"
      at the beginning of a sentence -->

<pattern>
  <token postag="SENT_START"></token>
  <token>another</token>
  <token>words</token>
  <token>,</token>
</pattern>
<message>Did you mean
      'in other words'?</message>
```

Part-of-speech tagger determines every word’s part of speech, helping the user to find tokens that belong to a certain class:

```
<!-- "ca" + [personal pronoun] instead of
      "can" + [personal pronoun] -->

<pattern>
  <token>ca</token>
  <token postag="PRP"></token>
</pattern>
<message>Did you mean 'can'?</message>
```

LanguageTool makes use of third-party libraries for splitting and tagging the input text. Fortunately, a number of ready solutions are available for this purpose (e.g., Ratnaparkhi’s MXPOST and MXTERMINATOR [10, 11]).

### III. INTRODUCING DEPENDENCY-BASED RULES

Despite the high expressive power and flexibility, LanguageTool’s rule system has a notable shortcoming: it treats the input text as a sequence of tokens, ignoring tree-like nature of natural language sentences.

Consider, for example, the following problem. In English, a/an article should never be used with a noun in a plural form. The current LanguageTool rule to detect such a case is defined as follows:

```
<!--"a/an" article, then a plural noun -->

<pattern>
  <token regexp="yes">a|an</token>
  <token postag="NNS|NNPS"></token>
</pattern>
<message>Don't use indefinite articles
      with plural words.</message>
```

However, this rule ignores the fact that there can be any number of words between a/an and the corresponding noun (“a box”, “a wooden box”, “a simple wooden box”). The rule definition can be improved if we allow any number of optional adjectives between the article and the noun, but in general case this solution is inadequate.

In order to handle such problems, the grammar checker should analyze nonlinear structure of the phrase. An article is logically linked with a noun, regardless of any words between them. This nonlinear structure can be obtained with an additional module, known as *dependency parser*. This instrument represents the structure of every sentence with a *parse tree*, having words as nodes and logical links between them as edges (see Fig. 1).

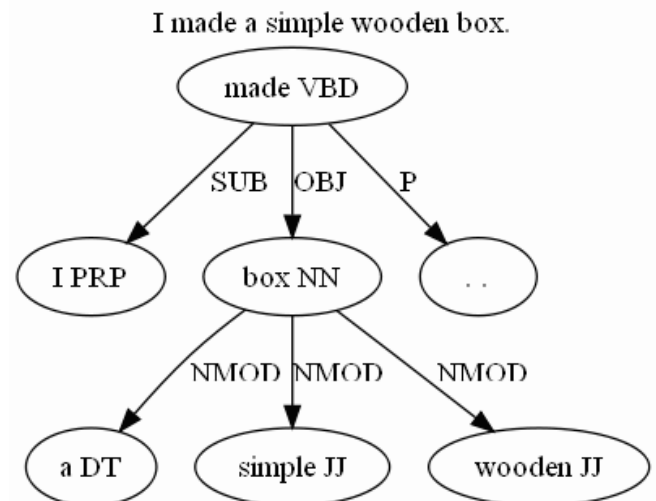


Fig. 1 Parse tree for the phrase “I made a simple wooden box”.

As it can be seen, the article “a” is linked directly to the noun “box”. Having such a tree, it is possible to extend the syntax of LanguageTool grammar rules, enabling the developers to analyze word-word relationships.

### IV. TECHNICAL APPROACH

In order to achieve our goals, we had to solve three sub-problems: 1) select a suitable dependency parsing instrument; 2) develop an appropriate syntax for dependency-based rules; 3) design the corresponding rule-matching algorithm.



### A. Selecting a Practical Dependency Parser

After examining currently available solutions, we decided to use one of two parsers: MaltParser [12] or LDPaR [13]. Both of them are high-quality dependency parsers, available as open source.

MaltParser is written in Java, and thus suits better for the use in combination with the current implementation of LanguageTool, also made with Java. LDPaR distribution contains cross-platform C++ code, providing compilable efficient implementation. Both parsers are based on machine learning: the parser first has to be trained with a collection of correctly parsed sentences (a *treebank*). MaltParser and LDPaR also share the same format of input and output data.

### B. Suggested Syntax for Dependency-Based Rules

Dependency-based rules should provide syntactic means for the following basic functions:

- 1) Match a link between two given words, optionally labeled with a given label. This function should be generalizable to the matching of the whole subtree.
- 2) Make sure that a certain word appears before or after another word, in order to control word precedence.
- 3) Ensure the absence of the given subtree in the parse tree.

In order to satisfy these requirements, we suggest the following syntax for an individual dependency-based rule. The rule definition is split into chunks, each representing a separate subtree to be matched:

```
CHUNK1
CHUNK2
...
CHUNKN
```

Every  $CHUNK_i$  is represented with a sequence of tokens, defined with token XML tag:

```
<token [attributes]>token-value</token>
```

Currently our system supports the following attributes:

- **pos**: the token should belong to the specified part-of-speech class;
- **label**: the link to the token's parent (according to the parse tree) should have the specified label;
- **parent**: the token should have the specified token as a parent (according to the parse tree);
- **except**: the token's value should not match token-value;
- **before**: the token should appear in the sentence before the specified token;
- **after**: the token should appear in the sentence after the specified token;
- **chunk\_start**: start-of-chunk marker;

- **inverse**: the current chunk (subtree) should not be found in the parse tree.

Attributes **parent**, **before**, and **after** expect a token's cardinal number within the current chunk as an argument. By default, every chunk of the rule has to be matched in the parse tree in order to satisfy the rule.

### C. Examples

The following examples illustrate the capabilities of dependency-based rules:

```
<!-- Example 1:
in non-interrogative sentences
the subject should be placed before
the predicate -->
```

```
<token pos="VB|VBP|VBZ|VBD"
label="ROOT"></token>
<token after="1" label="SUB"></token>
```

```
<token chunk_start="" inverse=""
label="ROOT"></token>
<token parent="1"?></token>
```

The first chunk ensures that the system has found a subject (labeled SUB), placed after the main verb. The second chunk asserts the absence of "?" mark, linked to the tree root.

```
<!-- Example 2:
"a/an" should not be used
with plural nouns -->
```

```
<token>a|an</token>
<token pos="NNS|NNPS" parent="1"></token>
```

This mal-rule finds a/an articles, linked to plural nouns (marked as NNS or NNPS by a part-of-speech tagger). Note that the determiner (such as an article) is always directly linked with the corresponding word, even if they are not adjacent in the original sentence.

```
<!-- Example 3:
the gerund should be used in conjunction
with auxiliary verbs -->
```

```
<token pos="VBG" label="ROOT"></token>
```

If a gerund (verb ing-form) is considered a parse tree root, this means the absence of an obligatory auxiliary verb (such as "is", "was"). If an auxiliary verb is present, it becomes a root element of the tree.

```
<!-- Example 4:
improper personal verb form used -->
```

```
<token pos="VBZ"></token>
<token parent="1"
label="SUB">I|we|you|they</token>
```

If the subject of a certain verb is I/we/you/they, the verb should not be in the 3<sup>rd</sup> person singular form.

Concerning the design of the actual subtree matching algorithm, it is implemented as a straightforward recursive depth-first search routine.

## V. DISCUSSION

LanguageTool is a good example of an extensible rule-based grammar checker. Basic grammatical rules can be expressed by means of standard regular expressions. If their expressive power is insufficient to describe a certain rule, one can make use of additional natural language processing-powered syntactic elements, backed with sentence splitter and part-of-speech tagger.

This architecture can be extended further by incorporating other language processing modules. An obvious candidate for this role is natural language parser that shows immediate word-word relationships. We have demonstrated several examples of grammar errors, detectable with parser-powered mal-rules.

Since we consider rule-based grammar checking to be an established technology, the discussion of its advantages and drawbacks is beyond the scope of our work. However, our experiments have revealed weak points of the language tools we use (parser and part-of-speech tagger, mainly).

Normally, these tools, being based on machine learning algorithms, need initial training on annotated text data. Most such training collections are represented with grammatically correct sentences. Thus, ungrammatical phrases may contain previously unseen patterns, causing incorrect results. For example, a part-of-speech tagger cannot reliably determine a tag for the word "like" in the phrase "he like dogs", since such a pattern never appears in the training collection.

Since processing ungrammatical sentences is a crucial feature for a grammar checking module, this issue needs further research. One of the possible solutions would be to extend the training collection with ungrammatical sentences.

## VI. CONCLUSION

We have designed and implemented the mechanism of natural language parser-backed rules for a LanguageTool-based grammar checking module. Our syntax allows writing rules that analyze word-word dependencies in a given phrase. We have shown real examples of language phenomena, where such rules are much more helpful than built-in LanguageTool instruments.

## REFERENCES

- [1] J. Burston, "Bon Patron: An Online Spelling, Grammar, and Expression Checker," *CALICO Journal*, vol. 25, no. 2, pp. 337-347, 2008.
- [2] H.J. Chen, "Evaluating Two Web-based Grammar Checkers-Microsoft ESL Assistant and NTNU Statistical Grammar Checker," *International Journal of Computational Linguistics & Chinese Language Processing*, vol. 14, no. 2, pp. 161-180, 2009.
- [3] B. O'Regan, A. Mompean and P. Desmet, "From Spell, Grammar and Style Checkers to Writing Aids for English and French as a Foreign Language: Challenges and Opportunities," *Revue française de linguistique appliquée*, vol. 15, no. 2, pp. 67-84, 2010.
- [4] E.M. Bender *et al.*, "Arboretum: Using a precision grammar for grammar checking in CALL," *Proceedings of the InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, pp. 83-86, 2004.
- [5] M. Milkowski, "Developing an open-source, rule-based proofreading tool," *Software: Practice and Experience*, vol. 40, no. 7, pp. 543-566, 2010.
- [6] M.J. Alam, N. UzZaman and M. Khan, "N-gram based statistical grammar checker for Bangla and English," *Proceedings of ninth International Conference on Computer and Information Technology (ICCIIT 2006)*, 2006.
- [7] J. Wagner, J. Foster and J. van Genabith, "Detecting grammatical errors using probabilistic parsing," *Workshop on Interfaces of Intelligent Computer-Assisted Language Learning*, 2006.
- [8] J. Sjobergh, "The Internet as a Normative Corpus: Grammar Checking with a Search Engine," *Technical Report, KTH Nada*, 2006.
- [9] D. Naber, "A rule-based style and grammar checker," *Master's thesis, University of Bielefeld*, 2003.
- [10] A. Ratnaparkhi, "A maximum entropy model for part-of-speech tagging," *Proceedings of the conference on empirical methods in natural language processing*, vol. 1, pp. 133-142, 1996.
- [11] J.C. Reynar and A. Ratnaparkhi, "A maximum entropy approach to identifying sentence boundaries," *Proceedings of the fifth conference on Applied natural language processing*, pp. 16-19, 1997.
- [12] J. Nivre *et al.*, "MaltParser: A language-independent system for data-driven dependency parsing," *Natural Language Engineering*, vol. 13, no. 2, pp. 95-135, 2007.
- [13] P. Jian and C. Zong, "Layer-Based Dependency Parsing," *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC 23)*, pp. 230-239, 2009.

# Preserving pieces of information in a given order in HRR and $GA_c$

Agnieszka Patyk-Łońska

**Abstract**—Geometric Analogues of Holographic Reduced Representations (GA HRR or  $GA_c$ —the continuous version of discrete GA described in [16]) employ role-filler binding based on geometric products. Atomic objects are real-valued vectors in  $n$ -dimensional Euclidean space and complex statements belong to a hierarchy of multivectors. A property of  $GA_c$  and HRR studied here is the ability to store pieces of information in a given order by means of trajectory association. We describe results of an experiment: finding the alignment of items in a sequence without the precise knowledge of trajectory vectors.

**Index Terms**—distributed representations, geometric algebra, HRR, BSC, word order, trajectory associations, bag of words.

## I. INTRODUCTION

OVER the years several attempts have been made to preserve the order in which the objects are to be remembered with the help of binding and superposition. While some solutions to the problem of preserving pieces of information in a given order have proved ingenious, others are obviously flawed. Let us consider the representation of the word *eye*—it has three letters, one of which occurs twice. The worst possible choice of binding and superposition would be to store quantities of letters, e.g.

$$eye = twice * e + once * y, \quad (1)$$

since we would not be able to distinguish *eye* from *eey* or *yee*. Another ambiguous representation would be to remember the neighborhood of each letter

$$eye = before_y * e + between_e * y + after_y * e. \quad (2)$$

Unfortunately, such a method of encoding causes words *eye* and *eyeye* to have the same representation

$$\begin{aligned} eyeye &= before_y * e + 2 \cdot between_e * y + \\ & (before_y + after_y) * e + after_y * e \\ &= 2(before_y * e + between_e * y + after_y * e) \\ &= 2 eye. \end{aligned} \quad (3)$$

Real-valued vectors are normalized in most distributed representation models, therefore the factor of 2 would be most likely lost in translation. Such *contextual roles* (Smolensky [19]) cause problems when dealing with certain types of palindromes. Remembering positions of letters is also not a good solution

$$eye = letter_{first} * e + letter_{second} * y + letter_{third} * e \quad (4)$$

as we need to redundantly repeat the first letter as the third letter, otherwise we could not distinguish *eye* from *ey* or *ye*.

Secondly, this method of encoding will not detect similarity between *eye* and *yeye*.

A quantum-like attempt to tackle the problem of information ordering was made in [1]—a version of semantic analysis, reformulated in terms of a Hilbert-space problem, is compared with structures known from quantum mechanics. In particular, an LSA matrix representation [1], [10] is rewritten by the means of quantum notation. Geometric algebra has also been used extensively in quantum mechanics ([2], [4], [3]) and so there seems to be a natural connection between LSA and  $GA_c$ , which is the ground for future work on the problem of preserving pieces of information in a given order.

As far as convolutions are concerned, the most interesting approach to remembering information in a given order has been described in [12]. Authors present a model that builds a holographic lexicon representing both word meaning and word order from unsupervised experience with natural language texts comprising altogether 90000 words. This model uses simple convolution and superposition to construct  $n$ -grams recording the frequency of occurrence of every possible word sequence that is encountered, a window of about seven words around the target word is usually taken into consideration. To predict a word in a completely new sentence, the model looks up the frequency with which the potential target is surrounded by words present in the new sentence. To be useful,  $n$ -gram models need to be trained on massive amounts of text and therefore require extensive storage space. We will use a completely different approach to remembering information order—trajectory association described by Plate in [18]. Originally, this technique also used convolution and correlation, but this time items stored in a sequence are actually superimposed, rather than being bound together.

## II. GEOMETRIC ANALOGUES OF HRR

Holographic Reduced Representations (HRR) and Binary Spatter Codes (BSC) are distributed representations of cognitive structures where binding of role-filler codevectors maintains predetermined data size. In HRR [17], [18] binding is performed by means of circular convolution

$$(x \circledast y)_j = \sum_{k=0}^{n-1} x_k y_{j-k \bmod n}. \quad (5)$$

of real  $n$ -tuples or, in ‘frequency domain’, by componentwise multiplication of (complex)  $n$ -tuples,

$$(x_1, \dots, x_n) \circledast (y_1, \dots, y_n) = (x_1 y_1, \dots, x_n y_n). \quad (6)$$

Bound  $n$ -tuples are superposed by addition, and unbinding is performed by an approximate inverse. A dual formalism, where real data are bound by componentwise multiplication, was discussed by Gayler [9]. In BSC [13], [14] one works with binary  $n$ -tuples, bound by componentwise addition mod 2,

$$\begin{aligned} (x_1, \dots, x_n) \oplus (y_1, \dots, y_n) &= (x_1 \oplus y_1, \dots, x_n \oplus y_n), \\ x_j \oplus y_j &= x_j + y_j \pmod{2}, \end{aligned} \quad (7)$$

and superposed by pointwise majority-rule addition; unbinding is performed by the same operation as binding.

One often reads that the above models represent data by *vectors*, which is not exactly true. Given two vectors one does not know how to perform, say, their convolution or componentwise multiplication since the result depends on basis that defines the components. Basis must be fixed in advance since otherwise all the above operations become ambiguous. Geometric Analogues of Holographic Reduced Representations (GA HRR) [5] can be constructed if one defines binding by the geometric product, a notion introduced in 19th century works of Grassmann [11] and Clifford [8].

In order to grasp the main ideas behind GA HRR let us consider an orthonormal basis  $b_1, \dots, b_n$  in some  $n$ -dimensional Euclidean space. Now consider two vectors  $x = \sum_{k=1}^n x_k b_k$  and  $y = \sum_{k=1}^n y_k b_k$ . The *scalar*

$$x \cdot y = y \cdot x \quad (8)$$

is known as the *inner product*. The *bivector*

$$x \wedge y = -y \wedge x \quad (9)$$

is the *outer product* and may be regarded as an oriented plane segment (alternative interpretations are also possible, cf. [7]).  $\mathbf{1}$  is the identity of the algebra. The geometric product of  $x$  and  $y$  then reads

$$xy = \underbrace{\sum_{k=1}^n x_k y_k}_{x \cdot y} \mathbf{1} + \underbrace{\sum_{k < l} (x_k y_l - y_k x_l) b_k b_l}_{x \wedge y}. \quad (10)$$

Grassmann and Clifford introduced geometric product by means of the basis-independent formula involving the *multivector*

$$xy = x \cdot y + x \wedge y \quad (11)$$

which implies the so-called Clifford algebra

$$b_k b_l + b_l b_k = 2\delta_{kl} \mathbf{1}. \quad (12)$$

when restricted to an orthonormal basis. Inner and outer product can be defined directly from  $xy$ :

$$x \cdot y = \frac{1}{2}(xy + yx), \quad x \wedge y = \frac{1}{2}(xy - yx).$$

The most ingenious element of (11) is that it adds two apparently different objects, a scalar and a plane element, an operation analogous to addition of real and imaginary parts of

a complex number. Geometric product for vectors  $x, y, z$  can be axiomatically defined by the following rules:

$$\begin{aligned} (xy)z &= x(yz), \\ x(y+z) &= xy + xz, \\ (x+y)z &= xz + yz, \\ xx &= x^2 = |x|^2, \end{aligned}$$

where  $|x|$  is a positive scalar called the magnitude of  $x$ . The rules imply that  $x \cdot y$  must be a scalar since

$$xy + yx = |x + y|^2 - |x|^2 - |y|^2.$$

Geometric algebra allows us to speak of inverses of vectors:  $x^{-1} = x/|x|^2$ .  $x$  is invertible (i.e. possesses an inverse) if its magnitude is nonzero. Geometric product of an arbitrary number of invertible vectors is also invertible. The possibility of inverting all nonzero-magnitude vectors is perhaps the most important difference between geometric and convolution algebras.

Geometric products of *different* basis vectors

$$b_{k_1 \dots k_j} = b_{k_1} \dots b_{k_j},$$

$k_1 < \dots < k_j$ , are called *basis blades* (or just *blades*). In  $n$ -dimensional Euclidean space there are  $2^n$  different blades. This can be seen as follows. Let  $\{x_1, \dots, x_n\}$  be a sequence of bits. Blades in an  $n$ -dimensional space can be written as

$$c_{x_1 \dots x_n} = b_1^{x_1} \dots b_n^{x_n}$$

where  $b_k^0 = \mathbf{1}$ , which shows that blades are in a one-to-one relation with  $n$ -bit numbers. A general multivector is a linear combination of blades,

$$\psi = \sum_{x_1 \dots x_n=0}^1 \psi_{x_1 \dots x_n} c_{x_1 \dots x_n}, \quad (13)$$

with real or complex coefficients  $\psi_{x_1 \dots x_n}$ . Clifford algebra implies that

$$c_{x_1 \dots x_n} c_{y_1 \dots y_n} = (-1)^{\sum_{k < l} y_k x_l} c_{(x_1 \dots x_n) \oplus (y_1 \dots y_n)}, \quad (14)$$

where  $\oplus$  is given by (7). Multiplication of two basis blades is thus, up to a sign, in a one-to-one relation with exclusive alternative of two binary  $n$ -tuples. Accordingly, (14) is a projective representation of the group of binary  $n$ -tuples with addition modulo 2.

GA HRR is based on binding defined by geometric product (14) of blades while superposition is just addition of blades (13). The discrete  $GA_d$  is a version of GA HRR obtained if  $\psi_{x_1 \dots x_n}$  in (13) equal  $\pm 1$ . The first recognition tests of  $GA_d$ , as compared to HRR and BSC, were described in [16]. In the present paper we go further and compare HRR and BSC with  $GA_c$ , a version of GA HRR employing “projected products” [5] and arbitrary real  $\psi_{x_1 \dots x_n}$ .

Throughout this paper we shall use the following notation: “\*” denotes binding roles and fillers by means of the geometric

product and “+” denotes the superposition of sentence chunks, e.g.

$$“Fido bit Pat” = bite_{agt} * Fido + bite_{obj} * Pat. \quad (15)$$

Additionally, “⊗” will denote binding performed by circular convolution used in the HRR model and  $a^*$  denotes the involution of a HRR vector  $a$ . A “+” in the superscript of  $x^+$  denotes the operation of reversing a blade or a multivector  $x$ :  $(b_{k_1 \dots k_j})^+ = b_{k_j} \dots b_{k_1}$ . Asking a question will be denoted with “‡”, as in

$$\begin{aligned} “Who bit Pat?” \\ &= (bite_{agt} * Fido + bite_{obj} * Pat) ‡ bite_{agt} \quad (16) \\ &\approx Fido. \end{aligned}$$

The *size* of a (multi)vector means the number of memory cells it occupies in computer’s memory, while the *magnitude* of a (multi)vector  $V = \{v_1, \dots, v_n\}$  is its Euclidean norm  $\sqrt{\sum_{i=1}^n v_i^2}$ .

For our purposes it is important that geometric calculus allows us to define in a very systematic fashion a hierarchy of associative, non-commutative, and invertible operations that can be performed on  $2^n$ -tuples. The resulting superpositions are less noisy than the ones based on convolutions, say. Geometric product preserves dimensionality at the level  $2^n$ -dimensional *multivectors*, where  $n$  is the number of bits indexing basis vectors. Moreover, all nonzero vectors are invertible with respect to geometric product, a property absent for convolutions and important for unbinding and recognition. A detailed analysis of links between GA HRR, HRR and BSC can be found in [5].

### III. THE GA<sub>C</sub> MODEL

The procedure we employ was suggested in [5]. The space of  $2^n$ -tuples is split into subspaces corresponding to scalars (0-vectors), vectors (1-vectors), bivectors (2-vectors), and so on. At the bottom of the hierarchy lay vectors  $V \in \mathbb{R}^n$ , having rank 1 and being denoted as  $\overset{1}{V}$ . An object of rank 2 is created by multiplying two elements of rank 1 with the help of the geometric product. Let  $\overset{1}{V} = \{\alpha_1, \alpha_2, \alpha_3\}$  and  $\overset{1}{W} = \{\beta_1, \beta_2, \beta_3\}$  be vectors in  $\mathbb{R}^3$ . A multivector  $\overset{2}{X}$  of rank 2 in  $\mathbb{R}^3$  comprises the following elements (cf. [15])

$$\overset{2}{X} = \overset{1}{V} \overset{1}{W} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} \alpha_1 \beta_1 + \alpha_2 \beta_2 + \alpha_3 \beta_3 \\ \alpha_1 \beta_2 - \alpha_2 \beta_1 \\ \alpha_1 \beta_3 - \alpha_3 \beta_1 \\ \alpha_2 \beta_3 - \alpha_3 \beta_2 \end{bmatrix}, \quad (17)$$

the first entry in the array on the right being a scalar and the remaining three entries being 2-blades. For arbitrary vectors in  $\mathbb{R}^n$  we would have obtained one scalar (or, more conveniently:  $\binom{n}{0}$  scalars) and  $\binom{n}{2}$  2-blades.

Let  $\overset{2}{X} = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$  and  $\overset{1}{V} = \{\alpha_1, \alpha_2, \alpha_3\}$  be two multivectors in  $\mathbb{R}^3$ . A multivector  $\overset{3}{Z}$  of rank 3 in  $\mathbb{R}^3$  may

be created in two ways: as a result of multiplying either  $\overset{1}{V}$  by  $\overset{2}{X}$  or  $\overset{2}{X}$  by  $\overset{1}{V}$ . Let us concentrate on the first case

$$\overset{3}{Z} = \overset{1}{V} \overset{2}{X} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \gamma_4 \end{bmatrix} = \begin{bmatrix} \alpha_1 \gamma_1 - \alpha_2 \gamma_2 - \alpha_3 \gamma_3 \\ \alpha_1 \gamma_2 + \alpha_2 \gamma_1 - \alpha_3 \gamma_4 \\ \alpha_1 \gamma_3 + \alpha_2 \gamma_4 + \alpha_3 \gamma_1 \\ \alpha_1 \gamma_4 - \alpha_2 \gamma_3 + \alpha_3 \gamma_2 \end{bmatrix}. \quad (18)$$

Here, the first three entries in the resulting matrix are 1-blades, while the last entry is a 3-blade. For arbitrary multivectors of rank 1 and 2 in  $\mathbb{R}^n$  we would have obtained  $\binom{n}{1}$  vectors

and  $\binom{n}{3}$  trivectors. We cannot generate multivectors of rank higher than 3 in  $\mathbb{R}^3$ , but it is easy to check that in spaces  $\mathbb{R}^{n>3}$  a multivector of rank 4 would have  $\binom{n}{0}$  scalars,  $\binom{n}{2}$

bivectors and  $\binom{n}{4}$  4-blades. The number of  $k$ -blades in a multivector of rank  $r$  is described by Table I. It becomes clear that a multivector of rank  $r$  over  $\mathbb{R}^n$  is actually a vector over a  $\sum_{i=0}^{\lfloor \frac{r}{2} \rfloor} \binom{n}{2i+r \bmod 2}$ -dimensional space.

As an example let us consider the following roles and fillers being normalized vectors drawn randomly from  $\mathbb{R}^n$  with Gaussian distribution  $N(0, \frac{1}{n})$

$$\begin{aligned} Pat &= \{a_1, \dots, a_n\}, & name &= \{x_1, \dots, x_n\}, \\ male &= \{b_1, \dots, b_n\}, & sex &= \{y_1, \dots, y_n\}, \\ 66 &= \{c_1, \dots, c_n\}, & age &= \{z_1, \dots, z_n\}. \end{aligned} \quad (19)$$

*PSmith*, who is a 66 year old male named Pat, is created by first multiplying roles and fillers with the help of the geometric product

$$\begin{aligned} PSmith &= name * Pat + sex * male + age * 66 \\ &= name \cdot Pat + name \wedge Pat + sex \cdot male + \\ &\quad sex \wedge male + age \cdot 66 + age \wedge 66 \end{aligned} \quad (20)$$

$$\begin{aligned} &= \begin{bmatrix} \sum_{i=1}^n (a_i x_i + b_i y_i + c_i z_i) \\ a_1 x_2 - a_2 x_1 + b_1 y_2 - b_2 y_1 + c_1 z_2 - c_2 z_1 \\ a_1 x_3 - a_3 x_1 + b_1 y_3 - b_3 y_1 + c_1 z_3 - c_3 z_1 \\ \vdots \\ a_{n-1} x_n - a_n x_{n-1} + b_{n-2} y_n - b_n y_{n-1} + c_{n-1} z_n - c_n z_{n-1} \end{bmatrix} \\ &= [d_0, d_{12}, d_{13}, \dots, d_{(n-1)n}]^T \\ &= d_0 + d_{12} e_{12} + d_{13} e_{13} + \dots + d_{(n-1)n} e_{(n-1)n}, \end{aligned} \quad (21)$$

where  $e_1, \dots, e_n$  are orthonormal basis blades. In order to be decoded as much correctly as possible, *PSmith* should have the same magnitude as vectors representing atomic objects, therefore it needs to be normalized. Finally, *PSmith* takes the form of

$$PSmith = [\hat{d}_0, \hat{d}_{12}, \hat{d}_{13}, \dots, \hat{d}_{(n-1)n}]^T, \quad (22)$$

where  $\hat{d}_i = \frac{d_i}{\sqrt{\sum_{j=0,12}^{(n-1)n} d_j^2}}$ .

TABLE I  
NUMBERS OF  $k$ -BLADES IN MULTIVECTORS OF VARIOUS RANKS IN  $\mathbb{R}^n$

rank	scalars	vectors	bivectors	trivectors	4-blades	...	data size
1	0	$\binom{n}{1}$	0	0	0	...	$o\left(\binom{n}{1}\right)$
2	$\binom{n}{0}$	0	$\binom{n}{2}$	0	0	...	$o\left(\binom{n}{0} + \binom{n}{2}\right)$
3	0	$\binom{n}{1}$	0	$\binom{n}{3}$	0	...	$o\left(\binom{n}{1} + \binom{n}{3}\right)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$2r$	$\binom{n}{0}$	0	$\binom{n}{2}$	0	$\binom{n}{4}$	...	$o\left(\sum_{i=0}^r \binom{n}{2i}\right)$
$2r + 1$	0	$\binom{n}{1}$	0	$\binom{n}{3}$	0	...	$o\left(\sum_{i=0}^r \binom{n}{2i+1}\right)$

$PSmith$  is now a multivector of rank 2. The decoding operation

$$\begin{aligned} & name^+PSmith \\ &= name^+(name \cdot Pat + name \wedge Pat + sex \cdot male \\ & \quad + sex \wedge male + age \cdot 66 + age \wedge 66) \end{aligned} \quad (23)$$

will produce a multivector of rank 3 consisting of vectors and trivectors. However, the original  $Pat$  did not contain any trivector components—they all belong to the noise part and the only interesting blades in  $name^+PSmith$  are vectors. The expected answer is a vector, therefore there is no point in calculating the whole multivector  $name^+PSmith$  and only then comparing it with items stored in the clean-up memory. To be efficient, one should generate only the vector-part while computing  $name^+PSmith$  and skip the noisy trivectors.

Let  $\langle \cdot \rangle_k$  denote the projection of a multivector on  $k$ -blades. To decode  $PSmith$ 's  $name$  we need to compute

$$\begin{aligned} & \langle name^+PSmith \rangle_1 \\ &= name^+namePat + \langle name^+(name \wedge Pat \\ & \quad + sex \cdot male + sex \wedge male + age \cdot 66 + age \wedge 66) \rangle_1 \\ &= Pat + noise = Pat'. \end{aligned} \quad (24)$$

The resulting  $Pat'$  will still be noisy, but to a lesser degree than it would have been if the trivectors were present.

Formally, we are using a map  $*_{1,2}^1$  that transforms a multivector of rank 1 (i.e. an  $n$ -tuple) and a multivector of rank 2 (i.e. a  $(1 + \frac{(n-1)n}{2})$ -tuple) into a multivector of rank 1 without computing the unnecessary blades. Let  $X$  be a multivector of rank 2

$$X = \langle X \rangle_0 + \langle X \rangle_2 = x_0 + \sum_{l < m} x_{lm} e_l e_m, \quad (25)$$

where  $x_{lm} = -x_{ml}$ . If  $A = (A_1, \dots, A_n)$  is a decoding vector (actually, an inverse of a role vector), then

$$\begin{aligned} A *_{1,2}^1 X &= x_0 A + \sum_{l,m} A_l x_{lm} e_m \\ &= \sum_k (x A_k + \sum_l A_l x_{lk}) e_k \\ &= \sum_k Y_k e_k = Y, \end{aligned} \quad (26)$$

with  $Y = (Y_1, \dots, Y_n)$  being an  $n$ -tuple, i.e. a multivector of rank 1. More explicitly,

$$Y_k = (A *_{1,2}^1 X)_k = x_0 A_k + \sum_{l=1}^{k-1} A_l x_{lk} - \sum_{l=k+1}^n A_l x_{kl}. \quad (27)$$

The map  $*_{1,2}^1$  is an example of a *projected product*, introduced in [5], reconstructing the vector part of  $AX$  without computing the unnecessary parts. The projected product is basis independent, as opposed to circular convolutions. In general,  $*_{l,k}^m$  transforms the geometric product of two multivectors  $A$  and  $B$  into a multivector  $C$ .

We now need to compare  $Pat'$  with other items stored in the clean-up memory using the dot product, and since  $Pat'$  is a vector, we need to compare only the vector part. That means, if the clean-up memory contained a multivector  $M$  of an odd rank, we would also need to compute  $Pat' \cdot \langle M \rangle_1$  while searching for the right answer.

This method of decoding suggests that items stored in the clean-up memory should hold information about their ranks, which is dangerously close to employing fixed data slots present in localist architectures. However, a rank of a clean-up memory item can be “guessed” from its size. In a distributed model we also should not “know” for sure how many parts the projected product should reject, but it can certainly reject parts spanned by blades of highest grades.

Before providing formulas for encoding and decoding a complex statement we need to introduce additional notation for the projected product and the projection. We have already introduced the projected product  $*_{l,k}^m$  transforming the geometric product of two multivectors of ranks  $l$  and  $k$  into a multivector of rank  $m$ . This will not always be the case for complex statements, since we can produce a multivector that will not be of any given rank. Let  $*_{l,\{\alpha_1, \alpha_2, \dots, \alpha_k\}}^m$  denote the projected product transforming the geometric product of a multivector  $A$  and a multivector  $B$  containing  $\alpha_1$ -blades,  $\alpha_2$ -blades, ... and  $\alpha_k$ -blades into a multivector  $C$ . In this way, the projected product  $*_{1,2}^1$  may be written down as  $*_{1,\{0,2\}}^1$ . By analogy, let  $\langle \cdot \rangle_{\{\alpha_1, \alpha_2, \dots, \alpha_k\}}$  denote the projection of a multivector on components spanned by  $\alpha_1$ -blades,  $\alpha_2$ -blades, ... and  $\alpha_m$ -blades.

Let  $\Psi$  denote the normalized multivector encoding the sentence “*Fido bit PSmith*”, i.e.

$$\Psi = \underbrace{\text{bite}_{agt} * \text{Fido}}_{\text{rank 2}} + \underbrace{\text{bite}_{obj} * \text{PSmith}}_{\text{rank 3}}. \quad (28)$$

Multivector  $\Psi$  will contain scalars, vectors, bivectors and trivectors and can be written down as the following vector of dimension  $\sum_{i=0}^3 \binom{n}{i}$

$$\Psi = \underbrace{\alpha}_{\text{a scalar}} + \underbrace{\sum_{i=1}^n \beta_i e_i}_{\text{vectors}} + \underbrace{\sum_{1 \leq i < j} \gamma_{ij} e_{ij}}_{\text{bivectors}} + \underbrace{\sum_{1 \leq i < j < k} \delta_{ijk} e_{ijk}}_{\text{trivectors}} \quad (29)$$

#### IV. TRAJECTORY ASSOCIACION

In the HRR model vectors are normalized and therefore can be regarded as radii of a sphere of radius 1. If we attach a sequence of items, say  $A, B, C, D, E$  to arrowheads of five of those vectors, we obtain a certain *trajectory* on the surface of a sphere, that is associated with sequence  $ABCDE$ . This is a geometric analogue to the *method of loci* which instructs to remember a list of items by associating each term with a distinctive location along a familiar path. Let  $k$  be a randomly chosen HRR vector and let

$$k^i = k \otimes k^{i-1} = k^{i-1} \otimes k, \quad i > 1 \quad (30)$$

be its  $i$ th power, with  $k^1 = k$ . The sequence  $S_{ABCDE}$  is then stored as

$$S_{ABCDE} = A \otimes k + B \otimes k^2 + C \otimes k^3 + D \otimes k^4 + E \otimes k^5. \quad (31)$$

Of course, each power of  $k$  needs to be normalized before being bound with a sequence item. Otherwise, every subsequent power of  $k$  would be larger or smaller than its predecessor. As a result, every subsequent item stored in a sequence would have a bigger or a smaller share in vector  $S_{ABCDE}$ . Obviously, this method cannot be applied to the discrete GA model (described in [16]) or to BSC, since it is impossible to obtain more than two distinct powers of a vector with the use of XOR as a means of binding.

This technique has a few obvious advantages present in HRR but not in GA<sub>C</sub> had we wished to use ordinary vectors as first powers—different powers of a vector  $k$  would then be multivectors of different ranks. While  $k^i$  and  $k^{i \pm 1}$  are very similar in HRR, in GA<sub>C</sub> they would not even share the same blades. Further, the similarity of  $k^i$  and  $k^{i+m}$  in HRR is the same as the similarity of  $k^j$  and  $k^{j+m}$ , whereas in GA<sub>C</sub> that similarity would depend on the parity of  $i$  and  $j$ . In the light of these shortcomings, we need to use another structure acting as a first power in order to make trajectories work in GA<sub>C</sub>. Let  $t$  be a random normalized full multivector over  $\mathbb{R}^n$  and let us define powers of  $t$  in the following way

$$t^1 = t, \quad t^i = (t^{i-1})t \quad \text{for } i > 1. \quad (32)$$

We will store vectors  $a_1 \dots a_l$  in a sequence  $S_{a_1 \dots a_l}$  using powers of the multivector  $t$

$$S_{a_1 \dots a_l} = a_1 t + a_2 t^2 + \dots + a_l t^l. \quad (33)$$

To answer a question “*What is the second item in a sequence?*” in GA<sub>C</sub> we need to use the projected product

$$\langle S_{a_1 \dots a_l} (t^2)^+ \rangle_1 \approx a_2, \quad (34)$$

and to find out the place of item  $a_i$  we need to compute

$$(a_i)^+ S_{a_1 \dots a_l} \approx t^i. \quad (35)$$

Some may argue that such encoding puts a demand on items in the clean-up memory to hold information if they are roles or fillers, which is dangerously close to employing fixed data slots present in localist architectures. Actually, elements of a sequence can be recognized by their size, relatively shorter than the size of multivector  $t$  and its powers.

#### V. ITEM ALIGNMENT

We present an experiment using trajectory association and we comment on test results for HRR and GA<sub>C</sub> models. We tested whether the HRR and GA<sub>C</sub> models were capable of performing the following task:

*Given only a set of letters  $A, B, C, D, E$  and an encoded sequence  $S_{????}$  comprised of those five letters find out the position of each letter in that sequence.*

We assumed that no direct access to  $t$  or its powers is given—they do belong to the clean-up memory, but cannot be retrieved “by name”. One may think of this problem as a “black box” that inputs randomly chosen letter vectors and in return outputs a (multi)vector representing always the same sequence, irrespectively of the dimension of data. Inside, the black box generates (multi)vectors  $t, t^2, t^3, t^4, t^5$ . Their values are known to the observer but their names are not. Since we can distinguish letters from non-letters, the naive approach would be to try out all 120 alignments of letters  $A, B, C, D$  and  $E$  using all possible combinations of non-letters as the powers of  $t$ . Unfortunately, powers of  $t$  are different each time the black box produces a sequence. We will use an algorithm based on two assumptions:

- $t^x$ , if not recognized correctly, is more similar to highest powers of  $t$ ,
- letters lying closer to the end of the sequence are often offered as the incorrect answer to questions concerning letters, as in  $S \# t^n$ .

Assumption (a) can be easily justified: since lower powers of  $t$  are recognized correctly more often, higher powers of  $t$  come up more often as the incorrect answer to  $S \# A$ . Vector  $t^3$  is the correct answer to  $S_{xxAxx} \# A$ . However, if  $t^3$  is not recognized, the next most similar answer will be  $t^5$  because it contains three “copies” of  $t^3$ , indicated here by brackets

$$\{ t * ( t * [ t ] * t ) * t \}. \quad (36)$$

The second most similar item will be  $t^4$  because it contains two “copies” of  $t$ , and so on. The item least similar to  $t^3$  will

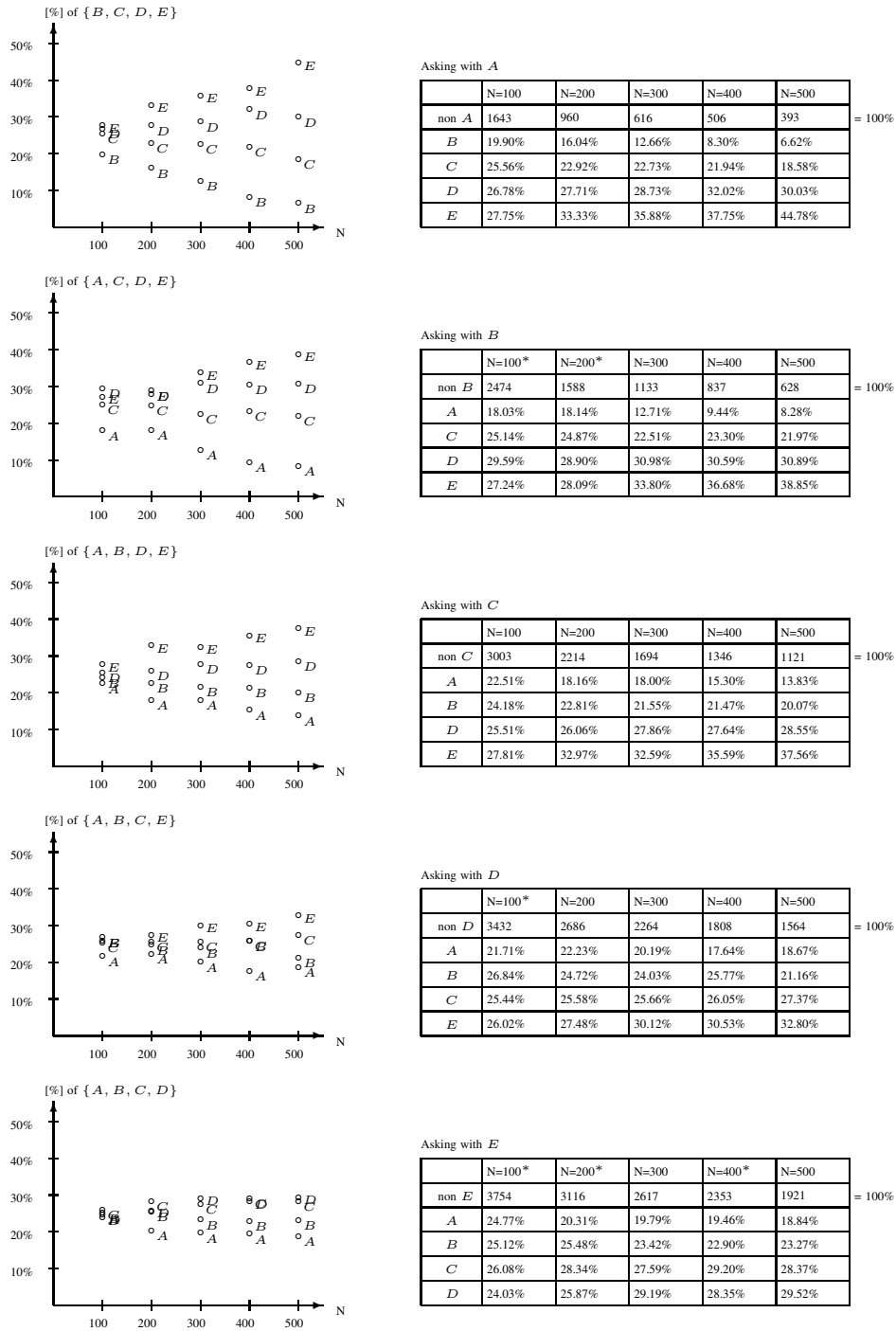


Fig. 1. Finding letter alignment in a sequence  $S_{ABCDE}$  in HRR, 10000 trials.

be  $t$ . Assumption (b) comes exactly from the same fact—a letter multiplied by one of the highest powers of  $t$  will be stored more “prominently” than other letters.

The clean-up memory  $\mathcal{C}$  for this experiment consists of all five letters and the five powers of  $t$ . We will also use an auxiliary clean-up memory  $\mathcal{L}$  containing letters only.

Since the normalization using the square root of the number of chunks proved very noisy in initial tests on statements containing powers of the trajectory vector, we decided to improve the HRR model. The HRR vectors in our tests were normalized by dividing them with their magnitude.



TABLE II  
 FINDING LETTER ALIGNMENT IN A SEQUENCE  $S_{ABCDE}$  IN GA<sub>C</sub>, 10000 TRIALS.

$\mathbb{R}^5$	$f_A$	$f_B$	$f_C$	$f_D$	$f_E$
asking with A	71.60%	<b>8.22%</b>	6.44%	2.72%	6.47%
asking with B	8.98%	65.92%	9.16%	6.76%	<b>9.18%</b>
asking with C	7.28%	9.52%	67.25%	<b>9.67%</b>	6.28%
asking with D	8.74%	6.75%	8.80%	66.83%	<b>8.88%</b>
asking with E	8.03%	<b>9.96%</b>	6.83%	9.28%	65.90%

$$\begin{array}{l}
 A \prec B \\
 *B \prec E \\
 C \prec D \\
 D \prec E \\
 *E \prec B
 \end{array}
 \Rightarrow
 \begin{array}{l}
 A \prec B \\
 C \prec D \prec E
 \end{array}$$
  

$\mathbb{R}^6$	$f_A$	$f_B$	$f_C$	$f_D$	$f_E$
asking with A	80.34%	<b>5.34%</b>	4.61%	5.08%	4.63%
asking with B	6.56%	73.91%	<b>7.48%</b>	4.67%	7.38%
asking with C	6.71%	7.47%	72.83%	<b>7.68%</b>	5.31%
asking with D	6.79%	5.37%	7.42%	72.30%	<b>8.12%</b>
asking with E	5.52%	7.77%	5.58%	<b>8.54%</b>	72.59%

$$\begin{array}{l}
 A \prec B \\
 B \prec C \\
 C \prec D \\
 *D \prec E \\
 *E \prec D
 \end{array}
 \Rightarrow
 \begin{array}{l}
 A \prec B \prec C \prec D
 \end{array}$$
  

$\mathbb{R}^7$	$f_A$	$f_B$	$f_C$	$f_D$	$f_E$
asking with A	89.78%	2.42%	<b>2.91%</b>	2.37%	2.52%
asking with B	3.78%	83.92%	4.04%	<b>4.44%</b>	3.82%
asking with C	4.30%	4.79%	80.54%	5.11%	<b>5.26%</b>
asking with D	4.35%	5.07%	5.41%	79.09%	<b>6.08%</b>
asking with E	4.57%	5.07%	<b>5.85%</b>	5.68%	78.83%

$$\begin{array}{l}
 A \prec C \\
 B \prec D \\
 *C \prec E \\
 D \prec E \\
 *E \prec C
 \end{array}
 \Rightarrow
 \begin{array}{l}
 A \prec C \\
 B \prec D \prec E
 \end{array}$$
  

$\mathbb{R}^8$	$f_A$	$f_B$	$f_C$	$f_D$	$f_E$
asking with A	95.33%	0.88%	1.26%	1.20%	<b>1.30%</b>
asking with B	1.34%	92.27%	1.72%	<b>2.93%</b>	1.74%
asking with C	2.15%	2.28%	88.99%	2.35%	<b>4.23%</b>
asking with D	1.93%	<b>3.75%</b>	2.82%	88.19%	3.31%
asking with E	2.60%	2.36%	<b>5.09%</b>	3.32%	86.63%

$$\begin{array}{l}
 A \prec E \\
 *B \prec D \\
 *C \prec E \\
 *D \prec B \\
 *E \prec C
 \end{array}
 \Rightarrow
 \begin{array}{l}
 A \prec E
 \end{array}$$

The algorithm for finding out the position of each letter begins with asking a question

$$\begin{aligned}
 S_{????} \# L_x &= \left\{ \begin{array}{l} S_{????} \otimes (L_x)^* \quad \text{in HRR} \\ (L_x)^+ S_{????} \quad \text{in GA}_c \end{array} \right\} \\
 &= (t^x)' \approx t^x \quad (37)
 \end{aligned}$$

for each letter  $L_x \in \mathcal{L}$ . Next, we need to find the item in the clean-up memory  $\mathcal{C} \setminus \mathcal{L}$  that is most similar to  $(t^x)'$ . Let us denote this item by  $z$ . With high probability,  $z$  is the power of  $t$  associated with the position of the letter  $L_x$  in the sequence  $S_{????}$ , although, if recognized incorrectly,  $z$  will most likely point to some other  $t^{y>x}$ . Now let us ask a second question

$$\begin{aligned}
 S_{????} \# z &= \left\{ \begin{array}{l} S_{????} \otimes z^* \quad \text{in HRR} \\ \langle S_{????} z^+ \rangle_1 \quad \text{in GA}_c \end{array} \right\} \\
 &= L' \approx L_x. \quad (38)
 \end{aligned}$$

We use the projected product in GA<sub>C</sub> because we are looking for a letter vector placed on the position indicated by  $z$ . In HRR the resulting  $L'$  should be compared with letters only. In most cases  $L'$  will point to the correct letter. However, in a small fraction of test results,  $L'$  will point to letters surrounding  $L_x$ , because  $z$  has been mistakenly decoded as  $t^y$  for some  $y \neq x$ . Also, letters preceding  $L_x$  should come up less often than letters proceeding  $L_x$ .

Figure 1 presents test results for HRR. The data in Figure 1 should be interpreted as follows: the first row of each table next to a graph contains the vector lengths of the data used in 5 consecutive experiments (10000 trials each). The second row contains the number of faulty answers within those 10000

trials. The next 4 rows present the percentage of occurrence of a "faulty" letter within all faulty answers presented in the second row.

Faulty alignments (i.e. those, for which the percentages corresponding to letters do not align increasingly within a single column) have been marked with a "\*" in the table headings. We used  $S_{ABCDE}$  as the mysterious encoded sequence  $S_{????}$ . In each case we crossed out the most frequently occurring letter and we concentrated on the frequency of the remaining letters. In HRR, for sufficiently large vector sizes, the frequencies  $f_L$  of all letters  $L \in \mathcal{L}$  aligned correctly

$$f_B < f_C < f_D < f_E \quad \text{asking with A,} \quad (39)$$

$$f_A < f_C < f_D < f_E \quad \text{asking with B,} \quad (40)$$

$$f_A < f_B < f_D < f_E \quad \text{asking with C,} \quad (41)$$

$$f_A < f_B < f_C < f_E \quad \text{asking with D,} \quad (42)$$

$$f_A < f_B < f_C < f_D \quad \text{asking with E.} \quad (43)$$

It was straightforward that these inequalities lead to  $f_A < f_B < f_C < f_D < f_E$  and correctly identify the encoded sequence as  $S_{ABCDE}$ . Test results are less accurate when we asked about letters lying closer to the end of a sequence, therefore the size of the vector should be adequately long. Moreover, the longer the vector, the larger the difference between the frequencies.

GA<sub>C</sub> was expected to perform worse in this experiment, because we can construct powers of a multivector  $t^{i-1}$  by

multiplying it with  $t$  from one side only. Indeed, at the first glance Table II shows that letter frequencies do not align correctly at all. We therefore needed to slightly modify the algorithm for finding letter alignment in  $GA_c$ : we concentrated on two largest frequencies in each series of asking questions—the largest frequency represents the letter  $L$  that was used to ask the question and the second largest frequency indicates letter  $\hat{L}$  that most likely proceeds letter  $L$ .

Table II presents the frequencies of letters recognized as the most probable answer to Equation (38), the second largest frequency in each row is printed in bold. Partial letter alignments have been placed next to each table and contradictory alignments have been preceded with a “\*”. When being asked with the last letter of the sequence, HRR provided less accurate answers and so did  $GA_c$  by yielding more contradictions than in case of previous letters. It is impossible to avoid contradictory alignments in  $GA_c$  because we do not know which letter is the last one and the algorithm for recovering letter alignment in  $GA_c$  instructs us to write down the partial alignment with that letter being preceded by some other letter. The remaining alignments point correctly to the sequence  $S_{ABCDE}$

$$\left. \begin{array}{l} A \prec B \\ C \prec D \prec E \\ A \prec B \prec C \prec D \\ A \prec C \\ B \prec D \prec E \\ A \prec E \end{array} \right\} \Rightarrow A \prec B \prec C \prec D \prec E. \quad (44)$$

## VI. CONCLUSION

We have shown that multivector powers in  $GAC$  have properties similar to convolutive powers of HRR vectors

- (multi)vectors  $t^{i-r}$  and  $t^i$  are similar in much the same way as  $t^i$  and  $t^{i+r}$ ,
- items placed near the beginning of a sequence are remembered more prominently and thus, are recognized correctly more often,
- items placed near the end of a sequence are remembered less precisely and often come up as the most probable answer when the correct item is not recognized.

We have used the last two properties to find the alignment of sequence items without the explicit knowledge of (multi)vector powers. While HRR retrieved the original alignment without greater problems,  $GA_c$  left us with an easily soluble logical puzzle providing fragmentary alignments.

These properties can be used to build holographic lexicons, dictionaries and other structures that require storing order information and word meaning in the same pattern.

## ACKNOWLEDGMENT

This work was supported by grant G.0405.08 of the Research Programme of the Research Foundation-Flanders (FWO, Belgium)

## REFERENCES

- [1] D. Aerts and M. Czachor, “Quantum aspects of semantic analysis and symbolic artificial intelligence”, *J. Phys. A*, vol. 37, pp. L123-L13, 2004.
- [2] D. Aerts and M. Czachor, “Cartoon computation: Quantum-like algorithms without quantum mechanics”, *J. Phys. A*, vol. 40, pp. F259-F266, 2007.
- [3] M. Czachor, “Elementary gates for cartoon computation”, *J. Phys. A*, vol. 40, pp. F753-F759, 2007.
- [4] D. Aerts and M. Czachor, “Tensor-product versus geometric-product coding”, *Physical Review A*, vol. 77, id. 012316, 2008.
- [5] D. Aerts, M. Czachor, and B. De Moor, “Geometric Analogue of Holographic Reduced Representation”, *J. Math. Psychology*, vol. 53, pp. 389-398, 2009.
- [6] D. Aerts, M. Czachor, and B. De Moor, “On geometric-algebra representation of binary spatter codes”. preprint arXiv:cs/0610075 [cs.AI], 2006.
- [7] D. Aerts, M. Czachor, and Ł. Orłowski, “Teleportation of geometric structures in 3D”, *J. Phys. A* vol. 42, 135307, 2009.
- [8] W.K. Clifford, “Applications of Grassmann’s extensive algebra”, *American Journal of Mathematics Pure and Applied*, vol. 1, 350–358, 1878.
- [9] R. W. Gayler, “Multiplicative binding, representation operators, and analogy”, *Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences*, K. Holyak, D. Gentner, and B. Kokinov, eds., Sofia, Bulgaria: New Bulgarian University, p. 405, 1998.
- [10] S. Deerwester et al. “Indexing by Latent Semantic Analysis”, *Journal of American Society for Information Science*, vol. 41, 391, 1990.
- [11] H. Grassmann, “Der Ort der Hamilton’schen Quaternionen in der Ausdehnungslehre”, *Mathematische Annalen*, vol. 3, 375–386, 1877.
- [12] M.N. Jones & D.J.K. Mewhort, “Representing Word Meaning and Order Information in a Composite Holographic Lexicon”, *Psychological Review*, vol. 114, No. 1, pp. 1-37, 2007.
- [13] P. Kanerva, “Binary spatter codes of ordered k-tuples”. In C. von der Malsburg et al. (Eds.), *Artificial Neural Networks ICANN Proceedings, Lecture Notes in Computer Science* vol. 1112, pp. 869-873, 1996.
- [14] P. Kanerva, “Fully distributed representation”. *Proc. 1997 Real World Computing Symposium (RWC97, Tokyo)*, pp. 358-365, 1997.
- [15] N.G. Marchuk, and D.S. Shirokov, “Unitary spaces on Clifford algebras”, *Advances in Applied Clifford Algebras*, vol 18, pp. 237-254, 2008.
- [16] A. Patyk, “Geometric Algebra Model of Distributed Representations”, in *Geometric Algebra Computing in Engineering and Computer Science*, E. Bayro-Corrochano and G. Scheuermann, eds. Berlin: Springer, 2010. Preprint arXiv:1003.5899v1 [cs.AI].
- [17] T. Plate, “Holographic Reduced Representations”, *IEEE Trans. Neural Networks*, vol. 6, no. 3, pp. 623-641, 1995.
- [18] T. Plate, *Holographic Reduced Representation: Distributed Representation for Cognitive Structures*. CSLI Publications, Stanford, 2003.
- [19] P. Smolensky, “Tensor product variable binding and the representation of symbolic structures in connectionist

# A comparison of geometric analogues of Holographic Reduced Representations, original Holographic Reduced Representations and Binary Spatter Codes

Agnieszka Patyk-Łońska, Marek Czachor, and Diederik Aerts

**Abstract**—Geometric Analogues of Holographic Reduced Representations (GA HRR) employ role-filler binding based on geometric products. Atomic objects are real-valued vectors in  $n$ -dimensional Euclidean space and complex statements belong to a hierarchy of multivectors. The paper reports a battery of tests aimed at comparison of GA HRR with Holographic Reduced Representation (HRR) and Binary Spatter Codes (BSC). Firstly, we perform a test of GA HRR which is analogous to the one proposed by Plate in [13]. Plate’s simulation involved several thousand 512-dimensional vectors stored in clean-up memory. The purpose was to study efficiency of HRR but also to provide a counterexample to claims that role-filler representations do not permit one component of a relation to be retrieved given the others. We repeat Plate’s test on a continuous version of GA HRR –  $GA_c$  (as opposed to its discrete version described in [12]) and compare the results with the original HRR and BSC. The object of the test is to construct statements concerning multiplication and addition. For example, “ $2 \cdot 3 = 6$ ” is constructed as  $times_{2,3} = times + operand * (num_2 + num_3) + result * num_6$ . To look up this vector one then constructs a similar statement with one of the components missing and checks whether it points correctly to  $times_{2,3}$ . We concentrate on comparison of recognition percentage for the three models for comparable data size, rather than on the time taken to achieve high percentage. Results show that the best models for storing and recognizing multiple similar statements are  $GA_c$  and Binary Spatter Codes with recognition percentage highly above 90.

**Index Terms**—distributed representations, geometric algebra, HRR, BSC, scaling.

## I. INTRODUCTION

**H**OLOGRAPHIC Reduced Representations (HRR) and Binary Spatter Codes (BSC) are distributed representations of cognitive structures where binding of role-filler codevectors maintains predetermined data size. In HRR [13] binding is performed by means of circular convolution

$$(x \otimes y)_j = \sum_{k=0}^{n-1} x_k y_{j-k \bmod n}. \quad (1)$$

of real  $n$ -tuples or, in ‘frequency domain’, by componentwise multiplication of (complex)  $n$ -tuples,

$$(x_1, \dots, x_n) \otimes (y_1, \dots, y_n) = (x_1 y_1, \dots, x_n y_n). \quad (2)$$

Bound  $n$ -tuples are superposed by addition, and unbinding is performed by an approximate inverse. A dual formalism, where real data are bound by componentwise multiplication,

was discussed by Gayler [6]. In BSC [8], [9] one works with binary  $n$ -tuples, bound by componentwise addition mod 2,

$$\begin{aligned} (x_1, \dots, x_n) \oplus (y_1, \dots, y_n) &= (x_1 \oplus y_1, \dots, x_n \oplus y_n), \\ x_j \oplus y_j &= x_j + y_j \bmod 2, \end{aligned} \quad (3)$$

and superposed by pointwise majority-rule addition; unbinding is performed by the same operation as binding.

One often reads that the above models represent data by *vectors*, which is not exactly true. Given two vectors one does not know how to perform, say, their convolution or componentwise multiplication since the result depends on basis that defines the components. Basis must be fixed in advance since otherwise all the above operations become ambiguous. It follows that neither of the above reduced representations can be given a true and meaningful geometric interpretation. Geometric Analogues of Holographic Reduced Representations (GA HRR) [2] can be constructed if one defines binding by the geometric product, a notion introduced in 19th century works of Grassmann [7] and Clifford [5].

The fact that GA HRR is intrinsically geometric may be important for various conceptual reasons — for example, the rules of geometric algebra may be regarded as a mathematical formalization of the process of *understanding* geometry. The use of geometric algebra in distributed representations has been inspired by a well-known fact, that most people think in pictures, i.e. two- and three-dimensional shapes, not by using sequences of ones and zeroes.

In order to grasp the main ideas behind GA HRR let us consider an orthonormal basis  $b_1, \dots, b_n$  in some  $n$ -dimensional Euclidean space. Now consider two vectors  $x = \sum_{k=1}^n x_k b_k$  and  $y = \sum_{k=1}^n y_k b_k$ . The *scalar*

$$x \cdot y = y \cdot x \quad (4)$$

is known as the *inner product*. The *bivector*

$$x \wedge y = -y \wedge x \quad (5)$$

is the *outer product* and may be regarded as an oriented plane segment (alternative interpretations are also possible, cf. [4]).  $\mathbf{1}$  is the identity of the algebra. The geometric product of  $x$

and  $y$  then reads

$$xy = \underbrace{\sum_{k=1}^n x_k y_k}_{x \cdot y} \mathbf{1} + \underbrace{\sum_{k < l} (x_k y_l - y_k x_l)}_{x \wedge y} b_k b_l. \quad (6)$$

Grassmann and Clifford introduced geometric product by means of the basis-independent formula involving the *multivector*

$$xy = x \cdot y + x \wedge y \quad (7)$$

which implies the so-called Clifford algebra

$$b_k b_l + b_l b_k = 2\delta_{kl} \mathbf{1}. \quad (8)$$

when restricted to an orthonormal basis. Inner and outer product can be defined directly from  $xy$ :

$$x \cdot y = \frac{1}{2}(xy + yx), \quad x \wedge y = \frac{1}{2}(xy - yx).$$

The most ingenious element of (7) is that it adds two apparently different objects, a scalar and a plane element, an operation analogous to addition of real and imaginary parts of a complex number. Geometric product for vectors  $x, y, z$  can be axiomatically defined by the following rules:

$$\begin{aligned} (xy)z &= x(yz), \\ x(y+z) &= xy + xz, \\ (x+y)z &= xz + yz, \\ xx &= x^2 = |x|^2, \end{aligned}$$

where  $|x|$  is a positive scalar called the magnitude of  $x$ . The rules imply that  $x \cdot y$  must be a scalar since

$$xy + yx = |x + y|^2 - |x|^2 - |y|^2.$$

Geometric algebra allows us to speak of inverses of vectors:  $x^{-1} = x/|x|^2$ .  $x$  is invertible (i.e. possesses an inverse) if its magnitude is nonzero. Geometric product of an arbitrary number of invertible vectors is also invertible. The possibility of inverting all nonzero-magnitude vectors is perhaps the most important difference between geometric and convolution algebras.

Geometric products of *different* basis vectors

$$b_{k_1 \dots k_j} = b_{k_1} \dots b_{k_j},$$

$k_1 < \dots < k_j$ , are called basis blades (or just blades). In  $n$ -dimensional Euclidean space there are  $2^n$  different blades. This can be seen as follows. Let  $\{x_1, \dots, x_n\}$  be a sequence of bits. Blades in an  $n$ -dimensional space can be written as

$$c_{x_1 \dots x_n} = b_1^{x_1} \dots b_n^{x_n}$$

where  $b_k^0 = \mathbf{1}$ , which shows that blades are in a one-to-one relation with  $n$ -bit numbers. A general multivector is a linear combination of blades,

$$\psi = \sum_{x_1 \dots x_n=0}^1 \psi_{x_1 \dots x_n} c_{x_1 \dots x_n}, \quad (9)$$

with real or complex coefficients  $\psi_{x_1 \dots x_n}$ . Clifford algebra implies that

$$c_{x_1 \dots x_n} c_{y_1 \dots y_n} = (-1)^{\sum_{k < l} y_k x_l} c_{(x_1 \dots x_n) \oplus (y_1 \dots y_n)}, \quad (10)$$

where  $\oplus$  is given by (3). Multiplication of two basis blades is thus, up to a sign, in a one-to-one relation with exclusive alternative of two binary  $n$ -tuples. Accordingly, (10) is a projective representation of the group of binary  $n$ -tuples with addition modulo 2.

GA HRR is based on binding defined by geometric product (10) of blades while superposition is just addition of blades (9). The discrete  $GA_d$  is a version of GA HRR obtained if  $\psi_{x_1 \dots x_n}$  in (9) equal  $\pm 1$ . The first recognition tests of  $GA_d$ , as compared to HRR and BSC, were described in [12]. In the present paper we go further and compare HRR and BSC with  $GA_c$ , a version of GA HRR employing “projected products” [2] and arbitrary real  $\psi_{x_1 \dots x_n}$ . We also repeat Plate’s scaling test ([13], Appendix I) and compare test results for  $GA_c$ , HRR and BSC models.

Throughout this paper we shall use the following notation: “\*” denotes binding roles and fillers by means of the geometric product and “+” denotes the superposition of sentence chunks, e.g.

$$“Fido bit Pat” = bite_{agt} * Fido + bite_{obj} * Pat. \quad (11)$$

Additionally, “ $\otimes$ ” will denote binding performed by circular convolution used in the HRR model and  $a^*$  denotes the involution of a HRR vector  $a$ . A “+” in the superscript of  $x^+$  denotes the operation of reversing a blade or a multivector  $x: (b_{k_1 \dots k_j})^+ = b_{k_j} \dots b_{k_1}$ . Asking a question will be denoted with “ $\sharp$ ”, as in

$$\begin{aligned} “Who bit Pat?” \\ &= (bite_{agt} * Fido + bite_{obj} * Pat) \sharp bite_{agt} \quad (12) \\ &\approx Fido. \end{aligned}$$

The *size* of a (multi)vector means the number of memory cells it occupies in computer’s memory, while the *magnitude* of a (multi)vector  $V = \{v_1, \dots, v_n\}$  is its Euclidean norm  $\sqrt{\sum_{i=1}^n v_i^2}$ .

For our purposes it is important that geometric calculus allows us to define in a very systematic fashion a hierarchy of associative, non-commutative, and invertible operations that can be performed on  $2^n$ -tuples. The resulting superpositions are less noisy than the ones based on convolutions, say. Such operations are in general unknown to a wider audience, which explains popularity of tensor and convolution algebras. Geometric product preserves dimensionality at the level  $2^n$ -dimensional *multivectors*, where  $n$  is the number of bits indexing basis vectors. Moreover, all nonzero vectors are invertible with respect to geometric product, a property absent for convolutions and important for unbinding and recognition. A detailed analysis of links between GA HRR, HRR and BSC can be found in [2]. In particular, it is shown that both GA HRR and BSC are based on two different representations (in

group theoretical sense) of the additive group of binary  $n$ -tuples with addition modulo 2. Actually, the latter observation was the starting point for studying geometric algebra forms of reduced representations [3].

## II. THE $GA_c$ MODEL

Multivector (9) associated with  $n$ -dimensional Euclidean space can be represented by the  $2^n$ -tuple  $(\psi_{0_1\dots 0_n}, \dots, \psi_{1_1\dots 1_n})$ . Geometric product of two such  $2^n$ -tuples is again a  $2^n$ -tuple. In this sense geometric product is analogous to bindings employed in HRR or BSC, but we can still proceed in several inequivalent ways. For example, since a product of two basis blades is again a basis blade multiplied by  $\pm 1$ , we can require that  $\psi_{x_1\dots x_n} = \pm 1$ . Such a discrete version of GA HRR was tested vs. HRR and BSC in [12], and will be denoted here by  $GA_d$  (discrete GA HRR).

The continuous  $GA_c$  model differs greatly from  $GA_d$ . First of all, we do not begin with a general  $2^n$ -dimensional multivector. Atomic objects are real-valued vectors in  $n$ -dimensional Euclidean space, in practice represented by  $n$ -tuples of components taken in some basis. A hierarchy of multivectors is reserved for complex statements, formed by binding and superposition of atomic objects. An  $n$ -dimensional vector, when seen from the multivector perspective, is a highly sparse  $2^n$ -tuple: Only  $n$  out of  $2^n$  components can be nonzero.

The procedure we employ was suggested in [2]. The space of  $2^n$ -tuples is split into subspaces corresponding to scalars (0-vectors), vectors (1-vectors), bivectors (2-vectors), and so on. At the bottom of the hierarchy lay vectors  $V \in \mathbb{R}^n$ , having rank 1 and being denoted as  $\overset{1}{V}$ . An object of rank 2 is created by multiplying two elements of rank 1 with the help of the geometric product. Let  $\overset{1}{V} = \{\alpha_1, \alpha_2, \alpha_3\}$  and  $\overset{1}{W} = \{\beta_1, \beta_2, \beta_3\}$  be vectors in  $\mathbb{R}^3$ . A multivector  $\overset{2}{X}$  of rank 2 in  $\mathbb{R}^3$  comprises the following elements (cf. [10])

$$\overset{2}{X} = \overset{1}{V} \overset{1}{W} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} \alpha_1\beta_1 + \alpha_2\beta_2 + \alpha_3\beta_3 \\ \alpha_1\beta_2 - \alpha_2\beta_1 \\ \alpha_1\beta_3 - \alpha_3\beta_1 \\ \alpha_2\beta_3 - \alpha_3\beta_2 \end{bmatrix}, \quad (13)$$

the first entry in the array on the right being a scalar and the remaining three entries being 2-blades. For arbitrary vectors in  $\mathbb{R}^n$  we would have obtained one scalar (or, more conveniently:  $\binom{n}{0}$  scalars) and  $\binom{n}{2}$  2-blades.

Let  $\overset{2}{X} = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$  and  $\overset{1}{V} = \{\alpha_1, \alpha_2, \alpha_3\}$  be two multivectors in  $\mathbb{R}^3$ . A multivector  $\overset{3}{Z}$  of rank 3 in  $\mathbb{R}^3$  may be created in two ways: as a result of multiplying either  $\overset{1}{V}$  by  $\overset{2}{X}$  or  $\overset{2}{X}$  by  $\overset{1}{V}$ . Let us concentrate on the first case

$$\overset{3}{Z} = \overset{1}{V} \overset{2}{X} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \gamma_4 \end{bmatrix} = \begin{bmatrix} \alpha_1\gamma_1 - \alpha_2\gamma_2 - \alpha_3\gamma_3 \\ \alpha_1\gamma_2 + \alpha_2\gamma_1 - \alpha_3\gamma_4 \\ \alpha_1\gamma_3 + \alpha_2\gamma_4 + \alpha_3\gamma_1 \\ \alpha_1\gamma_4 - \alpha_2\gamma_3 + \alpha_3\gamma_2 \end{bmatrix}. \quad (14)$$

Here, the first three entries in the resulting matrix are 1-blades, while the last entry is a 3-blade. For arbitrary multivectors of

rank 1 and 2 in  $\mathbb{R}^n$  we would have obtained  $\binom{n}{1}$  vectors and  $\binom{n}{3}$  trivectors. We cannot generate multivectors of rank higher than 3 in  $\mathbb{R}^3$ , but it is easy to check that in spaces  $\mathbb{R}^{n>3}$  a multivector of rank 4 would have  $\binom{n}{0}$  scalars,  $\binom{n}{2}$  bivectors and  $\binom{n}{4}$  4-blades. The number of  $k$ -blades in a multivector of rank  $r$  is described by Table I. It becomes clear that a multivector of rank  $r$  over  $\mathbb{R}^n$  is actually a vector over a  $\sum_{i=0}^{\lfloor \frac{r}{2} \rfloor} \binom{n}{2i+r \bmod 2}$ -dimensional space.

As an example let us consider the following roles and fillers being normalized vectors drawn randomly from  $\mathbb{R}^n$  with Gaussian distribution  $N(0, \frac{1}{n})$

$$\begin{aligned} Pat &= \{a_1, \dots, a_n\}, & name &= \{x_1, \dots, x_n\}, \\ male &= \{b_1, \dots, b_n\}, & sex &= \{y_1, \dots, y_n\}, \\ 66 &= \{c_1, \dots, c_n\}, & age &= \{z_1, \dots, z_n\}. \end{aligned} \quad (15)$$

$PSmith$ , who is a 66 year old male named Pat, is created by first multiplying roles and fillers with the help of the geometric product

$$\begin{aligned} PSmith &= \\ &= name * Pat + sex * male + age * 66 \\ &= name \cdot Pat + name \wedge Pat + sex \cdot male + \\ &\quad sex \wedge male + age \cdot 66 + age \wedge 66 \end{aligned} \quad (16)$$

$$\begin{aligned} &= \begin{bmatrix} \sum_{i=1}^n (a_i x_i + b_i y_i + c_i z_i) \\ a_1 x_2 - a_2 x_1 + b_1 y_2 - b_2 y_1 + c_1 z_2 - c_2 z_1 \\ a_1 x_3 - a_3 x_1 + b_1 y_3 - b_3 y_1 + c_1 z_3 - c_3 z_1 \\ \vdots \\ a_{n-1} x_n - a_n x_{n-1} + b_{n-2} y_n - b_n y_{n-1} + c_{n-1} z_n - c_n z_{n-1} \end{bmatrix} \\ &= [d_0, d_{12}, d_{13}, \dots, d_{(n-1)n}]^T \\ &= d_0 + d_{12}e_{12} + d_{13}e_{13} + \dots + d_{(n-1)n}e_{(n-1)n}, \end{aligned} \quad (17)$$

where  $e_1, \dots, e_n$  are orthonormal basis blades. In order to be decoded as much correctly as possible,  $PSmith$  should have the same magnitude as vectors representing atomic objects, therefore it needs to be normalized. Finally,  $PSmith$  takes the form of

$$PSmith = [\hat{d}_0, \hat{d}_{12}, \hat{d}_{13}, \dots, \hat{d}_{(n-1)n}]^T, \quad (18)$$

where  $\hat{d}_i = \frac{d_i}{\sqrt{\sum_{j=0,12}^{(n-1)n} d_j^2}}$ .

$PSmith$  is now a multivector of rank 2. The decoding operation

$$\begin{aligned} name^+ PSmith &= \\ &= name^+ (name \cdot Pat + name \wedge Pat + sex \cdot male \\ &\quad + sex \wedge male + age \cdot 66 + age \wedge 66) \end{aligned} \quad (19)$$

will produce a multivector of rank 3 consisting of vectors and trivectors. However, the original  $Pat$  did not contain any trivector components — they all belong to the noise part and the only interesting blades in  $name^+ PSmith$  are vectors. The expected answer is a vector, therefore there is no point in



TABLE I  
NUMBERS OF  $k$ -BLADES IN MULTIVECTORS OF VARIOUS RANKS IN  $\mathbb{R}^n$

rank	scalars	vectors	bivectors	trivectors	4-blades	...	data size
1	0	$\binom{n}{1}$	0	0	0	...	$\mathcal{O}\left(\binom{n}{1}\right)$
2	$\binom{n}{0}$	0	$\binom{n}{2}$	0	0	...	$\mathcal{O}\left(\binom{n}{0} + \binom{n}{2}\right)$
3	0	$\binom{n}{1}$	0	$\binom{n}{3}$	0	...	$\mathcal{O}\left(\binom{n}{1} + \binom{n}{3}\right)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$2r$	$\binom{n}{0}$	0	$\binom{n}{2}$	0	$\binom{n}{4}$	...	$\mathcal{O}\left(\sum_{i=0}^r \binom{n}{2i}\right)$
$2r+1$	0	$\binom{n}{1}$	0	$\binom{n}{3}$	0	...	$\mathcal{O}\left(\sum_{i=0}^r \binom{n}{2i+1}\right)$

calculating the whole multivector  $name^+PSmith$  and only then comparing it with items stored in the clean-up memory. To be efficient, one should generate only the vector-part while computing  $name^+PSmith$  and skip the noisy trivectors.

Let  $\langle \cdot \rangle_k$  denote the projection of a multivector on  $k$ -blades. To decode  $PSmith$ 's name we need to compute

$$\begin{aligned} & \langle name^+PSmith \rangle_1 \\ &= name^+namePat + \langle name^+(name \wedge Pat \\ & \quad + sex \cdot male + sex \wedge male + age \cdot 66 + age \wedge 66) \rangle_1 \\ &= Pat + noise = Pat'. \end{aligned} \quad (20)$$

The resulting  $Pat'$  will still be noisy, but to a lesser degree than it would have been if the trivectors were present.

Formally, we are using a map  $*_{1,2}^1$  that transforms a multivector of rank 1 (i.e. an  $n$ -tuple) and a multivector of rank 2 (i.e. a  $(1 + \frac{(n-1)n}{2})$ -tuple) into a multivector of rank 1 without computing the unnecessary blades. Let  $X$  be a multivector of rank 2

$$X = \langle X \rangle_0 + \langle X \rangle_2 = x_0 + \sum_{l < m} x_{lm} e_l e_m, \quad (21)$$

where  $x_{lm} = -x_{ml}$ . If  $A = (A_1, \dots, A_n)$  is a decoding vector (actually, an inverse of a role vector), then

$$\begin{aligned} A *_{1,2}^1 X &= x_0 A + \sum_{l,m} A_l x_{lm} e_m \\ &= \sum_k (x A_k + \sum_l A_l x_{lk}) e_k \\ &= \sum_k Y_k e_k = Y, \end{aligned} \quad (22)$$

with  $Y = (Y_1, \dots, Y_n)$  being an  $n$ -tuple, i.e. a multivector of rank 1. More explicitly,

$$Y_k = (A *_{1,2}^1 X)_k = x_0 A_k + \sum_{l=1}^{k-1} A_l x_{lk} - \sum_{l=k+1}^n A_l x_{kl}. \quad (23)$$

The map  $*_{1,2}^1$  is an example of a *projected product*, introduced in [2], reconstructing the vector part of  $AX$  without computing the unnecessary parts. The projected product is basis independent, as opposed to circular convolutions. In general,  $*_{l,k}^m$  transforms the geometric product of two multivectors  $A$  and  $B$  into a multivector  $C$ .

We now need to compare  $Pat'$  with other items stored in the clean-up memory using the dot product, and since  $Pat'$  is a vector, we need to compare only the vector part. That means, if the clean-up memory contained a multivector  $M$  of an odd rank, we would also need to compute  $Pat' \cdot \langle M \rangle_1$  while searching for the right answer.

This method of decoding suggests that items stored in the clean-up memory should hold information about their ranks, which is dangerously close to employing fixed data slots present in localist architectures. However, a rank of a clean-up memory item can be "guessed" from its size. In a distributed model we also should not "know" for sure how many parts the projected product should reject, but it can certainly reject parts spanned by blades of highest grades. Unfortunately, since the geometric product is non-commutative, questions concerning roles and fillers need to be asked on different sides of a sentence, forcing atomic objects to hold information on whether they are roles or fillers and thus, forcing them to be partly hand-generated. We can either ask question always on the same side of a sentence and be satisfied with less precise answers or always ask about only the roles or only the fillers. It becomes clear, that recognition based on the hierarchy of multivectors and the projected product is best applicable to tasks in which questions need to be asked only on one side of the sentence or in which sentences have predetermined structure.

Before providing formulas for encoding and decoding a complex statement we need to introduce additional notation for the projected product and the projection. We have already introduced the projected product  $*_{l,k}^m$  transforming the geometric product of two multivectors of ranks  $l$  and  $k$  into a multivector of rank  $m$ . This will not always be the case for complex statements, since we can produce a multivector that will not be of any given rank. Let  $*_{l,\{\alpha_1, \alpha_2, \dots, \alpha_k\}}^m$  denote the projected product transforming the geometric product of a multivector  $A$  and a multivector  $B$  containing  $\alpha_1$ -blades,  $\alpha_2$ -blades, ... and  $\alpha_k$ -blades into a multivector  $C$ . In this way, the projected product  $*_{1,2}^1$  may be written down as  $*_{1,\{0,2\}}^1$ . By analogy, let  $\langle \cdot \rangle_{\{\alpha_1, \alpha_2, \dots, \alpha_k\}}$  denote the projection of a multivector on components spanned by  $\alpha_1$ -blades,  $\alpha_2$ -blades, ... and  $\alpha_m$ -blades.

Let  $\Psi$  denote the normalized multivector encoding the sentence “*Fido bit PSmith*”, i.e.

$$\Psi = \underbrace{\text{bite}_{agt} * \text{Fido}}_{\text{rank 2}} + \underbrace{\text{bite}_{obj} * \text{PSmith}}_{\text{rank 2}}. \quad (24)$$

Multivector  $\Psi$  will contain scalars, vectors, bivectors and trivectors and can be written down as the following vector of dimension  $\sum_{i=0}^3 \binom{n}{i}$

$$\Psi = \underbrace{\alpha}_{\text{a scalar}} + \underbrace{\sum_{i=1}^n \beta_i e_i}_{\text{vectors}} + \underbrace{\sum_{1 \leq i < j} \gamma_{ij} e_{ij}}_{\text{bivectors}} + \underbrace{\sum_{1 \leq i < j < k} \delta_{ijk} e_{ijk}}_{\text{trivectors}} \quad (25)$$

The following example illustrates how to ask questions in the  $GA_c$  architecture.

“*Who was bitten?*”

The answer to that question will be a multivector of rank 2

$$\begin{aligned} \Psi \# \text{bite}_{obj} &= \langle \text{bite}_{obj}^+ \Psi \rangle_{\{0,2\}} = \text{bite}_{obj}^+ *_{1,\{0,1,2,3\}}^2 \Psi \\ &= \text{PSmith}' \approx \text{PSmith}. \end{aligned} \quad (26)$$

Let  $\text{bite}_{obj} = \{y_1, \dots, y_n\}$ ,  $\text{PSmith}'$  will then have the form

$$\begin{aligned} \text{PSmith}' &= (y_1 e_1 + \dots + y_n e_n) *_{1,\{0,1,2,3\}}^2 \\ &= \left( \sum_{i=1}^n \beta_i e_i + \sum_{1 \leq i < j < k} \delta_{ijk} e_{ijk} \right) \end{aligned} \quad (27)$$

$$= \underbrace{\sum_{k=1}^n y_k \beta_k}_{\text{a scalar}} + \underbrace{\sum_{1 \leq i < j} \theta_{ij} e_{ij}}_{\text{bivectors}}, \quad (28)$$

where

$$\theta_{ij} = y_i \beta_j - y_j \beta_i + \sum_{\substack{t=1 \\ t \notin \{i,j\}}}^n y_t \delta_{ijt} \quad (29)$$

with  $\delta_{ijt} = \delta_{tij} = -\delta_{itj}$ . As previously,  $\text{PSmith}'$  should be compared with appropriate items from the clean-up memory to produce the most probable answer.

### III. OVERVIEW OF PLATE’S SCALING TEST

Plate [13] describes a simulation in which approximately 5000 HRR 512-dimensional vectors were stored in the clean-up memory. The purpose of his simulation was to study efficiency of the HRR model but also to provide a counterexample to the claim that role-filler representations do not permit one component of a relation to be retrieved given the others. We will repeat Plate’s test on several models and compare the results.

Let us consider the following atomic objects

$$\left. \begin{array}{l} \text{num}_x \text{ (for } x = 0, \dots, 2500), \\ \text{times}, \\ \text{plus}, \end{array} \right\} \text{ fillers}, \quad (30)$$

$$\left. \begin{array}{l} \text{result}, \\ \text{operand}. \end{array} \right\} \text{ roles} \quad (31)$$

At the beginning of the scaling test, relations concerning multiplication and addition are constructed. For example, “ $2 \cdot 3 = 6$ ” is constructed as

$$\text{times}_{2,3} = \text{times} + \text{operand} * (\text{num}_2 + \text{num}_3) + \text{result} * \text{num}_6. \quad (32)$$

Generally, relations are constructed in the following way

$$\begin{aligned} \text{times}_{x,y} &= \text{times} + \text{operand} * (\text{num}_x + \text{num}_y) \\ &\quad + \text{result} * \text{num}_{x \cdot y}, \end{aligned} \quad (33)$$

$$\begin{aligned} \text{plus}_{x,y} &= \text{plus} + \text{operand} * (\text{num}_x + \text{num}_y) \\ &\quad + \text{result} * \text{num}_{x+y}. \end{aligned} \quad (34)$$

$x$  and  $y$  range from 0 to 50 with  $y \leq x$  making a total of 2501 number vectors and 2652 instances of each  $\text{times}_{x,y}$  and  $\text{plus}_{x,y}$ . As one can notice, the same *operand* role is used for both  $x$  and  $y$  to preserve commutativity of multiplication and addition.

Plate writes, that a relation can be “looked up” by supplying enough information to distinguish a specific relation from others. For example, to look up “ $2 \cdot 3 = 6$ ” one needs to find the most similar relation  $R$  to any of the following fragmentary statements

$$\begin{aligned} \text{(case 1)} \quad &\text{times} + \text{operand} * \text{num}_2 \\ &\quad + \text{operand} * \text{num}_3, \end{aligned} \quad (35)$$

$$\begin{aligned} \text{(case 2)} \quad &\text{times} + \text{operand} * \text{num}_2 \\ &\quad + \text{result} * \text{num}_6, \end{aligned} \quad (36)$$

$$\begin{aligned} \text{(case 3)} \quad &\text{times} + \text{operand} * \text{num}_3 \\ &\quad + \text{result} * \text{num}_6, \end{aligned} \quad (37)$$

$$\begin{aligned} \text{(case 4)} \quad &\text{operand} * \text{num}_2 + \text{operand} * \text{num}_3 \\ &\quad + \text{result} * \text{num}_6. \end{aligned} \quad (38)$$

Retrieving the missing piece of information in the first three cases can be done by asking any of the subquestions

$$\text{(case 1)} \quad R \# \text{result}, \quad (39)$$

$$\text{(case 2)} \quad R \# \text{operand}, \quad (40)$$

$$\text{(case 3)} \quad R \# \text{operand}. \quad (41)$$

Case 4 is somewhat more problematic — to look up a missing relation name (*times* or *plus*) one needs to have a separate clean-up memory containing only relation names or to use an alternative encoding in which there is a role for relation names. We will alter Plate’s test by using the latter method.

Plate states that he had tried one run of the system making a query for each component missing in every relation — this amounted to 10608 queries. A further 7956 queries had been made to decode the missing component except for the relation name. Plate goes on to claim, that the system made no errors.

There appear to be two misstatements in Plate’s claims. Firstly, we cannot treat subquestions regarding cases 2 and 3 separately, as there are two equally probable answers to each of these subquestions, provided that relations  $R_2$  and  $R_3$  point

correctly to  $times_{x,y}$ . Secondly, consider a fragmentary piece of information

$$times + operand * num_0 + result * num_0. \quad (42)$$

In this situation, the missing component can be any of the numbers  $num_x$  where  $x \in \{0, \dots, 50\}$  and thus, there are 51 atomic objects that are equally probable to be the right answer. This suggests that Plate regards several answers as valid ones, as long as the similarity of these answers exceeds some threshold. To work out the missing component, one then needs to check which of those potential answers is not in the original set used for retrieval.

Such a method of investigating scaling properties has more than a few advantages:

- Inaccuracies mentioned above act as a test if all atomic objects are created and treated equally. Ideally, every atomic object of the  $num_x$  form should be recognized as a correct answer to the “zero problem” for  $\frac{\text{number of trials}}{51} \cdot 100\%$  of the time.
- Prime numbers greater than 100 do not appear in any of  $times_{x,y}$  and  $plus_{x,y}$  relations, therefore they test if the model is immune to garbage data.
- Numbers ranging from  $num_0$  to  $num_{100}$  may be constructed in a multitude of ways by addition ( $num_0$  by multiplication) and any given sentence chunk  $result * num_z$  will appear quite often in the  $plus_{x,y}$  relation. Hence, this is a great way of checking if the model deals with excessive similarity of a number of complex statements.
- Atomic objects bound with  $operand$  and  $result$  range in variety. On the other hand, there are just two atomic objects acting as an  $operation$  — does it affect in any way the recognition of  $operation$  filler? Indeed, it will be shown in Section V that recognition of the  $operation$  chunk turns out to be quite interesting depending on the choice of the architecture.

#### IV. NOTATION

For the purpose of explaining test results, let us introduce the following notation. Let  $S_{x,y}^*$  and  $S_{x,y}^+$  denote relations

$$S_{x,y}^* = operation * times + operand * (num_x + num_y) + result * num_{x,y}, \quad (43)$$

$$S_{x,y}^+ = operation * plus + operand * (num_x + num_y) + result * num_{x+y}, \quad (44)$$

for  $y \leq x$ . We chose to use a separate role for a relation name to enable encoding the information given only operands and the result. Let  $F_{i,x,y}^{op}$  denote fragmentary statements for  $i \in \{1, 2, 3, 4\}$  and  $op \in \{*, +\}$

$$F_{1,x,y}^{op} = S_{x,y}^{op} - result * num_{x \ op \ y}, \quad (45)$$

$$F_{2,x,y}^{op} = S_{x,y}^{op} - operand * num_x, \quad (46)$$

$$F_{3,x,y}^{op} = S_{x,y}^{op} - operand * num_y, \quad (47)$$

$$F_{4,x,y}^{op} = S_{x,y}^{op} - operation * op. \quad (48)$$

If  $v$  is an element of the clean-up memory, then let  $N(v)$  denote the closest *neighbor* of  $v$ , i.e. an element of the clean-up memory that is most similar to  $v$ . If  $v$  has more than one neighbor, then all subquestions during the test are asked to all of  $v$ 's neighbors. In HRR,  $GA_d$  (with the Hamming measure of similarity) and  $GA_c$  it is extremely unlikely for an element of the clean-up memory to have more than one neighbor due to the continuous nature of data in these architectures. Let  $Q_{i,x,y}^{op} = N(F_{i,x,y}^{op})$  for  $i \in \{1, 2, 3, 4\}$  and  $op \in \{*, +\}$ . During the test we asked subquestions concerning components missing in  $F_{i,x,y}^{op}$  and obtained the following (sets of) answers

$$q_{1,x,y}^{op} = N(Q_{1,x,y}^{op} \# result), \quad (49)$$

$$q_{2,x,y}^{op} = N(Q_{2,x,y}^{op} \# operand), \quad (50)$$

$$q_{3,x,y}^{op} = N(Q_{3,x,y}^{op} \# operand), \quad (51)$$

$$q_{4,x,y}^{op} = N(Q_{4,x,y}^{op} \# operation). \quad (52)$$

We assume that a missing component is identified correctly if it is the only neighbor to appropriate answer  $q_{i,x,y}^{op}$  or it belongs to the set of neighbors of  $q_{i,x,y}^{op}$ .

#### V. TEST RESULTS

The software for all tests was developed by A. Patyk-Łońska in Java language. All tests were performed on an ordinary PC with dualcore AMD processor with 2 GB RAM.

Tables II through IV compare scaling test results for

- $GA_c$  and HRR, both using dot-product as a similarity measure.
- BSC using Hamming distance as a similarity measure.

Although BSC and HRR models need only  $n$ -dimensional vectors, this is not quite the case for and  $GA_c$ , which needs  $1 + \frac{n(n-1)}{2}$  numbers to represent multivectors of rank 2 over  $\mathbb{R}^n$ . We present recognitions test results close to 100% and comment on vector length required for each model to achieve such percentage. The real number of memory cells used up by each model is given in brackets in the table headings.

The answers to subquestions  $Q_{2,x,y}^{op} \# operand$  and  $Q_{3,x,y}^{op} \# operand$  were considered to be correct if any of the two possible operands came up as the item most similar to those subquestions. In case of other questions and subquestions only exact answers were taken into consideration.

50 runs of the test were performed on each model. Unlike in Plate's test,  $x$  and  $y$  ranged from 0 to only 20. Hence, there are 401 number vectors and 462 relation vectors.

The “zero problem” is clearly visible in each tested model, as the recognition percentage of  $Q_{3,x,y}^*$  barely exceeds 90%. Nevertheless,  $Q_{3,x,y}^*$  almost always contains at least one of the operands from the original sentence  $S_{x,y}^*$  since the recognition percentage of  $q_{3,x,y}^*$  reaches 100% for sufficiently large data size. On the whole, the recognition percentage of  $q_{2,x,y}^*$  and  $q_{3,x,y}^*$  does not differ greatly from the recognition percentage of  $q_{2,x,y}^+$  and  $q_{3,x,y}^+$  in any model.

Table entries marked with a “ $\Delta$ ” indicate that despite the wrong recognition of a fragmentary sentence, the missing component has been identified correctly. In all tested models such situations arise for sentences with one of the operands missing.



TABLE II  
RECOGNITION PERCENTAGE FOR GA<sub>c</sub>.

Questions	R <sup>10</sup> (46)	R <sup>20</sup> (191)	R <sup>30</sup> (436)	R <sup>40</sup> (781)
Q <sup>*</sup> <sub>1,x,y</sub>	89.76%	99.98%	99.99%	100.0%
q <sup>*</sup> <sub>1,x,y</sub>	39.44%	95.28%	99.58%	99.88%
Q <sup>*</sup> <sub>2,x,y</sub>	91.12%	99.73%	99.98%	100.0%
q <sup>*</sup> <sub>2,x,y</sub>	36.24%	83.86%	97.92%	99.81%
Q <sup>*</sup> <sub>3,x,y</sub>	83.97%	91.15%	91.33%	91.34%
q <sup>*</sup> <sub>3,x,y</sub>	41.27%	84.92%	98.05% <sup>Δ</sup>	99.82% <sup>Δ</sup>
Q <sup>*</sup> <sub>4,x,y</sub>	98.90%	99.60%	99.63%	99.59%
q <sup>*</sup> <sub>4,x,y</sub>	42.01%	95.56%	99.24%	99.52%
Q <sup>+</sup> <sub>1,x,y</sub>	89.39%	99.99%	100.0%	100.0%
q <sup>+</sup> <sub>1,x,y</sub>	39.09%	95.99%	99.76%	99.95%
Q <sup>+</sup> <sub>2,x,y</sub>	86.96%	99.59%	99.96%	100.0%
q <sup>+</sup> <sub>2,x,y</sub>	35.32%	83.84%	97.97%	99.79%
Q <sup>+</sup> <sub>3,x,y</sub>	87.00%	99.63%	99.96%	100.0%
q <sup>+</sup> <sub>3,x,y</sub>	35.12%	83.84%	97.98%	99.79%
Q <sup>+</sup> <sub>4,x,y</sub>	99.05%	99.53%	99.51%	99.54%
q <sup>+</sup> <sub>4,x,y</sub>	45.84%	94.73%	99.14%	99.49%

TABLE III  
RECOGNITION PERCENTAGE FOR HRR.

Questions	N = 200	N = 300	N = 400	N = 500
Q <sup>*</sup> <sub>1,x,y</sub>	29.1%	27.06%	26.28%	28.51%
q <sup>*</sup> <sub>1,x,y</sub>	31.08% <sup>Δ</sup>	30.03% <sup>Δ</sup>	30.30% <sup>Δ</sup>	32.23% <sup>Δ</sup>
Q <sup>*</sup> <sub>2,x,y</sub>	54.72%	52.06%	53.10%	53.32%
q <sup>*</sup> <sub>2,x,y</sub>	98.99% <sup>Δ</sup>	99.92% <sup>Δ</sup>	99.98% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>*</sup> <sub>3,x,y</sub>	50.53%	47.93%	49.80%	51.21%
q <sup>*</sup> <sub>3,x,y</sub>	98.92% <sup>Δ</sup>	99.90% <sup>Δ</sup>	99.97% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>*</sup> <sub>4,x,y</sub>	89.23%	90.56%	90.51%	90.29%
q <sup>*</sup> <sub>4,x,y</sub>	90.28% <sup>Δ</sup>	92.69% <sup>Δ</sup>	92.42% <sup>Δ</sup>	92.31% <sup>Δ</sup>
Q <sup>+</sup> <sub>1,x,y</sub>	28.26%	29.46%	28.03%	28.81%
q <sup>+</sup> <sub>1,x,y</sub>	27.32%	29.37%	28.02%	28.80%
Q <sup>+</sup> <sub>2,x,y</sub>	53.91%	54.48%	55.26%	54.68%
q <sup>+</sup> <sub>2,x,y</sub>	98.72% <sup>Δ</sup>	99.90% <sup>Δ</sup>	99.99% <sup>Δ</sup>	99.99% <sup>Δ</sup>
Q <sup>+</sup> <sub>3,x,y</sub>	53.73%	55.23%	55.34%	54.62%
q <sup>+</sup> <sub>3,x,y</sub>	98.67% <sup>Δ</sup>	99.91% <sup>Δ</sup>	99.98% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>+</sup> <sub>4,x,y</sub>	98.70%	98.75%	98.66%	98.75%
q <sup>+</sup> <sub>4,x,y</sub>	97.16%	98.55%	98.64%	98.74%

TABLE IV  
RECOGNITION PERCENTAGE FOR BSC.

Questions	N = 200	N = 300	N = 400	N = 500
Q <sup>*</sup> <sub>1,x,y</sub>	86.71%	91.65%	93.78%	94.74%
q <sup>*</sup> <sub>1,x,y</sub>	82.82%	90.62%	93.87% <sup>Δ</sup>	94.95% <sup>Δ</sup>
Q <sup>*</sup> <sub>2,x,y</sub>	94.42%	97.60%	99.03%	99.44%
q <sup>*</sup> <sub>2,x,y</sub>	99.68% <sup>Δ</sup>	99.97% <sup>Δ</sup>	99.98% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>*</sup> <sub>3,x,y</sub>	86.87%	89.43%	90.50%	90.97%
q <sup>*</sup> <sub>3,x,y</sub>	99.15% <sup>Δ</sup>	99.47% <sup>Δ</sup>	99.65% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>*</sup> <sub>4,x,y</sub>	94.39%	95.58%	95.39%	95.50%
q <sup>*</sup> <sub>4,x,y</sub>	90.78%	94.89%	95.22%	95.44%
Q <sup>+</sup> <sub>1,x,y</sub>	86.38%	91.59%	93.65%	94.71%
q <sup>+</sup> <sub>1,x,y</sub>	81.71%	90.28%	93.27%	94.57%
Q <sup>+</sup> <sub>2,x,y</sub>	94.23%	97.77%	99.19%	99.52%
q <sup>+</sup> <sub>2,x,y</sub>	99.36% <sup>Δ</sup>	99.94% <sup>Δ</sup>	100.0% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>+</sup> <sub>3,x,y</sub>	94.54%	97.39%	98.77%	99.48%
q <sup>+</sup> <sub>3,x,y</sub>	99.41% <sup>Δ</sup>	99.94% <sup>Δ</sup>	100.0% <sup>Δ</sup>	100.0% <sup>Δ</sup>
Q <sup>+</sup> <sub>4,x,y</sub>	95.40%	95.38%	95.65%	95.66%
q <sup>+</sup> <sub>4,x,y</sub>	91.81%	94.27%	95.02%	95.27%

For HRR, however the missing item has been “accidentally” correctly identified also in cases of missing *operation\*times* and *result\*times<sub>x,y</sub>* components. Such recognition did not occur in cases of missing *operation\*plus* and *result\*plus<sub>x,y</sub>* components, which is distressingly asymmetric.

HRR turned out to be the worst model during this experiment. The recognition percentage of  $Q_{1,x,y}^*$  and  $Q_{1,x,y}^+$  is dangerously low when compared to other  $Q$ 's. Both  $Q_{1,x,y}^*$  and  $Q_{1,x,y}^+$  are retrieved from the clean-up memory given only two operands and the operation type. Since we have only two operation types,  $Q_{1,x,y}^*$  and  $Q_{1,x,y}^+$  will not differ greatly from each other. This phenomenon is also observable in BSC (but not in  $GA_c$ ), where the recognition percentage of  $Q_1$ 's is only slightly lower than that of the other  $Q$ 's. Apart from that weakness, BSC performs as well as  $GA_c$  for adequate data size.

## VI. CONCLUSION

Authors developed a new model of distributed representations based on geometric algebra. Although the data representations of sentences encoded in this model may have varying lengths (as opposed to HRR and BSC), it can be justified by the fact that it is quite logical for sentences that hold more information to have larger “volume”.

Tedious calculations presented in Section 2 imply that the  $GA_c$  model is best applicable to sentences having a similar or identical complexity structure, otherwise it may be hard to make the process of asking questions and retrieving answers automatic. Because of this limitation, this construction seems to be a promising candidate for a holographic database.

## ACKNOWLEDGMENT

This work was supported by grant G.0405.08 of the Research Programme of the Research Foundation-Flanders (FWO, Belgium)

## REFERENCES

- [1] D. Aerts and M. Czachor, “Tensor-product versus geometric-product coding”, *Physical Review A*, vol. 77, id. 012316, 2008.
- [2] D. Aerts, M. Czachor, and B. De Moor, “Geometric Analogue of Holographic Reduced Representation”, *J. Math. Psychology*, vol. 53, pp. 389-398, 2009.
- [3] D. Aerts, M. Czachor, and B. De Moor, “On geometric-algebra representation of binary spatter codes”, preprint arXiv:cs/0610075 [cs.AI], 2006.
- [4] D. Aerts, M. Czachor, and Ł. Orłowski, “Teleportation of geometric structures in 3D”, *J. Phys. A* vol. 42, 135307, 2009.
- [5] W.K. Clifford, “Applications of Grassmann’s extensive algebra”, *American Journal of Mathematics Pure and Applied*, vol. 1, 350–358, 1878.
- [6] R. W. Gayler, “Multiplicative binding, representation operators, and analogy”, *Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences*, K. Holyak, D. Gentner, and B. Kokinov, eds., Sofia, Bulgaria: New Bulgarian University, p. 405, 1998.
- [7] H. Grassmann, “Der Ort der Hamilton’schen Quaternionen in der Ausdehnungslehre”, *Mathematische Annalen*, vol. 3, 375–386, 1877.
- [8] P. Kanerva, “Binary spatter codes of ordered k-tuples”. In C. von der Malsburg et al. (Eds.), *Artificial Neural Networks ICANN Proceedings, Lecture Notes in Computer Science* vol. 1112, pp. 869-873, 1996.
- [9] P. Kanerva, “Fully distributed representation”. *Proc. 1997 Real World Computing Symposium (RWC97, Tokyo)*, pp. 358-365, 1997.
- [10] N.G. Marchuk, and D.S. Shirokov, “Unitary spaces on Clifford algebras”, *Advances in Applied Clifford Algebras*, vol 18, pp. 237-254, 2008.
- [11] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2000.
- [12] A. Patyk, “Geometric Algebra Model of Distributed Representations”, in *Geometric Algebra Computing in Engineering and Computer Science*, E. Bayro-Corrochano and G. Scheuermann, eds. Berlin: Springer, 2010. Preprint arXiv:1003.5899v1 [cs.AI].
- [13] T. Plate, *Holographic Reduced Representation: Distributed Representation for Cognitive Structures*. CSLI Publications, Stanford, 2003.

# Workshop on Computational Optimization

**M**ANY real world problems arising in engineering, economics, medicine and other domains can be formulated as optimization tasks. These problems are frequently characterized by non-convex, non-differentiable, discontinuous, noisy or dynamic objective functions and constraints which ask for adequate computational methods.

The aim of this workshop is to stimulate the communication between researchers working on different fields of optimization and practitioners who need reliable and efficient computational optimization methods.

We invite original contributions related to both theoretical and practical aspects of optimization methods.

The list of topics includes, but is not limited to:

- unconstrained and constrained optimization
- combinatorial optimization
- global optimization
- multiobjective optimization
- optimization in dynamic and/or noisy environments
- large scale optimization
- parallel and distributed approaches in optimization
- random search algorithms, simulated annealing, tabu search and other derivative free optimization methods
- nature inspired optimization methods (evolutionary algorithms, ant colony optimization, particle swarm optimization, immune artificial systems etc)
- hybrid optimization algorithms involving natural computing techniques and other global and local optimization methods
- optimization methods for learning processes and data mining
- computational optimization methods in statistics, econometrics, finance, physics, medicine, biology, engineering etc

## PROGRAM COMMITTEE

**Janez Brest**, University of Maribor, Slovenia,  
janez.brest@uni-mb.si

**Jouni Lampinen**, University of Vaasa, Finland,  
antlam@uwasa.fi

**Ponnuthurai Nagarathnam Suganthan**, Nanyang Technological University, Singapore, epnsugan@ntu.edu.sg

**Michael Vrahatis**, University of Patras, Greece,  
vrahatis@math.upatras.gr

**Hideaki Iiduka**, Kyushu Institute of Technology, Japan,  
iiduka@ndrc.kyutech.ac.jp

**Patrick Siarry**, Universite Paris XII Val de Marne, France, siarry@univ-paris12.fr

**Dirk Arnold**, Dalhousie University, Canada,  
dirk@cs.dal.ca

**Kenneth Price**, US, kvprice@pacbell.net

**Antanas Zilinskas**, Research Institute of Mathematics and Informatics, Lithuania, antanasz@ktl.mii.lt

**Radomil Matousek**, University of Technology, Brno, Czech Republic, matousek@fme.vutbr.cz

**Kalin Penev**, Southampton Solent University, UK,  
kalin.penev@solent.ac.uk

**Tomas Stuetzle**, Universite Libre de Bruxelles, Belgium,  
stuetzle@ulb.ac.be

**Hiroshi Hosobe**, National Institute of Informatics, Japan,  
hosobe@nii.ac.jp

**Panos Pardalos**, University of Florida, US,  
pardalos@ufl.edu

**Joaquim Judice**, University of Coimbra, Portugal,  
Joaquim.Judice@co.it.pt

**Juan Enrique Martinez Legaz**, Universitat Autònoma de Barcelona, Spain, JuanEnrique.Martinez.Legaz@uab.cat

**Krzysztof Sikorski**, University of Utah, US,  
sikorski@cs.utah.edu

**Le Thi Hoai An**, Paul Verlaine University, Metz, France,  
lethi@sciences.univ-metz.fr

**Andries Engelbrecht**, University of Pretoria, South Africa, engel@driesie.cs.up.ac.za

**Olgierd Hryniewicz**, Polish Academy of Sciences, Poland, Olgierd.Hryniewicz@ibspan.waw.pl

**Igor Konnov**, Kazan University, Russia,  
Igor.Konnov@ksu.ru

**Stefan Stefanov**, Neofit Rilski University, Bulgaria,  
stefm@aix.swu.bg

## ORGANIZING COMMITTEE

**Stefka Fidanova**, Academy of Sciences, Bulgaria  
stefka@parallel.bas.bg

**Josef Tvrdik**, University of Ostrava, Czech Republic  
josef.tvrdik@osu.cz

**Daniela Zaharie**, West University of Timisoara, Romania  
dzaharie@info.uvt.ro



# Task Scheduling with Restricted Preemptions

Tomasz Barański

Email: tbaransk@poczta.onet.pl

**Abstract**—One of basic problems in task scheduling is finding the shortest schedule for a given set of tasks. In this paper we analyze a restricted version of the general preemptive scheduling problem, where tasks can only be divided into parts at least as long as a given parameter  $k$ . We introduce a heuristic scheduling method TSRP3. Number of processors  $m$ , number of tasks  $n$  and task lengths  $p_i$  are assumed to be known. If  $n \geq 2m$  and  $k$  is sufficiently small, TSRP3 finds shortest possible schedule with  $O(m)$  divisions in polynomial time. In addition we introduce a more robust algorithm TSRP4 based on combining TSRP3 with multi-fit.

## I. INTRODUCTION

UNRESTRICTED scheduling on parallel processors is solved in linear time using McNaughton's [7] algorithm. Unfortunately this approach often leads to processing of some tasks only for a very short time before or after preemption. Preemptions are usually costly in some way, so we introduced "granularity" threshold  $k$ , so that no part of any divided task can be shorter than  $k$ , which should be large enough to make any preemption times and costs negligible. Unless  $k$  is  $0^1$  this problem is in general NP-hard (see [4]). This paper is based on research thesis by Tomasz Barański [2].

## II. THE MODEL

In this paper we consider deterministic scheduling problem of type<sup>2</sup>

$$P \mid k - pmtn \mid C_{MAX}$$

We schedule  $n$  tasks with known lengths  $p_i$  on  $m$  identical processors<sup>3</sup> numbered from 1 to  $m$ . Tasks can be divided, but no part of any task may be shorter than some given parameter  $k$ . Any processor can work on at most one task at any given time, and tasks cannot be executed in parallel. All tasks must be completed. The aim is to find a schedule of minimum makespan  $C_{MAX}$ . There is also a secondary goal of decreasing number of divisions, and out of 2 schedules with the same  $C_{MAX}$  we prefer the one with fewer divisions.

We assume task lengths  $p_i$ , division threshold  $k$  and lengths of all divided task parts to be integer. This doesn't make our solution less general, since real numbers within

<sup>1</sup>Actually 1, since we assume lengths of all tasks and their parts are integer.

<sup>2</sup>See [6] or [8] for description of three-field notation of scheduling problems.

<sup>3</sup>By that we mean some abstract portions of work to be done on abstract machines, not actual microprocessors.

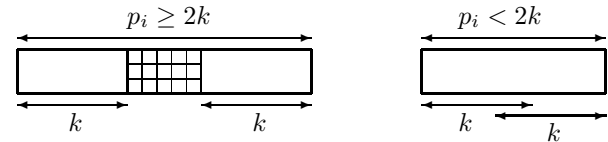


Fig. 1. Example of divisible task (left) and non-divisible task (right). First task can be divided anywhere within the grated area.

a few orders of magnitude can be scaled to integers with reasonable precession. Using integers eliminates rounding errors and simplifies calculations.

In [4] two similar models were presented. In first tasks could be preempted after being processed for at least  $k$  time. In second tasks could be preempted after being processed for a multiple of  $k$ , which allowed for easier task switching. Algorithms for solving both models were proposed, however there were no restriction on lengths of final parts of tasks, so in some cases they could be processed for a very short time after preemption.

## III. TSRP3 ALGORITHM

### A. TSRP3 introduction

TSRP3 was developed by Tomasz Barański while working on research thesis [2] and based on earlier work by Michał Bakałarczyk [1]. It is basically a heavily modified version of McNaughton's algorithm with best-fit heuristics, and it never divides any task in more than 2 parts.

1) *Sorting and grouping Tasks*: Before TSRP3 begins scheduling all tasks are sorted and divided into four groups:

$T_{SK}$	$p_j \leq k$	Shorter than $k$
$T_{ND}$	$k < p_j < 2k$	Not divisible
$T_{DIV}$	$2k \leq p_j < 3k$	Divisible
$T_{GD}$	$3k \leq p_j$	Well divisible

Divisible tasks in groups  $T_{GD}$  and  $T_{DIV}$  are generally scheduled shortest to longest, while non-divisible  $T_{ND}$  and  $T_{SK}$  are generally scheduled longest to shortest. The longer a divisible task is, the easier it is to divide in such a way, that schedule of current processor has desired length. The shorter a non-divisible task is, the easier it is to fit into schedule on current processor. Indivisible tasks from  $T_{ND}$  generally cause the most problems while scheduling, so TSRP3 tries to schedule them first.

2) *Estimating  $C_{MAX}$* : TSRP3 starts scheduling from the first processor and in each step either assigns some task or task fragment to current processor or closes it,

and moves to next processor. This decision is based on task lengths and current approximation of schedule length  $worst \geq C_{MAX}$ . After TSRP3 finishes scheduling tasks on a processor it moves to next one, and never returns to any previous processor. To explain  $worst$  we first need to introduce a few other estimates.

While scheduling tasks we always know the number of current processor  $proc$ , hence the number of processors on which we can still schedule tasks  $prc$  is (2). We also know the sum of lengths of all unscheduled tasks<sup>4</sup>  $\sigma$  and loads of all processors  $t_Z^j$  including current  $t_Z^{proc}$ .

$$C^* := \max \left\{ \max_j p_j, \left\lceil \frac{1}{n} \sum_{j=1}^n p_j \right\rceil \right\} \quad (1)$$

McNaughton's lower bound on schedule length is defined in (1). Since only one processor can work on each task at any given time, no schedule can be shorter than the longest task. All schedules must also be at least as long as the sum of all task lengths divided by the number of processors.<sup>5</sup>  $C^*$  is calculated only once, while (2) to (6) must be recomputed once for each processor.

$$prc := m + 1 - proc \quad (2)$$

Number of processors where we can still schedule tasks  $prc$  must account for current processor as well, hence in (2) we add 1 to  $m - proc$ .

$$slack := prc \cdot C^* - \sigma - t_Z^{proc} \quad (3)$$

In (3)  $slack$  is the total amount of extra space that we could in theory allocate among remaining processors without going over  $C^*$ . To put it another way:  $slack$  tells how "sloppy" we can be scheduling tasks, while still (in theory) being able to complete optimal schedule.

$$hb := \begin{cases} \left\lceil \frac{\sigma}{prc} \right\rceil & \text{if } slack \leq 0, \\ C^* & \text{if } slack > 0. \end{cases} \quad (4)$$

As was shown in explanation of McNaughton's lower bound on  $C_{MAX}$ , any schedule must be at least as long as the sum of task lengths divided by the number of processors, hence from  $prc$  and  $\sigma$  we can derive approximation  $hb$  (4). If  $slack < 0$  definition (4) is equivalent to  $hb := C^* + \left\lceil \frac{-slack}{prc} \right\rceil$ .

$$slack^{eff} := \begin{cases} \left\lfloor \frac{slack}{prc} \right\rfloor & \text{if } hb = C^* \text{ and } slack > 0, \\ 0 & \text{else.} \end{cases} \quad (5)$$

Effective slack  $slack^{eff}$  (5) is  $slack$  divided among processors that we have left. It is assumed, that if on

<sup>4</sup>In some cases we may not change processor after task division, so  $\sigma$  has to include  $t_Z^{proc+1}$

<sup>5</sup>We compute *ceiling* to get an integer value. For example if sum of task lengths is 31 and we have 3 processors,  $C_{MAX}$  with integer task parts' lengths cannot be less than 11.

current processor we have  $t_Z^{proc} \geq worst - slack^{eff}$ , it is safe to close current processor and move to next one without increasing  $C_{MAX}$ .

$$lwb := \begin{cases} C^* - slack - (hb - C^*)(prc - 1) & \text{if } slack < 0, \\ C^* - slack & \text{if } slack \geq 0. \end{cases} \quad (6)$$

Approximation  $lwb \geq 0$  in (6) is defined as "What is the shortest schedule on current processor that won't increase  $hb$  when we change processor to next one".

$$worst := \max \left\{ C^*, hb, \max_{j \in 1..prc+1} t_Z^j \right\} \quad (7)$$

Our final estimate of  $C_{MAX}$  is  $worst$  defined in (7). It is the smallest number that is at least as big as  $C^*$  and current  $hb$  and greatest processor load so far. *TSRP3 tries to make load of current processor as close to worst as possible before closing it, and moving to next processor.* This is explained in detail in next subsection.

3) *Load zones:* TSRP3 behaves differently depending on the load of current processor  $t_Z^{proc}$ . Load zones are based on our current estimate  $worst$ . They are:

$$\begin{array}{llll} (S_1) & 0 & \leq & t_Z^{proc} < worst - 2k \\ (S_2) & worst - 2k & \leq & t_Z^{proc} < worst - k \\ (S_3) & worst - k & \leq & t_Z^{proc} < worst \\ (S_4) & worst & \leq & t_Z^{proc} \end{array}$$

4) *Choosing the task that fits best on current processor and goal function  $dist(x)$ :*<sup>6</sup>

TSRP3 uses best-fit heuristics to determine which task or task part is the best candidate to schedule on active processor before moving to next one. It is generally best to close processor when  $lwb \leq t_Z^{proc} \leq worst$  and  $t_Z^{proc}$  is as close to  $worst$  as possible. As can clearly be seen in (7) increasing  $t_Z^{proc}$  above  $worst$  will increase  $worst$  and thus possibly  $C_{MAX}$ . This is still the best course of action under some circumstances, but should be severely discouraged when it comes to choosing among other options. If current processor is closed when its load was below  $lwb$ , future  $hb$  will increase. This is almost as bad as going above  $worst$ , but its effects may be spread among all remaining  $prc - 1$  processors. The "punishment" for schedule length beyond  $[lwb, worst]$  can be some sufficiently big number, like  $4k$  or  $2k + worst - lwb$ . The former is more convenient, and the latter more correct. From these considerations we get the following function that can choose the best candidate (or no candidate) to schedule on current processor:

$$dist(x) := \begin{cases} 4k + \frac{lwb-x}{prc-1} & \text{if } x < lwb, \\ worst - x & \text{if } x \in [lwb, worst], \\ 4k + x - worst & \text{if } worst < x. \end{cases} \quad (8)$$

<sup>6</sup>This is actually  $dist(x, worst, lwb, prc)$ , but we abbreviated it to  $dist(x)$

TABLE I  
EXPLANATION OF USED SYMBOLS

Symbol	Meaning	See	Upd.
$n$	Total number of tasks	—	No
$m$	Total number of processors	—	No
$p_j$	Durations of tasks $j \in 1 \dots n$	—	No
$C_{MAX}$	Schedule length	—	No
$C^*$	Lower bound on $C_{MAX}$	(1)	No
$proc$	Current processor $proc \in 1 \dots m$	—	Proc.
$prc$	Number of processors left	(2)	Proc.
$t_Z^j$	Load of processor $j$ . Usually $t_Z^{proc}$	—	Yes
$\sigma$	Sum of unscheduled task durations	—	Yes
$slack$	Amount of “Slack space” under $C^*$	(3)	Proc.
$hb$	Estimate $C_{MAX}$ from $\sigma$ and $prc$	(4)	Proc.
$lwb$	Lowest $t_Z^{proc}$ that won't increase $hb$	(6)	Proc.
$worst$	Current estimate of $C_{MAX}$	(7)	Yes.
$slack^{eff}$	Amortized slack space on $proc$	(5)	Proc.

When choosing task or task part  $z_i$  to assign to current processor, TSRP3 finds  $z_i$ , for which  $dist(t_Z^{proc} + p_i)$  or  $dist(t_Z^{proc} + length)$  is the lowest. If  $dist(t_Z^{proc})$  is lower than for any task part we can schedule, TSRP3 moves to the next processor without scheduling anything on current one. It is more convenient to make  $dist(x)$  a real function. Integer values can be used, but for  $x < lwb$  you have to use  $\lceil dist(x) \rceil$  and choose the longest task among those with the same (lowest) value of  $dist()$ .

Table I presents a list of symbols used by TSRP3, along with references to equations defining these symbols. Column Upd. says how often a variable is updated: No means a fixed parameter, Proc. means update once per processor, and Yes means update before scheduling another task.

### B. TSRP3 Step by step

TSRP3 begins from the first processor. It assigns tasks to the current (open) processor, trying to make its load as close to  $worst$  as possible. When current processor is sufficiently loaded, TSRP3 closes it, and moves to (opens) next processor, which in turn becomes the current processor. For each assignment it generally chooses some task or task part  $z_i$ , for which  $dist(t_Z^{proc} + z_i)$  is lowest. This task or task part is called candidate, and may be compared to more tasks, scheduled, or discarded if  $dist(t_Z^{proc} + z_i) > dist(t_Z^{proc})$  where  $t_Z^{proc}$  is the load of current processor. If a task is divided, it's remaining part is assigned to the next processor. TSRP3 can be presented in the following steps:

- 1) Sort and divide tasks in groups  $T_{SK}$ ,  $T_{ND}$ ,  $T_{DIV}$ ,  $T_{GD}$ . Compute  $C^*$ .
- 2) If there are at least  $m$  well divisible tasks  $T_{GD}$ , schedule all tasks of length  $C^*$  on separate processors.
- 3) If there are no unscheduled tasks, finish.
- 4) If  $m \geq n$ , schedule all remaining tasks on separate processors and finish.
- 5) If  $proc = m$ , schedule all remaining tasks on  $proc$ , and finish.
- 6) Recompute  $slack$ ,  $slack^{eff}$ ,  $hb$ ,  $lwb$ ,  $worst$ , updating them once per processor.

- 7) Update  $t_Z^{proc}$  and  $worst$ . If  $(worst - slack^{eff}) \leq t_Z^{proc}$  and  $0 < t_Z^{proc}$  change processor to next one and go to step 5.
- 8) Take action depending on zone in which  $t_Z^{proc}$  is. This is explained in detail below, and may result in scheduling some task or task part, changing processor to the next one, or choosing a candidate task to schedule.
- 9) Choose candidate task to schedule. Take into account:
  - Candidate from step 8 if chosen
  - Longest and shortest tasks in each group.
  - Task division
  - *combo* (divides 2 or 3 tasks at once, explained later).

If  $dist(t_Z^{proc} + z_i) \leq dist(t_Z^{proc})$  for chosen candidate  $z_i$ , schedule it on current processor, otherwise move to next processor. Go to step 3

In step 2 we slightly reduce number of divisions in special cases, with lots of well divisible tasks. Step 4 takes care of some trivial cases. It is better to compare the number of unscheduled tasks to number of open processors except current one, rather than total numbers of processors and tasks. Steps 3 to 9 constitute the main loop of TSRP3. Step 7 is there to avoid divisions where not necessary. If  $slack^{eff}$  is positive, we have a good chance to complete optimal schedule. Step 8 is the most complex, and is explained below.

Following subsections are divided by zones  $S_1$  to  $S_4$ , and in one run of step 8 only one subsection gets executed depending on  $t_Z^{proc}$ . When a task is scheduled, it is considered to be removed from appropriate task set, therefore “ $T_{SK} \neq \emptyset$ ” means “there is at least one unscheduled task shorter than  $k$ ”. In some cases we use term “divisible task”  $z_i$ . It means that  $z_i \in T_{DIV} \cup T_{GD}$ , and we generally use it, when we want to select as candidate the shortest task, that we can conveniently divide to make load of current processor equal to  $worst$ . Each subsection consists of sequentially checking some conditions and executing some instructions, such as scheduling a task. If conditions for an item are met, then its instructions are executed, including possibly jump to step 3 described above. If conditions for an item are not met, ignore it's instructions, and jump to next item on the same level. By making some task  $z_i$  a candidate to assign in step 9 we mean comparing it first to current candidate (if any) with  $dist()$  and only making  $z_i$  candidate, if it's  $dist(p_i + t_Z^{proc})$  is lower than for current candidate.

1) Step 8, zone  $S_1$ :  $t_Z^{proc} \in [0, worst - 2k)$ :

- If  $T_{ND} \neq \emptyset$ , choose as candidate the longest non-divisible task  $z_i$  such that either  $p_i \leq (worst - k - t_Z^{proc})$  or  $(worst - slack^{eff} - t_Z^{proc}) \leq p_i$ . If the latter condition is met, or  $z_i$  is the longest task in  $T_{ND}$ , schedule that task on current processor and go to step 3.

- If there are at least  $prc$  well divisible unscheduled tasks left in  $T_{GD}$ 
  - Find the shortest divisible task  $z_i \in T_{DIV} \cup T_{GD}$  such that either  $(p_i + t_Z^{proc}) \in [worst - slack^{eff}, worst]$  or  $(p_i + t_Z^{proc}) \leq (worst - k)$ . If such a task exists schedule it and go to step 3.
  - Find the shortest divisible task  $z_i$  such that  $(p_i + t_Z^{proc}) \geq (worst + k - slack^{eff})$  If it exists divide it, scheduling part of length  $\min\{worst - t_Z^{proc}, p_i - k\}$  on processor  $proc$  and the rest on  $proc + 1$ , change current processor to next one and go to step 3.
- If there are more unscheduled tasks in  $T_{SK}$  than in  $T_{DIV} \cup T_{GD}$ , schedule the longest task from  $T_{SK}$  on current processor and go to step 3.
- If  $T_{DIV} \cup T_{GD} \neq \emptyset$ 
  - If the shortest divisible task  $z_i$  is longer than  $(worst - k - t_Z^{proc})$  and shorter than  $(worst + k - slack^{eff} - t_Z^{proc})$  and  $(p_i + t_Z^{proc}) \notin [worst - slack^{eff}, worst]$  and there are unscheduled tasks in  $T_{SK}$ , keep scheduling them on current processor until they run out, or one of above conditions is no longer satisfied or scheduling even shortest task would cause  $t_Z^{proc}$  to increase above  $(worst - k)$ .<sup>7</sup>
  - Choose the shortest divisible task  $z_i$  as candidate.
  - If task  $z_i$  is longer than  $(worst - k - t_Z^{proc})$  and shorter than  $(worst + k - slack^{eff} - t_Z^{proc})$  and  $(p_i + t_Z^{proc}) \notin [worst - slack^{eff}, worst]$ , choose as candidate  $z_i$  the shortest divisible task of length at least  $(worst + k - slack^{eff} - t_Z^{proc})$ . If it does not exist, then make the longest divisible task candidate.<sup>8</sup>
  - If  $(p_i + t_Z^{proc}) \leq (worst - k)$  or  $(p_i + t_Z^{proc}) \in [worst - slack^{eff}, worst]$ , schedule  $z_i$  on current processor and go to step 3.<sup>9</sup>
  - If  $(p_i + t_Z^{proc}) \geq (worst + k - slack^{eff})$  divide  $z_i$ , scheduling part of length  $\min\{worst - t_Z^{proc}, p_i - k\}$  on processor  $proc$  and the rest on  $proc + 1$ , change current processor to next one and go to step 3.
  - If there are at least 2 unscheduled divisible tasks and  $(worst - t_Z^{proc}) \geq 2k$  and  $t_Z^{proc} \geq k$ , try dividing 2 or 3 tasks using *combo*<sup>10</sup> heuristics. If successful, change processor to the next one and go to step 3.
  - Find the longest divisible task such that  $(p_i + t_Z^{proc}) \leq worst$ . If it exists, make it candidate to add  $z_i$ .
- Go to step 9.

<sup>7</sup>If there is a problem with division,  $t_Z^{proc}$  can be increased by scheduling some short tasks.

<sup>8</sup>By dividing task as short as possible, we preserve longer divisible tasks to be scheduled on another processor.

<sup>9</sup>We avoid zone  $S_3$  with this check.

<sup>10</sup>Explained in III-C.

Operations in this zone are the most complex, because we are actively avoiding zones  $S_3$  and  $S_4$  except for range  $[worst - slack^{eff}, worst]$ . We also divide long tasks, if scheduling them would make  $t_Z^{proc} \geq (worst + k - slack^{eff})$ . Before going to step 9, where we do some operations common to all zones, we may or may not choose a task as candidate to schedule. If selected, it is compared in step 9 to other possibilities, including scheduling nothing and moving to next processor.

2) Step 8, zone  $S_2$ :  $t_Z^{proc} \in [worst - 2k, worst - k]$ :

- If  $T_{ND} \neq \emptyset$ , find the longest task  $z_i \in T_{ND}$  such that  $(p_i + t_Z^{proc}) \leq worst$ . If it exists make it candidate to schedule. If it satisfies  $(p_i + t_Z^{proc}) \geq (worst - slack^{eff})$  schedule it, and go to step 3.
- If there is some divisible unscheduled task left.
  - If the shortest divisible task is shorter than  $(worst + k - slack^{eff} - t_Z^{proc})$ , and there are unscheduled tasks in  $T_{SK}$ , keep scheduling them on current processor until they run out, or the above condition is no longer satisfied or scheduling even the shortest task would cause  $t_Z^{proc}$  to increase above  $worst - k$ .
  - Find the shortest divisible task  $z_i$  such that  $(p_i + t_Z^{proc}) \geq (worst + k - slack^{eff})$ . If it exists, divide  $z_i$ , scheduling part of length  $\min\{worst - t_Z^{proc}, p_i - k\}$  on processor  $proc$  and the rest on  $proc + 1$ , change current processor to next one and go to step 3.
- Go to step 9.

In this zone we first try to find a non-divisible task of length  $(p_i + t_Z^{proc}) \in [worst - slack^{eff}, worst]$ . If that fails, we try to divide some task instead, possibly increasing  $t_Z^{proc}$  with tasks from  $T_{SK}$ . Other actions, common to all zones, are considered in step 9.

3) Step 8, zone  $S_3$ :  $t_Z^{proc} \in [worst - k, worst)$ :

- If there is at least one unscheduled task in  $T_{SK}$ , find the longest task  $z_i$  in  $T_{SK}$  such that  $(p_i + t_Z^{proc}) \leq worst$ . If it exists, make  $z_i$  the candidate to schedule. If  $(p_i + t_Z^{proc}) \geq (worst - slack^{eff})$  schedule  $z_i$ , and go to step 3.
- Go to step 9.

Here the only actions that make sense are: fitting a task from  $T_{SK}$  or scheduling task part of length  $k$  or changing current processor to next one without scheduling anything. Only the first option is not covered in step 9.

4) Step 8, zone  $S_4$ :  $t_Z^{proc} \geq worst$ :

- Move to next processor, and go to step 9.

We never actually get to this zone, because condition  $t_Z^{proc} \geq (worst - slack^{eff})$  is checked in step 7, but it is safer to include this check.

### C. Dividing 2 or 3 tasks with *combo*

Description of TSRP3 algorithm has two references to *combo* heuristics, that tries to divide 2 or 3 tasks before changing current processor to next one. This is useful,



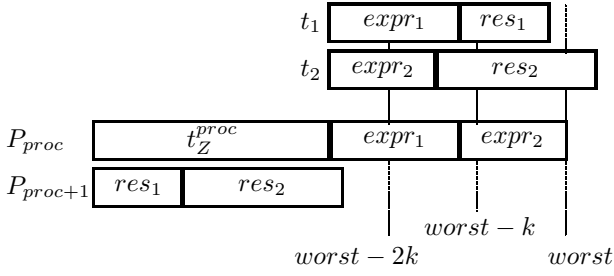


Fig. 2. Dividing two tasks with combo.

when there are some short divisible tasks, and scheduling any one task, divided or not, will guarantee  $t_Z^{proc}$  to be in either zone  $S_3$  or  $S_4$  and outside  $[worst - slack^{eff}, worst]$ . While this is counter to our stated goal of reducing the number of divisions, our *primary* aim is still to minimize  $C_{MAX}$ .

Consider the following example:  $slack^{eff} = 0$ ,  $t_Z^{proc} = (worst - 2k - \epsilon)$ , all indivisible tasks have lengths  $p_i > (k + \epsilon)$ , and all divisible tasks have lengths  $p_i = (2k + \lambda_i)$ , where  $\epsilon, \lambda \geq 0$  and  $\epsilon \neq \lambda_i$  and  $\forall i \epsilon - k < \lambda_i < \epsilon + k$ . No single task can be assigned or divided so as to avoid zones  $S_3$  and  $S_4$  or make load of current processor equal to  $worst$ . This can be done by increasing  $t_Z^{proc}$  by about  $k$ , or more precisely between  $\epsilon + k - \lambda_i$  and  $\epsilon + k$  for some  $i$ . It can be achieved by double division of tasks, as shown in Fig. 2.

If there are 2 divisible tasks  $z_1$  and  $z_2$  such that  $p_1 + p_2 \geq 4k + \epsilon - slack^{eff}$ <sup>11</sup> and they can be divided in a way that satisfies (9) to (12), and scheduled like in Fig. 2, then it won't result in any conflicts (overlapping task execution on different processors).

$$t_Z^{proc} \geq k \quad (9)$$

$$expr_1, res_1, expr_2, res_2 \geq k \quad (10)$$

$$t_Z^{proc} \geq res_1 \quad (11)$$

$$res_1 + res_2 \leq t_Z^{proc} + expr_1 \quad (12)$$

$$res_1 + res_2 + expr_2 \leq worst \quad (13)$$

Inequality (9) is a sanity check derived from (10) and (11). Inequality (10) ensures that there are no task parts shorter than  $k$ . Inequality (11) is more precise than (9) and makes conflict between  $expr_1$  and  $res_1$  impossible. Inequality (12) does the same for  $expr_2$  and  $res_2$ . Finally inequality (13) forbids *combo* on too long tasks, and makes sure that  $t_Z^{proc+1} \leq (worst - k)$ .

<sup>11</sup>This is a bit too restrictive. In some cases the next best course of action chosen with  $dist()$  is so bad, that it is better to use *combo* anyway, even if it makes  $t_Z^{proc} \notin [worst - slack^{eff}, worst]$ . If this next best thing would be assigning whole task  $z_i$  longer than  $(worst - t_Z^{proc})$ , scheduling 2 task parts shorter than  $p_i$  may be a better option. On the other hand when there are few processors left, and we are to either move to next processor while  $t_Z^{proc} < lvb$ , or assign some task  $z_i$  and land in zone  $S_3$  or  $S_4$ , scheduling 2 task parts whose sum is longer than  $p_i$  and shorter than the resulting  $hb$ , may also be better.

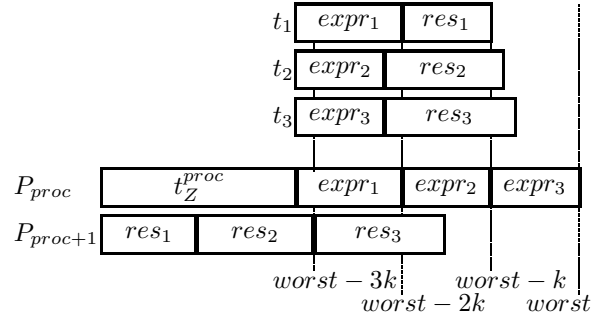


Fig. 3. Dividing three tasks with combo.

As it turns out, sometimes dividing 2 tasks with *combo* is not enough to make  $t_Z^{proc} = worst$ . Such situation arises when there are only divisible tasks left, there is at least 3 of them,  $t_Z^{proc} = (worst - 3k + \epsilon)$  and  $p_i = (2k + \lambda_i)$  where  $\forall i 0 \leq \epsilon < \lambda_i < \frac{k + \epsilon}{2}$ . In this case scheduling whole task or dividing 2 tasks causes  $t_Z^{proc}$  to be in zone  $S_3$ , which is undesirable. To solve this problem we find 3 tasks that satisfy inequalities (14) to (19) and assign their parts like in Fig. 3.

$$t_Z^{proc} \geq k \quad (14)$$

$$expr_1, res_1, expr_2, res_2, expr_3, res_3 \geq k \quad (15)$$

$$t_Z^{proc} \geq res_1 \quad (16)$$

$$t_Z^{proc} + expr_1 \geq res_1 + res_2 \quad (17)$$

$$t_Z^{proc} + expr_1 + expr_2 \geq res_1 + res_2 + res_3 \quad (18)$$

$$res_1 + res_2 + res_3 + expr_3 \leq worst \quad (19)$$

It is generally best to choose for *combo* as short a tasks as possible, divide shortest first and longest last, and always make *expr* as long as possible, while reserving  $k$  space for all future *expr*.

Scheduling  $res_i$  on different processors would relax some restrictions, but it would complicate computing  $\sigma$ ,  $hb$ ,  $lvb$ , and require keeping track of loads on too many processors. We do not recommend it, and in practice task divisions between current and next processors suffice.

It is never necessary to divide more than three tasks at once. If scheduling any divisible task  $z_i$  would cause  $t_Z^{proc}$  to end up in  $S_3$  or  $S_4$ , than increasing  $t_Z^{proc}$  by at most  $2k$  by dividing at most two other tasks, makes  $(t_Z^{proc} + p_i) \geq (worst + k)$ . Therefore dividing at most 3 tasks will always suffice, provided it is possible.

#### D. Sufficient conditions of optimality for TSRP3

With  $k = 0$  (or  $k = 1$  for all task parts' lengths integer) optimal schedule can be completed in  $O(n)$  time with McNaughton's algorithm. Scheduling indivisible tasks is in general an NP-hard problem (although there are reasonably good rough algorithms). The question arises of where is the threshold value  $k_l$  between these two cases or how many well divisible tasks are needed for guaranteed

completion of optimal schedule in polynomial time. This subsection addresses these questions by showing sufficient conditions for TSRP3 to complete an optimal schedule. As shown in section V these conditions are very restrictive, and in fact TSRP3 often completes optimal schedule with fewer than  $m$  divisible tasks.

Case a)

$$\text{For } 0 \leq l \leq m \text{ tasks } p_i = C^* \quad (20)$$

$$\text{For other tasks: } p_i \leq C^* - 2k \quad (21)$$

$$\text{There are } 2(m-l-1) \text{ tasks in } T_{GD} \quad (22)$$

$$\text{For pairs of them } p_a + p_b \leq (C^* + k) \quad (23)$$

All tasks (20) of length  $C^*$  are assigned to separate processors. This way we will decrease number of divisions while still getting optimal schedule. Condition (23) stands for this: We have at least  $2(m-l-1)$  well divisible tasks, and we can form  $(m-l-1)$  pairs, such that sum of their lengths is  $\leq (C^* + k)$ . This may be done by choosing some subset  $M$  of  $T_{GD}$ , and pairing longest and shortest tasks in  $M$ . They may be used for *combo*. Let us consider what actions TSRP3 will take depending on  $t_Z^{proc}$ . If  $t_Z^{proc} = 0$ , then after scheduling any task we have  $t_Z^{proc} \leq C^* - k$  because (21), so we avoid zones  $S_3$  and  $S_4$ . If  $t_Z^{proc} \leq C^* - 2k$  and by scheduling some task  $z_i$  we would end up in zone  $S_3$ , we can do a double division instead. If  $C^* - 2k < t_Z^{proc} \leq C^* - k$ , we can divide one task from (22). Therefore we never get to zones  $S_3$  and  $S_4$ , and we always have enough long tasks to make  $t_Z^{proc} = C^*$  by dividing one or two of them.

Case b)

$$\text{For } 0 \leq l \leq m \text{ tasks } p_i = C^* \quad (24)$$

$$\text{For other tasks: } p_i \leq C^* - 2k \quad (25)$$

$$\text{There are } (m-l-1) \text{ well divisible tasks} \quad (26)$$

$$\text{And } 2(m-l-1) \text{ more divisible tasks} \quad (27)$$

$$\text{Sum of 1 (26) and 2 (27) is } \leq C^* + 2k \quad (28)$$

Now we have at least  $(m-l-1)$  well divisible tasks and additionally at least  $2(m-l-1)$  divisible tasks. This is different from above case in that we can have fewer well divisible tasks, but need more divisible tasks overall. We also need to be able to combine these tasks in triplets satisfying (28). Reasoning proceeds as above, but if necessary we divide 3 tasks.

Case c)

$$\text{For } 0 \leq l \leq m \text{ tasks we have } p_i = C^* \quad (29)$$

$$\text{For other tasks: } p_i \leq C^* - 2k \quad (30)$$

$$\text{There are } (m-l-1) \text{ well divisible tasks} \quad (31)$$

$$\text{Sum of tasks in } T_{SK} \text{ is } \geq 2(m-l-1) \quad (32)$$

If by scheduling a well divisible task  $z_i$  we get  $t_Z^{proc} > (C^* - k)$ , then we need to extend  $t_Z^{proc}$  by at most  $2k$  to divide  $z_i$  and make  $t_Z^{proc} = C^*$ . Tasks shorter than  $k$  are useful for this. Since zone of possible division as shown in Fig. 1 for well divisible task is at least  $k$  long, and  $T_{SK}$  are at most  $k$  long, there never will be any trouble fitting some  $T_{SK}$  on current processor to divide  $z_i$  as long as there are enough of them.

Case d)

$$\text{For } 0 \leq l \leq m \text{ tasks } p_i = C^* \quad (33)$$

$$\text{For other tasks: } p_i \leq C^* - 2k \quad (34)$$

$$\text{We have } (m-l-1) \text{ tasks of length } p_i \geq 4k \quad (35)$$

$$\text{And } (m-l-1) \text{ tasks of length } p_i \geq k \quad (36)$$

Tasks (36) are there to avoid some malicious cases. Long tasks (35) have division zone of length at least  $2k$ . This makes it easy to make loads of all processors at most  $C^*$ . If by scheduling an indivisible task, we would get to zone  $S_3$ , we can instead divide one of tasks (35). If by scheduling a divisible task we would get to zone  $S_3$ , we can do a double division instead.

Case e)

$$|a| + |b| + |c| + |d| \geq (m-l-1) \quad (37)$$

We have a combination of cases a) b) c) d) and the sum of number of processors, for which at least one of them can be used to make  $t_Z^{proc} = C^*$  is at least  $(m-l-1)$ . We use different methods to complete optimal schedule.

1) *Conclusions:* These are sufficient, but not necessary conditions for completion of optimal schedule by TSRP3. As shown in section V, in practice TSRP3 will usually find optimal schedule as long as there are at least  $m/2$  divisible tasks and  $2m$  other tasks. Having at most  $l$ , where  $0 \leq l \leq m$  tasks of length  $C^*$  and other tasks no longer than  $C^* - 2k$  is forced just to exclude malicious data sets.

A simple rule for finding  $k_l$  for which TSRP3 will complete optimal schedule is as follows (38):

$L(i) :=$  Length of  $i$ -th longest task

$$\begin{aligned} k_a &= \left\lfloor \frac{L(2m)}{3} \right\rfloor \\ k_b &= \min \left\{ \left\lfloor \frac{L(3m)}{2} \right\rfloor, \left\lfloor \frac{L(m)}{3} \right\rfloor \right\} \\ k_l &= \max \{k_a, k_b\} \end{aligned} \quad (38)$$

As shown in [2], computational complexity of TSRP3 is  $O(n \cdot \log(n))$ , which is comparable to sorting. Memory requirements depend on data structures used for schedule, but are around 100b or less per task.

#### IV. TSRP4 ALGORITHM

TSRP3 works well when there are many long divisible tasks, but noticeably worse when there are few or none. TSRP4 is a method for making TSRP3 more robust with indivisible tasks. As can be concluded from (7) on page 2 *worst* either increases or stays the same when we move to the next processor. TSRP3 tries to make the load of current processor as close to current *worst* as possible, which may lead to overloading last processors while underloading first processors. To address this we introduce parameter  $C_{OGR} \geq C^*$ , and modify (7) to make sure that  $worst \geq C_{OGR}$ . Our goal then becomes to find such  $C_{OGR}^*$  that modified TSRP3 can complete a schedule of length  $C_{OGR}^*$ , but not  $C_{OGR}^* - 1$ .

This is accomplished by first setting  $C_{OGR}$  to  $C^*$ , and running TSRP3. If the length of resulting schedule  $len$  is  $C^*$  then finish, else  $C^* < C_{OGR}^* \leq len$  and  $C_{OGR}^*$  can be found with bisection of  $(C^*, len]$  by completing  $O(\log(len - C^*))$  schedules with modified TSRP3. Schedule completed with TSRP4 is never longer than completed with TSRP3 for the same input.

#### V. EXPERIMENTAL COMPARISON OF TSRP3 WITH OTHER SCHEDULING ALGORITHMS

To show behavior of TSRP3 and TSRP4 depending on  $k$ , they were tested for 10 randomly generated data sets. Each contained 500 tasks with Gauss distribution with  $\mu = 100$  and  $\sigma = 30$  to schedule on 100 processors. If there are fewer tasks than processors then scheduling is trivial, and if there are many (10+) tasks of wildly varying lengths, there is usually a good schedule without task divisions. Scheduling is most problematic, when there are few tasks per processor. Even statistically small sample of 10 data sets is enough to form opinion on behavior of TSRP3 and TSRP4 for changing  $k$ . The results were averaged and are shown in Fig. 4. They were taken from [2], where a slightly different version of TSRP3 was implemented.

Some labels in Fig. 4 require explanation.  $C^*$  is McNaughton's lower bound on schedule length (1). LPT and Multi-fit are algorithms scheduling tasks without dividing them. LPT assigns the longest task to the first processor that becomes idle. Multi-fit was our inspiration for TSRP4. It has "superior" and "subordinate" parts. Superior part manipulates parameter  $C_{OGR}$  to find  $C_{OGR}^*$  as described in IV. Subordinate part starts from the first processor and in each step either schedules on current processor the longest task that fits under  $C_{OGR}$  or moves to the next processor.

In Fig. 4 there are two main areas. Below  $k = 60$  there are at least  $m$  divisible tasks, and there are some well divisible tasks. Above  $k = 80$  the number of divisible tasks drops rapidly.

In III-D it was shown, that with sufficient number of divisible tasks TSRP3 can complete optimal schedule of length  $C^*$ . There may however exist some optimal schedules with fewer task divisions. Sufficient conditions

for completing optimal schedule given in III-D turn out to be very restrictive. During tests and barring maliciously chosen data sets TSRP3 was generally not failing at completing optimal schedules until number of divisible tasks dropped below  $m/2$ . When there are few or none divisible tasks, TSRP3 usually completes schedules longer than LPT, which does not divide tasks at all. When there are no divisible tasks, and  $k$  changes, length of TSRP3 schedule fluctuates, because border between  $T_{SK}$  and  $T_{ND}$  moves, so tasks are scheduled in different order. This leads to a somewhat paradoxical observation that sometimes a shorter schedule may be completed by increasing  $k$ . These fluctuations generally stop well before all tasks are classified as  $T_{SK}$ .

Algorithm TSRP4 was written to mitigate shortcomings of TSRP3 when there are few or none divisible tasks. As you can see in Fig. 4 schedule completed with TSRP4 is never longer than with TSRP3. For relatively short  $k$  TSRP4 schedule length is optimal, and as  $k$  grows,  $C_{MAX}$  approaches length of schedule completed with Multi-fit. This shows similarity of these two algorithms. Fluctuations with changing  $k$  and no divisible tasks are also smaller. This is an improvement over TSRP3, but it may require completing several schedules to find  $C_{OGR}^*$ .

In paper [2] an even better scheduling algorithm was devised by completing 2 schedules: one with TSRP4 and another with a good scheduling algorithm that doesn't divide tasks: "LPT + PSS", and choosing shorter one, or one with fewer divisions, if both schedule lengths were the same. In short LPT + PSS uses local search to balance loads of processors in schedule completed by LPT.

Algorithm LPT + PSS doesn't divide tasks, and was developed by Tomasz Barański and presented in [2]. It has proven to be superior to both multi-fit and LPT in terms of  $C_{MAX}$ , but takes longer to compute. Several speed improvements were proposed to mitigate this. LPT + Presistent Simple Swap starts by computing a schedule with LPT and sorting processors by load. In each step it chooses the most loaded processor  $P_{MAX}$ , and the least loaded  $P_{MIN}$  and tries to balance their loads by either moving one task from  $P_{MAX}$  to  $P_{MIN}$  or switching a pair of tasks between them. It tries to balance loads of these two processors as much as possible. If successful it updates  $P_{MAX}$  and  $P_{MIN}$  and continues balancing loads, otherwise it changes  $P_{MIN}$  to another processor with schedule shorter than that of  $P_{MAX}$ . PSS stops when  $C_{MAX}$  reaches  $C^*$  or no further balancing can be done. Any feasible schedule without tasks division can be used as starting point for PSS, such as multi-fit or even scheduling all tasks on a single processor, but in [2] LPT + PSS produced the best results. For data sets used in Fig. 4 in 9 out of 10 cases LPT + PSS produced schedule with  $C_{MAX} = C^*$ . It may seem superior to TSRP4, but in some cases optimal schedule can only be obtained by dividing tasks.

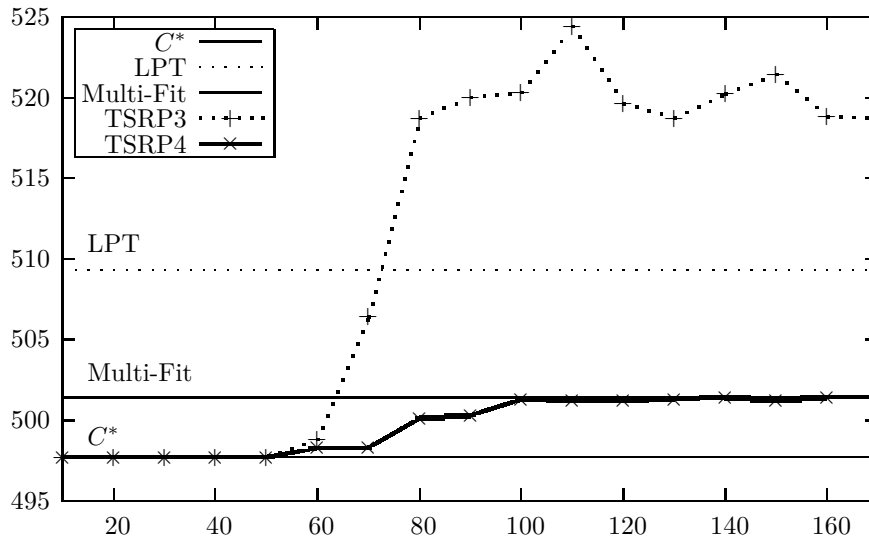


Fig. 4. Relation between  $k$  (horizontal axis) and  $C_{MAX}$  (vertical axis) for TSRP3 and TSRP4. Less is better. Results are averaged for 10 data sets with similar  $C^*$ . While  $k$  was gradually increased, Task durations stayed the same.  $C^*$  is lower bound on schedule length. LPT and Multi-fit don't divide tasks, and are shown for comparison. As  $k$  increases, number of divisible tasks drops, and TSRP3 produces worse results. TSRP4 is much more robust.

## VI. CONCLUDING REMARKS

In this article we proposed TSRP3, a heuristic algorithm for completing schedules with partial task division. As we have shown in (III-D) it is guaranteed to find optimal schedule if there are at least  $2m$  tasks  $3k$  long or longer.

While schedules completed with TSRP4 may be satisfactory in general, and optimal with enough divisible tasks, in some cases dividing tasks in at most two parts is not enough. We will illustrate this with a simple example: We have  $m = 3$  processors and  $n = 7$  tasks, with  $k = 4$ . There is one task of length  $p_1 = 12$  and six tasks  $p_i = 6$ . To get optimal schedule we need to divide the longest task in 3 parts of length 4 and schedule one on each machine with two indivisible tasks. This way we get  $C_{MAX} = C^* = 16$ . If we divide the longest task in at most 2 parts, then no part of it will be on one processor  $P_3$ . Therefore we either schedule 2 or 3 indivisible tasks on that processor. If we schedule 3 tasks, their combined length is  $6 \cdot 3 = 18$ . If we schedule only 2 tasks, then we need to divide tasks of length  $12 + 6 \cdot 4 = 36$  among 2 processors, so schedule won't be shorter than 18. Therefore in this example we need to cut some task in more than 2 fragments to get optimal schedule. A method of doing this is hinted in [2].

TSRP3 or TSRP4 may be used to solve one-dimensional bin-packing problem, like cut-weld problem, by changing  $m$ , running TSRP4 and comparing resulting  $C_{MAX}$  to target bin size. This is similar to method used in TSRP4

to find  $C_{OGR}^*$ . Please note, however, that in bin-packing problem there are no conflicts. We recommend using a dedicated algorithm for bin-packing rather than TSRP4. We suggest introducing some cost of dividing tasks for further study of TSRP3. Division of tasks in more than two parts is also worth investigating.

## REFERENCES

- [1] Michał Bakalarczyk, "Szeregowanie zadań z ograniczoną podzielnością na procesorach równoległych", *Praca inżynierska*, PW EiT, Warszawa 2006.
- [2] Tomasz Barański, "Szeregowanie zadań z częściową podzielnością na procesorach równoległych", *Praca magisterska*, PW EiT, Warszawa 2010.
- [3] Krzysztof Trakiewicz "Modele i algorytmy optymalizacji rozkroju i spawania kształtów", *Praca magisterska*, PW EiT, Warszawa 2004.
- [4] K. Ecker, R. Hirschberg "Task scheduling with restricted preemptions", p. 464-475, LNCS vol. 694, Springer-Verlag 1993.
- [5] Nir Menakerman, Raphael Rom "Bin packing with item fragmentation", WADS 2001, LNCS 2125, p. 213-324, Springer-Verlag, Heidelberg 2001.
- [6] A. Janiak "Wybrane problemy i algorytmy szeregowania zadań i rozdziału zasobów", Akademicka Oficyna wydawnicza PLJ, Warszawa 1999.
- [7] Robert Mc Naughton, "Scheduling with Deadlines and Loss Functions", *Management Science*, Vol. 6, No. 1, 1959.
- [8] B. Chen, C.N. Potts, G.J. Woeginger. "A review of machine scheduling: Complexity, algorithms and approximability.", *Handbook of Combinatorial Optimization*, Vol. 3, 1998, 21-169.
- [9] Manfred Kunde "A multifit algorithm for uniform multiprocessor scheduling", *Lecture Notes in Computer Science*, 1982, Volume 145/1982, 175-185.

# A Branch-and-Cut-and-Price Algorithm for a Fingerprint-Template Compression Application

Andreas M. Chwatal\*, Corinna Thöni\*, Karin Oberlechner\*, Günther R. Raidl\*

\*Vienna University of Technology, Institute of Computer Graphics and Algorithms  
 1040 Vienna, Austria

Email: chwatal@ads.tuwien.ac.at, corinnathoeni@gmail.com,  
 karin.oberlechner@gmail.com, raidl@ads.tuwien.ac.at

**Abstract**—In this work we present a branch-and-cut-and-price algorithm for the compression of fingerprint minutiae templates, in order to embed them into passport images by watermarking techniques as an additional security feature. For this purpose the minutiae data, which is a set of characteristic points of the fingerprint, is encoded by a spanning tree whose edges are encoded efficiently by a reference to an element in a dictionary (*template arc*) and a small correction vector. Our proposed branch-and-cut-and-price algorithm creates meaningful template arcs from a huge set of possible ones on demand in the pricing-procedure. Cutting-planes are separated in order to obtain connected subgraphs from which spanning trees can then be easily deduced. Our computational experiments confirm the superior performance of the algorithm in comparison to previous approaches for the spanning tree based encoding scheme.

## I. INTRODUCTION

THIS work is based on [1], where the authors introduced a combinatorial model to perform data compression specifically for a (small) set of unordered, multidimensional data-points. The need for such a kind of compression arises for instance when fingerprint minutiae data should be encoded in a way permitting this data to be embedded into images by watermarking techniques as additional security feature. For a detailed description of the application background we refer the reader to [1].

The compression model is based on a special kind of tree structure, connecting a subset of certain size of the given nodes. The nodes themselves correspond to the set of input data-points, i.e. the fingerprint minutiae. Compression is achieved by computing a suitable small set of *template arcs*, which enable a more efficient encoding of the intended tree structure. The compression model can thus be seen as a dictionary approach. Decompression is achieved by reconstructing the according subset of data-points from this particular tree structure. In section II we review the problem model.

In this work we focus on a more efficient solution of the underlying combinatorial optimization problem (COP). The model is actually a combination of two well-known COPs, the *minimum label spanning tree problem* [2] and the *k-cardinality minimum spanning tree problem* [3]. The resulting problem is called *k-minimum label spanning arborescence* (*k-MLSA*) problem.

The contribution of this article is a *branch-and-cut-and-price* (BCP) framework, detailed in section III, to solve the *k-MLSA* problem, and has for the first time been investigated

in the theses [4]–[6]. Within this framework we use cutting-planes to obtain validity of the initially incomplete model and dynamically create and add new label variables in the pricing phase. In section III-A we present efficient methods to solve the according pricing subproblem, and the experimental results in section IV show that the presented exact algorithm is able to solve the problem to provable optimality within a reasonable amount of time for practical purposes.

## II. PROBLEM MODEL

The following general compression model as well as the underlying combinatorial optimization model have originally been proposed in [1]. The input data is given as a vector of  $n$   $d$ -dimensional points  $V = \{v_1, \dots, v_n\}$  from a discrete domain  $\mathbb{D} = \{0, \dots, \tilde{v}^1 - 1\} \times \dots \times \{0, \dots, \tilde{v}^d - 1\}$ , with  $\mathbb{D} \subseteq \mathbb{N}^d$ . In our application these points correspond to the given minutiae data, where a single minutia is defined as a 4-tuple  $(x, y, \theta, t)$  with  $x$  and  $y$  being the Cartesian coordinates of the point,  $\theta$  its orientation and  $t$  its type. Within our compression model we usually only use  $x$  and  $y$  and eventually  $\theta$ , and store the further data unprocessed.

Our aim is to select a subset of exactly  $k$  points to be stored in the compressed fingerprint template; i.e., our compression is not lossless, but with a suitable selection of  $k$  this can usually be considered sufficient for a reliable verification, i.e. matching to the features extracted by a fingerprint scanner. For this purpose we start with a complete directed graph  $G = (V, A)$  with  $A = \{(u, v) \mid u, v \in V, u \neq v\}$  on which we search for an optimal directed tree (outgoing arborescence), spanning at least  $k$  nodes, by optimization. Thus, all nodes and arcs in the graph have a topological as well as a geometrical interpretation. In the following, we emphasize the context we primarily refer to by denoting these elements in normal and bold letters, respectively.

Furthermore, we use a small set  $T$  of template arcs which act as dictionary elements for our compression approach. Instead of directly storing the coordinate values of the mutual difference vectors of the tree nodes for each tree arc, we encode these arcs by a reference to a *template arc* in combination with a *correction vector* from a prespecified small domain. Compression is achieved by optimizing the selection of the  $k$  points, as well as building up a feasible dictionary of template arcs  $T$  of minimal cardinality.

Consequently, a solution to our problem consists of

- 1) an ordered set of template arcs  $T = (\mathbf{t}_1, \dots, \mathbf{t}_m) \in \mathbb{D}^m$ , and
- 2) an outgoing arborescence  $G_T = (V_T, A_T)$  with  $V_T \subseteq V$  and  $A_T \subseteq A$  connecting exactly  $|V_T| = k$  nodes, in which each tree arc  $(i, j) \in A_T$  has associated
  - a template arc index  $\kappa_{i,j} \in \{1, \dots, m\}$  and
  - a correction vector  $\delta_{i,j} \in \mathbb{D}'$  from a prespecified, small domain  $\mathbb{D}' \subseteq \mathbb{D}$  with  $\mathbb{D}' = \{0, \dots, \delta^1 - 1\} \times \dots \times \{0, \dots, \delta^d - 1\}$ .

For any two points  $\mathbf{v}_i$  and  $\mathbf{v}_j$  connected by a tree arc  $(i, j) \in A_T$  the relation

$$\mathbf{v}_j = (\mathbf{v}_i + \mathbf{t}_{\kappa_{i,j}} + \delta_{i,j}) \bmod \bar{\mathbf{v}}, \quad \forall (i, j) \in A_T, \quad (1)$$

must hold; i.e.  $\mathbf{v}_j$  can be derived from  $\mathbf{v}_i$  by adding the corresponding template and correction vectors. We use the modulo-calculation to avoid negative values and to be able to cross the domain-border within the arborescence. Our main objective now is to find a feasible solution with a smallest possible dictionary size  $m$ .

Details about encoding and decoding of the solution and the calculation and results regarding achieved compression ratios are given in the preceding work [1]. Here, we focus on an improved exact solution method, a new branch-and-cut-and-price algorithm.

#### A. The $k$ -Minimum Label Spanning Arborescence Problem

Based on the corresponding section in [1], we summarize the problem formulation in the following. To be able to choose the root node of the arborescence by optimization we extend  $V$  to  $V^+$  by adding an artificial root node 0. Further we extend  $A$  to  $A^+$  by adding arcs  $(0, i)$ ,  $\forall i \in V$ . We use the following variables for modeling the problem as an integer linear program (ILP):

- For each candidate template arc  $\mathbf{t} \in T^c$ , we define a variable  $y_t \in \{0, 1\}$ , indicating whether or not the arc is part of the dictionary  $T$ .
- Further we use variables  $x_{ij} \in \{0, 1\}$ ,  $\forall (i, j) \in A^+$ , indicating which arcs belong to the tree.
- To express which nodes are covered by the tree, we introduce variables  $z_i \in \{0, 1\}$ ,  $\forall i \in V$ .

Let  $A(t) \subset A$  denote the set of tree arcs a template arc  $t \in T^c$  is able to represent when considering the allowed domains for the correction vectors, and let  $T(a)$  be the set of template arcs that can be used to represent an arc  $a \in A$ , i.e.  $T(a) = \{t \in T^c \mid a \in A(t)\}$ . Hence, we can consider the template arcs  $t \in T^c$  as the *labels* of their corresponding arcs  $T(a)$ , revealing the strong relation to the minimum label spanning tree problem.

We can now formulate the  $k$ -MLSA problem as follows:

$$\min \sum_{t \in T^c} y_t \quad (2a)$$

$$\text{s.t.} \quad \sum_{t \in T(a)} y_t \geq x_a \quad \forall a \in A \quad (2b)$$

$$\sum_{i \in V} z_i = k \quad (2c)$$

$$\sum_{a \in A} x_a = k - 1 \quad (2d)$$

$$\sum_{i \in V} x_{(0,i)} = 1 \quad (2e)$$

$$\sum_{(j,i) \in A^+} x_{ji} = z_i \quad \forall i \in V \quad (2f)$$

$$x_{ij} \leq z_i \quad \forall (i, j) \in A \quad (2g)$$

$$x_{ij} + x_{ji} \leq z_i \quad \forall (i, j) \in A \quad (2h)$$

$$\sum_{a \in C} x_a \leq |C| - 1 \quad \forall \text{ cycles } C \text{ in } G, |C| > 2 \quad (2i)$$

$$\sum_{a \in \delta^-(S)} x_a \geq z_i \quad \forall i \in V, \forall S \subseteq V, \quad i \in S, 0 \notin S \quad (2j)$$

Inequalities (2b) ensure that for each used tree arc  $a \in A$  at least one valid template arc  $t$  is selected. Equalities (2c) and (2d) enforce the required number of nodes and arcs to be selected. Equation (2e) requires exactly one arc from the artificial root to one of the tree nodes, which will be the actual root node of the outgoing arborescence.

Equations (2f) state that selected nodes must have in-degree one. Inequalities (2g) ensure, that an arc may only be selected if its source node is selected as well. Inequalities (2h) forbid cycles of length two, and finally inequalities (2i) forbid all further cycles ( $|C| > 2$ ).

In order to strengthen the ILP we can additionally add (directed) connectivity-constraints, given by inequalities (2j), where  $\delta^-(S)$  represents the ingoing cut of node set  $S$ . These constraints ensure the existence of a path from the root 0 to any node  $i \in V$  for which  $z_i = 1$ , i.e. which is selected for connection. In principle, equations (2j) render (2f), (2g), (2h) and (2i) redundant [7], but using them jointly turned out to be sometimes beneficial in practice.

#### B. Candidate Template Arcs

The set of candidate template arcs  $T^c$  used in ILP (2) is, however, not explicitly given within the input data. The requirements to  $T^c$  are that it has to be sufficiently large to permit an overall optimal solution with regard to the pre-specified correction vector domain  $\tilde{\mathbf{d}}$ . Let  $B = \{\mathbf{v}_{ij} = (\mathbf{v}_j - \mathbf{v}_i) \bmod \tilde{\mathbf{v}} \mid (i, j) \in A\} = \{\mathbf{b}_1, \dots, \mathbf{b}_{|B|}\}$  be the set of different vectors we eventually have to represent. Let further  $D(\mathbf{t}) \subseteq \mathbb{D}$  be the subspace of all vectors a particular template arc  $\mathbf{t} \in \mathbb{D}$  is able to represent when considering the restricted domain  $\mathbb{D}'$  for the correction vectors, i.e.  $D(\mathbf{t}) = \{t^1, \dots, (t^1 + \delta^1 - 1) \bmod \tilde{v}^1\} \times \dots \times \{t^d, \dots, (t^d + \delta^d - 1) \bmod \tilde{v}^d\}$ . The subset of vectors from  $B$  that a particular template arc  $\mathbf{t}$  is able to represent is denoted by  $B(\mathbf{t}) = \{\mathbf{b} \in B \mid \mathbf{b} \in D(\mathbf{t})\}$ . The set of candidate template arcs  $T^c$  must have the property that all possible elements  $\mathbf{t}$  with mutually different and non-dominated  $B(\mathbf{t})$  should be included. Within this context a template-arc  $\mathbf{t}'$  is said to be dominated by  $\mathbf{t}''$  iff  $B(\mathbf{t}') \subset B(\mathbf{t}'')$ .

In a previous branch-and-cut approach [1] the set of candidate template arcs  $T^c$  has been computed in a relatively time-consuming preprocessing step. Within the approach presented in this work, this preprocessing is not necessary anymore.

Suitable candidate template arcs are on demand derived in the pricing-step, described in Section III.

### C. Cutting-plane Separation

The description of the cutting-plane separation again follows [1]. As there are exponentially many cycle elimination and connectivity inequalities (2i) and (2j), directly solving the ILP would be only feasible for very small problem instances. Instead, we apply branch-and-cut [8], i.e. we just start with the constraints (2b) to (2h) and add violated cycle elimination constraints and connectivity constraints only on demand during the optimization process.

Cycle elimination cuts (2i) can be easily separated by shortest path computations with Dijkstra's algorithm. Hereby we use  $(1 - x_{ij}^{\text{LP}})$  as the arc weights with  $x_{ij}^{\text{LP}}$  denoting the current value of the LP-relaxation for  $(i, j)$  in the current node of the branch-and-bound tree. We obtain cycles by iteratively considering each edge  $(i, j) \in A$  and searching for the shortest path from  $j$  to  $i$ . If the value of a shortest path plus  $(1 - x_{ij}^{\text{LP}})$  is less than 1, we have found a cycle for which inequality (2i) is violated. We add this inequality to the LP and resolve it. In each node of the branch-and-bound tree we perform these cutting plane separations until no further cuts can be found.

The directed connection inequalities (2j) strengthen our formulation. Compared to the cycle elimination cuts they lead to better theoretical bounds, i.e. a tighter characterization of the spanning-tree polyhedron [7], but their separation usually is computationally more expensive. We separate them by computing the maximum flow (and therefore minimum  $(0, i)$ -cut) from the root node to each of the nodes with  $z_i > 0$  as target node. If the value of this cut is less than  $z_i^{\text{LP}}$ , we have found an inequality that is violated by the current LP-solution. Our separation procedure utilizes Cherkassky and Goldberg's implementation of the push-relabel method for the maximum flow problem [9] to perform the required minimum cut computations.

The branch-and-cut algorithm has been implemented using C++ with CPLEX in version 11.2 [10].

## III. BRANCH-AND-CUT-AND-PRICE

The former branch-and-cut algorithm presented in [1] has two major shortcomings. First, the preprocessing method is quite time-consuming, and second, the large amount of template-arc-variables yields large LPs to be solved within every node of the branch-and-bound tree. These observations support the idea to develop a branch-and-cut-and-price (BCP) approach, where after starting with a small, very restricted set of template arcs, further template arcs are dynamically added on demand. This approach has for the first time been investigated in the theses [4]–[6].

Following the idea of the valid inequalities proposed in [11], we introduce inequalities

$$\sum_{t \in T(\Gamma^-(v_i))} y_t \geq z_i - x_{ri}, \quad \forall i \in V, \quad (3)$$

to provide (besides inequalities (2b)) further information for the pricing step, but also to further strengthen the LP. Inequalities (3) state that for each selected node except the artificial root node, the sum over the template-arc variables associated to the nodes ingoing arcs, must be greater or equal than one.

### A. Pricing Problem

Let  $\pi_a$  denote the dual variables corresponding to inequalities (2b), and  $\mu_j$  denote the dual variables corresponding to inequalities (3). The reduced costs for each template arc  $t$  are then given by

$$\bar{c}_t = 1 - \left( \sum_{a \in A(t)} \pi_a + \sum_{j \in \{v | (u,v) \in A(t)\}} \mu_j \right). \quad (4)$$

Any template arc with negative reduced costs  $\bar{c}_t$  may potentially improve the current objective function value; if there are no such template arcs, the solution cannot be further improved. We define the pricing problem as finding the template arc with maximal negative reduced costs.

*Definition 1 (Pricing Problem):*

$$t^* = \operatorname{argmin}_{t \in T} \left\{ 1 - \left( \sum_{a \in A(t)} \pi_a + \sum_{j \in \{v | (u,v) \in A(t)\}} \mu_j \right) \right\} \quad (5)$$

### B. Solving the Pricing Problem

A nice geometrical interpretation for the pricing problem arises, when considering the two-dimensional case. Each tree arc corresponds to a point in  $\mathbb{D}$  according to its associated geometric information. Furthermore, each point in  $\mathbb{D}$  in the same way corresponds to a potential template arc. Hence, we will use the terms tree/template arc and their corresponding points interchangeably within this section. All template arcs potentially representing an arbitrary tree arc  $\mathbf{b}$  must have their endpoint in the rectangle  $D(\mathbf{b} - \delta + \epsilon)$  with  $\mathbf{b}$  corresponding to its upper right corner, and  $\epsilon$  denoting the  $d$ -dimensional vector with all components being one. Let  $T'(\mathbf{b})$  denote this area whose points correspond to the potential template arcs able to represent  $\mathbf{b}$ . To each  $T'(\mathbf{b})$  we now associate the value

$$\zeta_b = \sum_{i \in \{a | a \in A \wedge a = b\}} \pi_i + \sum_{j \in \{v | (u,v) \in A \wedge (u,v) = b\}} \mu_j. \quad (6)$$

The first term on the right hand side in (6) corresponds to the sum of all dual values associated to the constraints for the tree arcs corresponding to  $\mathbf{b}$ , given by inequalities (2b). The second term results from the dual values of all nodes according to constraints (3) which are incident to a tree arc corresponding to  $\mathbf{b}$ . We can now imagine these rectangles  $T'(\mathbf{b})$  as transparently shaded with a color of intensity  $\zeta_b$ , where higher values corresponding to darker shades. See Fig. 1 for an example of two elements  $\mathbf{b}_1$  and  $\mathbf{b}_2$  and their corresponding regions  $T'(\mathbf{b}_1)$  and  $T'(\mathbf{b}_2)$  being drawn in the domain.

Let us now consider the situation of all  $\mathbf{b} \in B$  and their corresponding  $T'(\mathbf{b})$ , shaded accordingly with  $\zeta_b$ , being drawn in the area corresponding to the two-dimensional domain  $\mathbb{D}$ . Due to the transparency of the rectangles, regions of

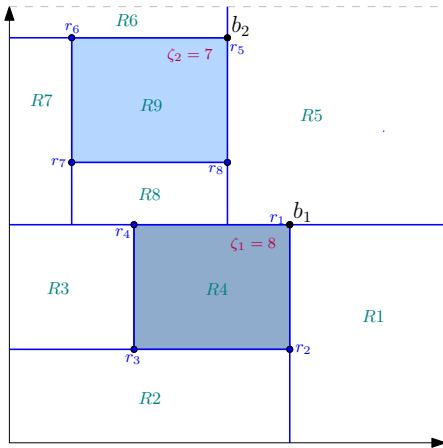


Fig. 1. Example of two elements  $b_1$  and  $b_2$  and corresponding regions  $T'(b_1)$  and  $T'(b_2)$  drawn in the domain  $\mathbb{D}$ . (Image with minor modifications taken from [4])

overlapping rectangles will obtain darker colors. Formally we define for each uniform region  $R$  a corresponding value

$$\zeta_R = \sum_{b \in A(t), t \in R} \zeta_b \quad (7)$$

for some arbitrary  $t$  being located in region  $R$ . Figure 2 shows an example with two overlapping elements  $b_i$ . We can now see that the pricing problem given by Definition 1 exactly corresponds to finding the darkest such area. This analogy remains valid even in the higher dimensional case if we use regions of corresponding dimensionality instead of areas with dimensionality two. The correspondence of the presented illustration to the pricing problem becomes evident by considering the correspondence of equation (6) to the two sums in (5). The only difference is that (6) is formulated in terms of unique points  $b$  and (5) in terms of template arcs  $t$ , which we actually want to determine.

Based on this observation, we now outline an algorithm to solve the pricing problem. This algorithm was primary subject of the diploma thesis [4]; for details according to the implementation of the algorithm, the reader is referred to this work.

The underlying data structure is a  $k$ -d tree [12] which is used to partition the domain into the corresponding regions resulting from  $T'(b)$ , for all  $b \in B$  and resulting overlapping regions. Here  $k$  denotes the number of dimensions to be used within the tree, and should not be mixed up with the number of nodes to be connected to the arborescence. However, as the term  $k$ -d tree is commonly used for this data structure, we refrain from referring to it as  $d$ -d tree. For convenience, we briefly review the principles of  $k$ -d trees. Their primary application is to store multidimensional data points and allow efficient range and nearest neighbor searches. The tree defines a hierarchical partitioning of the underlying domain. Each node of the binary tree defines a division of the subspace in which it is located into exactly two further subspaces.

Within each level  $l$  coordinate  $l \bmod k$  is used to define this subdivision. At the root node the whole domain is subdivided according to some coordinate of the first dimension. Each child node then defines a subdivision according to a coordinate of the second dimension, and so forth. For our purpose we define each node to have either two children, or to be a leaf node. In our case, a leaf node may either correspond to a region that cannot contain a template arc, or otherwise, a region that contains all possible template arcs representing a unique subset of elements  $b_i \in B$ .

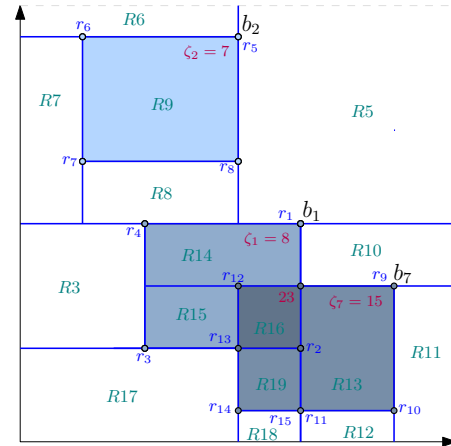


Fig. 2. This illustration shows the situation after the insertion of a further element ( $b_7$ ) into the segmentation tree shown in Figures 1 and 3. (Image with minor modifications taken from [4])

Figures 1 and 2 show examples of two and three elements being drawn in the domain, respectively. These figures illustrate how the domain is segmented into subregions according to  $T'(b_1)$  and  $T'(b_2)$  (and  $T'(b_7)$  in Figure 2). Corresponding  $k$ -d trees, which we will from now on call *segmentation trees*, are depicted in Figures 3 and 4. As each node of the tree subdivides its subspace into two subspaces, it defines a *hyperplane*, which we will also call *splitting-hyperplane*. In the two-dimensional examples of Figures 1 and 2 these hyperplanes correspond to lines. In the example of Figure 1 the first split is performed according to the second coordinate at point  $r_1 = b_1$ . Nodes  $r_i$  denote the coordinates corresponding to each node  $i$  of the segmentation tree. The area  $T'(b_1)$  is finally defined by nodes  $r_2$ ,  $r_3$  and  $r_4$ , area  $T'(b_2)$  by nodes  $r_5$ ,  $r_6$ ,  $r_7$  and  $r_8$ . In the figure all points  $r_i$  correspond to the corner points of areas  $T'(b)$  for all elements  $b$  in the tree, which is, however, an arbitrary decision for a better illustration. In fact, only one coordinate is required to define a hyperplane being orthogonal to the basis vector of the considered dimension, which is always the case in the segmentation tree. Besides the intermediate “splitting” nodes, the tree in Figure 3 also contains the leaf nodes, with corresponding regions depicted in Figure 1. The second example, given by Figures 2 and 4 shows the resulting tree after the insertion of element  $b_7$ . Again, splitting nodes and leaves (corresponding to regions) are contained in the visualization of the tree, as well as in



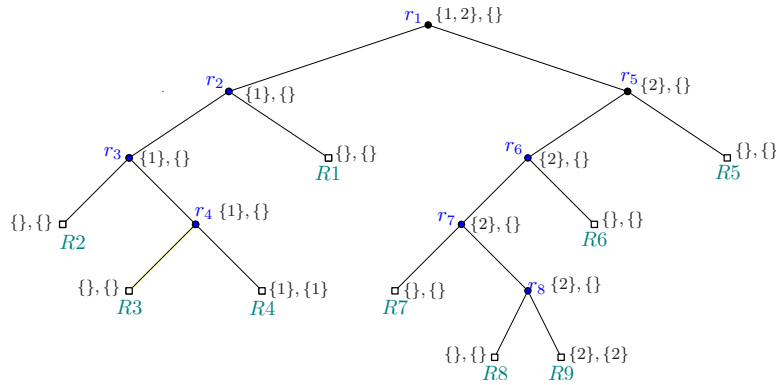


Fig. 3. Segmentation tree corresponding to the example shown in Figure 1. (Image credits: [4])

the corresponding illustration of the fragmented domain. To build up the whole tree, regions  $T'(b)$  for all  $b \in B$  are iteratively inserted. For each such  $T'(b)$  we need to find the correct position for inserting it into the tree. This is done by checking at each tree node  $r$  if  $T'(b)$  is entirely located in one of the subspaces defined by  $r$ . If a region is entirely contained in a region defined by a current leaf of the tree, this leaf is replaced with an according subtree corresponding to the splitting hyperplanes required to properly define  $T'(b)$ . However,  $T'(b)$  is part of both subspaces defined by the current node  $r$ , we need to split  $T'(b)$  accordingly, and insert the resulting subregions into both branches of  $r$ . Having now described, how the segmentation tree is created, we focus on how the tree can be used to efficiently search for the best template arc.

At this point we assume that the whole segmentation tree has been created in advance. As we will see later, this is not a real requirement. Our goal is to find the region  $R$  with maximum  $\zeta_R$ , which is the solution to the pricing problem. As the pricing problem needs to be solved many times, the search has to be efficient. In particular we want to avoid to assign  $\zeta_B$  to all leaves (corresponding to regions)  $B$  in the tree according to the values  $\zeta_b$  derived by the dual values. Therefore the search is based on upper and lower bounds used to prune branches at an early stage. Let  $R(r)$  denote the subspace corresponding to node  $r$  of the segmentation tree. Upper and lower bounds for each node  $r$  of the tree can be derived based on the following definitions.

*Definition 2 (Upper Bound Set):* The upper bound set is given by all elements  $b \in B$  which can be represented by some potential template arc in the subspace corresponding to tree node  $r$ .

$$UB(r) = \{b \in B \mid \exists t \in R(r) \wedge b \in B(t)\}$$

*Definition 3 (Lower Bound Set):* The lower bound set is given by all elements  $b \in B$  which can be represented by all potential template arcs in the subspace corresponding to tree node  $r$ .

$$LB(r) = \{b \in B \mid \forall t \in R(r) \wedge b \in B(t)\}$$

These bound sets are stored for each node of the search tree. In Figures 1 and 2 these sets are denoted in braces at each node.

Based on these sets we can immediately derive numeric bounds, based on the dual values.

*Definition 4 (Upper Bound):*

$$ub(r) = \sum_{b \in UB(r)} \zeta_b$$

*Definition 5 (Lower Bound):*

$$lb(r) = \max_{b \in UB(r)} \zeta_b$$

The search process is performed based on these upper and lower bounds. Starting at the root node, the set  $B$  is divided into two not necessarily disjoint sets. These sets  $UB(r)$  correspond to the nodes which are representable by some template arc of the subspaces introduced by the splitting-hyperplane defined by the current tree node  $r$ . With  $ub(r)$  we directly obtain a numeric value being the upper bound for this particular branch. A lower bound is given by  $lb(r)$ , i.e. the element with maximal  $\zeta_b$  in this branch. For each node we check if  $UB(r) = LB(r)$  which implies that we have found a leaf node. A global lower bound  $lb^*$  is used to prune the search tree, as we do not have to follow branches with  $ub(r) < lb^*$ . Initialization of the global lower bound can be performed with  $lb^* = \max_{b \in B} \zeta_b$ . The search strategy to be used is *best first search* based on the upper bounds  $ub(r)$ .

Within the description of the algorithm, we have omitted many implementation issues. One important aspect to be considered is the fact that regions may cross the domain border. This needs to be checked in advance, and corresponding subregions must be inserted in this case. Furthermore a lot of design issues are involved in order to implement the bounding procedure efficiently. Also the reconstruction of the coordinate values of the corner points of each region requires to take care of some special cases. For a detailed presentation and analysis of this issues we refer to [4].

A further substantial improvement of the overall process can be achieved if the entire tree is not completely built in advance, but rather in a dynamic on demand way during the

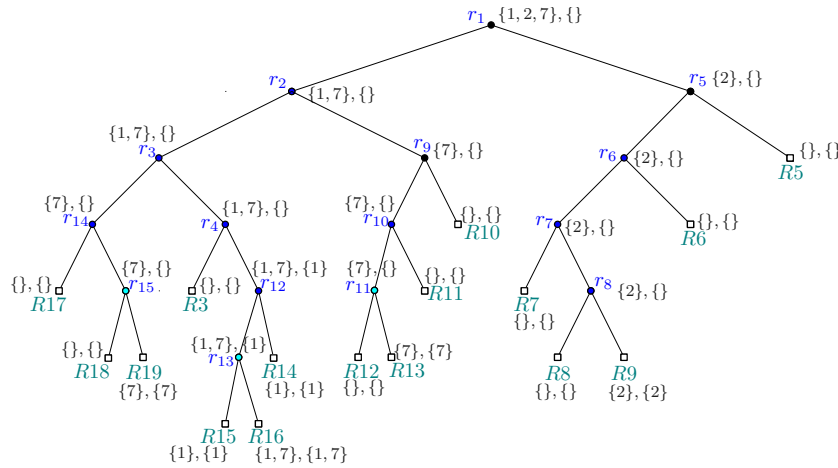


Fig. 4. Segmentation tree corresponding to the example shown in Figure 2. (Image credits: [4])

search process. Each time the search is according to the bounds directed toward a certain branch of the tree, we check if this branch has already been created. If this is not the case, it is expanded as needed. Hence construction and traversing the tree is performed in an intertwined way. This has not only the advantage of the initial construction step to be omitted, but will also result in smaller trees to operate with. As certain regions of the domain will not contain any useful template arcs, corresponding branches are unlikely to be created during the whole BCP solution process, saving space and time.

Corresponding pseudocodes are omitted within this presentation, as they would require a more detailed formal description and notation. In the following section we show how this algorithmic framework for solving the pricing problem can be used within a branch-and-cut-and-price approach.

### C. Branch-and-Cut-and-Price Algorithm

The first step of the entire branch-and-cut-and-price (BCP) is to determine a feasible starting solution. Any connected subgraph of  $k$  nodes is sufficient for this purpose. Hence, we determine a starting solution by connecting arbitrary  $k$  nodes by a star-shaped spanning tree, assign big values to the dual variables corresponding to this set of arcs, and use the pricing algorithm to determine a feasible starting solution.

The restricted master problem (RMP) is defined according to the ILP from Section II-A, however the entire set  $T^c$  is replaced by  $T^p$  denoting the set of template arc variables that have already been priced in. Within each node of the branch-and-bound-tree directed connection cuts and cycle elimination cuts are separated to obtain a feasible LP-relaxation. Afterwards new template arc variables are priced in as long as such variables with negative reduced costs according to Equation (5) can be found and no further cutting-planes can be added. It turned out to be advantageous to add all variables with negative reduced costs within each pricing iteration.

## IV. RESULTS

In this section we present the results of our computational experiments with the outlined branch-and-cut-and-price algorithm. For this purpose two different data sets have been used. The first set of 20 instances was provided by the Fraunhofer Institute Berlin and is in the following referred to as *Fraunhofer Templates*. Furthermore 14 randomly selected instances from the U.S. National Institute of Standards and Technology [13] have been used.

All test runs have been performed on an Intel Core 2 Quad running at 2.83 GHz with 8 GB RAM and Ubuntu 11.04. The branch-and-cut-and-price algorithm has been implemented in C++ within the SCIP framework [14], and CPLEX in version 11.2 [10], which is also used for the comparison to the branch-and-cut (BC) algorithm from [1].

For each run a time-limit of two hours has been imposed. Table I shows average solution values for various parameter settings ( $k$ ,  $\tilde{\delta}$ ) and groups of instances. These averages are taken over all instances that have been solved within the time limit (indicated in the last column). The Fraunhofer instances with  $|V| < 30$  are not included in the corresponding groups with  $k = 30$ . Column “pit” reports the average numbers of pricing iterations, column “pvar” the average numbers of priced in variables, and column “bbn” the average branch and bound nodes. Column “cuts” reports the numbers of applied cuts, which consist of directed connection cuts (“DCC”) and cycle elimination cuts (“CEC”).

Table II shows the comparison of the new BCP algorithm to the branch-and-cut algorithm presented in [1]. The average running times of the BC algorithm do not include the preprocessing step. The reported average running times and numbers of branch-and-bound nodes are not directly comparable, as not always the same number of instances has been solved by both algorithms. The BCP method is, however, clearly superior to the BC algorithm w.r.t. the number of solved instances, and also yields to significantly lower average running times and numbers of branch-and-bound nodes.

TABLE I  
RESULTS OF THE BRANCH-AND-CUT-AND-PRICE ALGORITHM. AVERAGE VALUES FOR ALL SOLVED INSTANCES IN THE PARTICULAR GROUP ARE REPORTED.

Instances	Parameters	avg t[s]	pit	pvar	bbn	cuts	DCC	CEC	inst.solved
Fraunhofer	$\tilde{\delta}^\top = (10, 10, 10), k = 20$	7.7	134	86	39	113	100	13	20/20
Fraunhofer	$\tilde{\delta}^\top = (10, 10, 10), k = 30$	4.7	90	65	18	188	167	21	18/18
NIST	$\tilde{\delta}^\top = (10, 10, 10), k = 40$	2945.0	20006	311	19636	1150	1061	89	5/15
NIST	$\tilde{\delta}^\top = (10, 10, 10), k = 80$	886.6	11950	174	11616	8032	7625	407	12/15
NIST	$\tilde{\delta}^\top = (10, 10, 10), k =  V $	237.1	1665	162	1464	3702	3400	302	11/14
Fraunhofer	$\tilde{\delta}^\top = (20, 20, 20), k = 20$	23.0	304	174	120	164	134	30	20/20
Fraunhofer	$\tilde{\delta}^\top = (20, 20, 20), k = 30$	14.4	253	162	79	280	224	56	18/18
NIST	$\tilde{\delta}^\top = (20, 20, 20), k = 40$	890.0	1067	632	399	1192	972	220	6/15
NIST	$\tilde{\delta}^\top = (20, 20, 20), k = 80$	987.8	691	412	257	1532	1356	177	14/15
NIST	$\tilde{\delta}^\top = (20, 20, 20), k =  V $	232.7	496	353	125	1653	1493	159	14/14
Fraunhofer	$\tilde{\delta}^\top = (30, 30, 30), k = 20$	132.3	1379	594	762	410	305	105	20/20
Fraunhofer	$\tilde{\delta}^\top = (30, 30, 30), k = 30$	28.0	537	311	207	552	447	105	18/18
NIST	$\tilde{\delta}^\top = (30, 30, 30), k = 40$	2970.0	2647	960	1591	3724	2995	729	6/15
NIST	$\tilde{\delta}^\top = (30, 30, 30), k = 80$	2219.3	1873	767	988	8688	7608	1079	12/15
NIST	$\tilde{\delta}^\top = (30, 30, 30), k =  V $	2318.3	3032	986	1997	4580	4124	457	11/14
Fraunhofer	$\tilde{\delta}^\top = (40, 40, 40), k = 20$	163.3	2069	1119	936	212	167	44	20/20
Fraunhofer	$\tilde{\delta}^\top = (40, 40, 40), k = 30$	148.3	1195	709	474	273	220	53	18/18
NIST	$\tilde{\delta}^\top = (40, 40, 40), k = 40$	3139.8	3556	1124	2223	9304	7193	2111	4/15
NIST	$\tilde{\delta}^\top = (40, 40, 40), k = 80$	1808.5	2598	982	1492	9535	8368	1166	6/15
NIST	$\tilde{\delta}^\top = (40, 40, 40), k =  V $	2844.3	5258	1265	3919	5940	5154	786	5/14
Fraunhofer	$\tilde{\delta}^\top = (50, 50, 50), k = 20$	47.5	547	451	88	118	97	21	18/20
Fraunhofer	$\tilde{\delta}^\top = (50, 50, 50), k = 30$	83.6	886	645	230	285	223	62	18/18
NIST	$\tilde{\delta}^\top = (50, 50, 50), k = 40$	2996.8	3566	1477	1901	7411	5832	1579	2/15
NIST	$\tilde{\delta}^\top = (50, 50, 50), k = 80$	1313.5	3546	1089	2397	4589	3995	594	2/15
NIST	$\tilde{\delta}^\top = (50, 50, 50), k =  V $	4190.9	7542	1747	5645	13019	11098	1921	1/14
Fraunhofer	$\tilde{\delta}^\top = (60, 60, 60), k = 20$	588.4	1869	1656	202	173	134	39	16/20
Fraunhofer	$\tilde{\delta}^\top = (60, 60, 60), k = 30$	65.5	791	574	199	472	358	114	18/18
NIST	$\tilde{\delta}^\top = (60, 60, 60), k = 40$	5177.2	5740	1951	3712	2725	2091	634	1/15
NIST	$\tilde{\delta}^\top = (60, 60, 60), k = 80$	2039.9	3928	1450	2469	659	583	76	3/15
NIST	$\tilde{\delta}^\top = (60, 60, 60), k =  V $	4066.7	6836	2083	4709	3764	3292	472	4/14

The results clearly show that the new BCP approach is able to solve a significantly larger number of instances and also requires shorter running times on average for most classes of instances.

## V. CONCLUSIONS

In this work we have presented a branch-and-cut-and-price framework to solve the problem of compressing a relatively small unordered set of multidimensional points with the application background of embedding fingerprint minutiae data into passport images by watermarking techniques as an additional security feature. Compared to a previously used exact branch-and-cut algorithm, a significant speedup of solving the underlying combinatorial optimization problem ( $k$ -MLSA problem) could be achieved. Furthermore the preprocessing step being necessary for the preceding approach needs not to be performed anymore. As a result more instances can be

solved to proven optimality within the considered time limit by the new method.

Although the overall compression ratios achieved by our particular model are rather limited, they are clearly superior to other popular compression mechanisms, which cannot perform any compression on the considered data at all. Further improvements regarding compression ratios can possibly be achieved by devising refined models and algorithms. However, in consideration of being a new approach to data compression by combinatorial optimization techniques, as well as being a novel approach of directly exploiting the property that the order of the underlying data needs not to be preserved, our approach can be regarded a successful proof-of-concept to be able to compress weakly structured data sets and appears to be a promising origin for further research in the field of combinatorial optimization based compression methods.

TABLE II  
COMPARISON OF THE RESULTS ACHIEVED BY BRANCH-AND-CUT-AND-PRICE AND THE BRANCH-AND-CUT.

Instances	Parameters	branch-and-cut			branch-and-cut-and-price		
		avg t[s]	bbn	inst.solved	avg t[s]	bbn	inst.solved
Fraunhofer	$\tilde{\delta}^\top = (10, 10, 10), k = 20$	3.8	27	20/20	7.7	39	20/20
Fraunhofer	$\tilde{\delta}^\top = (10, 10, 10), k = 30$	4.4	253	18/18	4.7	18	18/18
NIST	$\tilde{\delta}^\top = (10, 10, 10), k = 40$	2889.4	1574	1/15	2945.0	19636	5/15
NIST	$\tilde{\delta}^\top = (10, 10, 10), k = 80$	1416.4	6371	4/15	886.6	11616	12/15
NIST	$\tilde{\delta}^\top = (10, 10, 10), k =  V $	498.7	623	3/14	237.1	1464	11/14
Fraunhofer	$\tilde{\delta}^\top = (20, 20, 20), k = 20$	28.0	579	20/20	23.0	120	20/20
Fraunhofer	$\tilde{\delta}^\top = (20, 20, 20), k = 30$	11.3	191	18/18	14.4	79	18/18
NIST	$\tilde{\delta}^\top = (20, 20, 20), k = 40$	2220.7	1691	8/15	890.0	399	6/15
NIST	$\tilde{\delta}^\top = (20, 20, 20), k = 80$	537.2	157	13/15	987.8	257	14/15
NIST	$\tilde{\delta}^\top = (20, 20, 20), k =  V $	322.4	69	13/14	232.7	125	14/14
Fraunhofer	$\tilde{\delta}^\top = (30, 30, 30), k = 20$	76.6	1513	20/20	132.3	762	20/20
Fraunhofer	$\tilde{\delta}^\top = (30, 30, 30), k = 30$	98.7	2657	18/18	28.0	207	18/18
NIST	$\tilde{\delta}^\top = (30, 30, 30), k = 40$	2922.5	3860	4/15	2970.0	1591	6/15
NIST	$\tilde{\delta}^\top = (30, 30, 30), k = 80$	1291.0	880	9/15	2219.3	988	12/15
NIST	$\tilde{\delta}^\top = (30, 30, 30), k =  V $	2281.8	2515	8/14	2318.3	1997	11/14
Fraunhofer	$\tilde{\delta}^\top = (40, 40, 40), k = 20$	241.4	5471	20/20	163.3	936	20/20
Fraunhofer	$\tilde{\delta}^\top = (40, 40, 40), k = 30$	256.4	3493	18/18	148.3	474	18/18
NIST	$\tilde{\delta}^\top = (40, 40, 40), k = 40$	2712.3	4218	2/15	3139.8	2223	4/15
NIST	$\tilde{\delta}^\top = (40, 40, 40), k = 80$	1071.6	1296	5/15	1808.5	1492	6/15
NIST	$\tilde{\delta}^\top = (40, 40, 40), k =  V $	2762.4	4232	3/14	2844.3	3919	5/14
Fraunhofer	$\tilde{\delta}^\top = (50, 50, 50), k = 20$	292.4	2246	13/20	47.5	88	18/20
Fraunhofer	$\tilde{\delta}^\top = (50, 50, 50), k = 30$	604.1	6606	9/18	83.6	230	18/18
NIST	$\tilde{\delta}^\top = (50, 50, 50), k = 40$	3681.7	5030	1/15	2996.8	1901	2/15
NIST	$\tilde{\delta}^\top = (50, 50, 50), k = 80$	1611.7	5991	2/15	1313.5	2397	2/15
NIST	$\tilde{\delta}^\top = (50, 50, 50), k =  V $	1711.7	3633	1/14	4190.9	5645	1/14
Fraunhofer	$\tilde{\delta}^\top = (60, 60, 60), k = 20$	864.7	2425	12/20	588.5	202	16/20
Fraunhofer	$\tilde{\delta}^\top = (60, 60, 60), k = 30$	710.1	5504	11/18	65.5	199	18/18
NIST	$\tilde{\delta}^\top = (60, 60, 60), k = 40$	1854.6	5652	1/15	5177.2	3712	1/15
NIST	$\tilde{\delta}^\top = (60, 60, 60), k = 80$	3073.4	4121	1/15	2039.9	2469	3/15
NIST	$\tilde{\delta}^\top = (60, 60, 60), k =  V $	3069.7	4544	1/14	4066.7	4709	4/14

## REFERENCES

- [1] A. M. Chwatal, G. R. Raidl, and K. Oberlechner, "Solving a  $k$ -node minimum label spanning arborescence problem to compress fingerprint templates," *Journal of Mathematical Modelling and Algorithms*, vol. 8, pp. 293–334, 2009.
- [2] R.-S. Chang and S.-J. Leu, "The minimum labeling spanning trees," *Information Processing Letters*, vol. 63, no. 5, pp. 277–282, 1997.
- [3] M. Chimani, M. Kandyba, I. Ljubić, and P. Mutzel, "Obtaining optimal  $k$ -cardinality trees fast," *Journal of Experimental Algorithmics*, vol. 14, pp. 5.1–5.23, 2010.
- [4] C. Thöni, "Compressing fingerprint templates by solving the  $k$ -node minimum label spanning arborescence problem by branch-and-price," Master's thesis, Vienna University of Technology, Vienna, Austria, 2010.
- [5] K. Oberlechner, "Solving a  $k$ -node minimum label spanning arborescence problem with exact and heuristic methods," Master's thesis, Vienna University of Technology, Vienna, Austria, 2010.
- [6] A. M. Chwatal, "On the minimum label spanning tree problem: Solution methods and applications," Ph.D. dissertation, Vienna University of Technology, 2010.
- [7] T. Magnanti and L. Wolsey, "Optimal trees," *Handbook in Operations Research and Management Science*, vol. Network Models, pp. 503–615, 1995.
- [8] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*. Wiley-Interscience, November 1999. [Online]. Available: <http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0471359432>
- [9] B. V. Cherkassky and A. V. Goldberg, "On implementing the push-relabel method for the maximum flow problem," *Algorithmica*, vol. 19, no. 4, pp. 390–410, 1997. [Online]. Available: [citeseer.ist.psu.edu/cherkassky94implementing.html](http://citeseer.ist.psu.edu/cherkassky94implementing.html)
- [10] ILOG Concert Technology, CPLEX, "ILOG," <http://www.ilog.com>, version 11.0.
- [11] A. M. Chwatal and G. R. Raidl, "Solving the minimum label spanning tree problem by mathematical programming techniques," *Advances in Operations Research*, 2011, (in press).
- [12] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.
- [13] Garris M. D. and McCabe R. M., "NIST special database 27: Fingerprint minutiae from latent and matching tenprint images." National Institute of Standards and Technology, Tech. Rep., 2000.
- [14] SCIP – Solving Constraint Integer Programs, "ILOG," <http://scip.zib.de/>, version 1.2.

# Improved asymptotic analysis for SUMT methods

Jean-Pierre Dussault  
 Département d'informatique  
 Faculté des Sciences  
 Université de Sherbrooke  
 Sherbrooke (Québec) CANADA J1K 2R1  
 Email: Jean-Pierre.Dussault@USherbrooke.ca

**Abstract**—We consider the SUMT (*Sequential Unconstrained Minimization Technique*) method using extrapolations to link successive unconstrained sub-problems. The case when the extrapolation is obtained by a first order Taylor estimate and Newton's method is used as a correction in this predictor-corrector scheme was analyzed in [1]. It yields a two-steps super-linear asymptotic convergence with limiting order of  $\frac{4}{3}$  for the logarithmic barrier and order two for the quadratic loss penalty.

We explore both lower order variants (approximate extrapolations correction computations) as well as higher order variants (second order and further) Taylor estimate.

First, we address inexact solutions of the linear systems arising within the extrapolation and the Newton's correction steps. Depending on the inexactness allowed, asymptotic convergence order reduces, more severely so for interior variants.

Second, we investigate the use of higher order path following strategies in those methods. We consider the approach based on a high order expansion of the so-called central path, somewhat reminiscent of Chebyshev's third order method and its generalizations. The use of higher order representation of the path yields spectacular improvement in the convergence property, even more so for the interior variants.

## I. INTRODUCTION

WE CONSIDER non linear programs (NLP) of the form

$$\begin{aligned} \min_{x \in \mathbb{R}^n} f(x) \\ \text{subject to } g(x) \leq 0 \end{aligned} \quad (1)$$

or

$$\begin{aligned} \min_{x \in \mathbb{R}^n} f(x) \\ \text{subject to } g(x) = 0 \end{aligned} \quad (2)$$

with  $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ .

We will address both formulations using SUMT, Eq.(1) using the classical log barrier method and Eq.(2) will be developed using the quadratic loss penalty function. A slight emphasis is put on the logarithmic barrier variant.

Fiacco and McCormick [5] pioneered the study of SUMT, and obtained the important result that close to a solution, the unconstrained sub-problems induce differential trajectories, and proposed the use of extrapolation to follow the trajectories. In linear programming, Mehrotra [6] popularized the use of so called predictor-corrector algorithms, intimately related to the extrapolations of the SUMT trajectory.

This research was partially supported by NSERC grant OGP0005491

We present asymptotic results related to inexact versions of both variants, looking for super-linear convergence of order lower than 2, as well as high order extrapolations to aim for faster asymptotic convergence.

Results from [1] state that interior variants achieve a limiting convergence order of two-steps  $\frac{4}{3}$  while exterior variants reach a limiting 2-steps quadratic order. We call two-steps convergence since the convergence order requires the solution of two linear systems, one to compute the extrapolation and another one to compute the Newton step.

Asymptotic order is not all, and interior variants are known to yield polynomial complexity in a wide variety of contexts while exterior variants could be plagued by combinatorial aspects related to active set identification when applied to inequality constrained problems.

We will first analyze inexact versions of SUMT, and conclude that the deterioration with respect to the asymptotic order is more severe for interior variants, exterior variants' degradation being benign.

On the other hand, high order versions bring both variants comparable with respect to their asymptotic behavior. For example, the second order extrapolation allows to bypass any Newton correction asymptotically. The second order correction involves solving two linear systems, but both systems share the same matrix. Therefore, only one factorization is require. For unstructured dense problems, factorisation of a matrix in a linear system entails  $\mathcal{O}(n^3)$  arithmetic operations while solving the system needs a further  $\mathcal{O}(n^2)$  operations. The second order extrapolation requires only one factorization, much better than two independant linear systems.

Finally, we propose an approximate high order version based on the Shamanskii approach which is simple to implement and shares the good asymptotic improvements of the exact high order versions.

## II. SUMT BASIC EXTRAPOLATIONS

We now recall two path following approaches, allowing to settle our notation and present some basic properties. Path following methods are related to the so-called central path, and involves steps named as predictor, corrector, centrality corrector, higher order predictor. We will detail below the terms we will use.

We stress that the results for the exterior quadratic loss penalty function and the interior log-barrier function are very

similar. The only difference is the limiting convergence order,  $\frac{4}{3}$  for the interior approach and 2 for the exterior penalty.

#### A. Log-Barrier

The log barrier approach to solve Eq.(1) consists in solving a sequence of sub-problems of the form

$$\min_{g(x)<0} \phi(x, \rho_k) = f(x) - \rho_k \sum \log(-g_i(x)) \quad (3)$$

in the interior of the feasible set  $E = \{x : g(x) \leq 0\}$ . Writing out the optimality conditions

$$\nabla_x \phi(x, \rho_k) = \nabla f(x) - \rho_k \sum \frac{1}{g_i(x)} \nabla g_i(x) = 0 \quad (4)$$

and by making the substitution  $y_i = \frac{\rho_k}{g_i(x)}$  and introducing the residual  $\Phi_k$ , one arrives at the primal-dual equations

$$\Theta(x, y, \Phi_k, \rho_k) = \begin{cases} \nabla f(x) - y \nabla g(x) = \Phi_k \\ y G(x) = \rho_k e \end{cases} \quad (5)$$

where  $e = (1, 1 \dots 1)^t$  and  $G(x) = \text{diag}(g_i(x))$ .

Under suitable assumptions, this last system of equations implicitly defines differentiable trajectories  $x(\rho, \Phi)$  and  $y(\rho, \Phi)$  close to  $\rho = 0$ .

#### B. Quadratic loss

The quadratic loss approach to solve Eq.(2) consists in solving a sequence of sub-problems of the form

$$\min \phi(x, \rho_k) = f(x) + \frac{1}{\rho_k} \|g(x)\|^2. \quad (6)$$

Writing out the optimality conditions

$$\nabla_x \phi(x, \rho_k) = \nabla f(x) + \frac{g(x)^t}{\rho_k} \nabla g(x) = 0 \quad (7)$$

and by making the substitution  $y = \frac{g(x)^t}{\rho_k}$  and introducing the residual  $\Phi_k$ , one arrives at the primal-dual equations

$$\Theta(x, y, \Phi_k, \rho_k) = \begin{cases} \nabla f(x) + y \nabla g(x) = \Phi_k \\ g(x) = \rho_k y \end{cases} \quad (8)$$

Under suitable assumptions, this last system of equations implicitly defines differentiable trajectories  $x(\rho, \Phi)$  and  $y(\rho, \Phi)$  close to  $\rho = 0$ .

#### C. Common properties

Penalty and barriers trajectories share much properties. Those may be expressed conveniently using the  $\Theta$  function in a unified way. In the following result,  $g_{I^*}$  refers to the active constraints in the log barrier case, and the whole  $g$  vector in the quadratic penalty case.

*Theorem 2.1:* [1] Let  $x^*$  be a regular point of the constraints  $g_{I^*}(x) = 0$  which satisfies to the second order sufficient optimality conditions for (1) as well as to the strict complementarity condition  $y_{I^*} > 0$  for the log barrier case. If the functions  $f$  and  $g$  are  $C^p(\mathbb{R}^n)$ , then there exists differentiable trajectories  $x(\rho, \Phi)$  and  $y(\rho, \Phi)$  of class  $C^{p-1}(\mathbb{R}^n)$  such that

- 1)  $x(0, 0) = x^*$  and  $y(0, 0) = y^*$ ;

- 2) if  $\rho$  and  $\|\Phi\|$  are small enough,  $x(\rho, 0)$  satisfies to the second order sufficient optimality conditions for the penalized sub-problems  $\min f(x) - \rho \sum_{i=1}^m \log(-g_i(x))$ , where  $x(r, \Phi)$ ,  $y(r, \Phi)$  are solutions of the following equations:

$$\Theta(x, y, \Phi, 0) = 0 \quad (9)$$

Moreover, the following bounds hold asymptotically:

- a)  $\|x(\rho, \Phi) - x^*\| \sim \mathcal{O}(\max(\rho, \|\Phi\|))$ ;
- b)  $\|y(\rho, \Phi) - y^*\| \sim \mathcal{O}(\max(\rho, \|\Phi\|))$ ;
- c)  $\|g_{I^*}(x(\rho, \Phi))\| \sim \mathcal{O}(\rho)$ .

*Remark 2.1:* Although we use primal-dual equations, in this SUMT variant, the dual variables  $y$  are dependent on the primal  $x$ , so that global convergence is inferred from the fact that the penalty or barrier is minimized (using globally convergent algorithms), allowing to prove that cluster points of the generated sequence are indeed stationary.

We will denote  $G(x) = \text{diag}(g_i(x))$  and for the log barrier,

$$\Phi(x, \rho) = \nabla_x \phi(x, \rho) \quad (10)$$

$$= \nabla f(x) - \sum \frac{\rho}{g_i(x)} \nabla g_i(x) \quad (11)$$

$$= \nabla f(x) - \rho \nabla g^t(x) G(x)^{-1} e. \quad (12)$$

$\phi$  is closely related to the Lagrangian  $l(x, \lambda) = f(x) + g(x)\lambda$ , and by defining  $\lambda = -\rho G(x)^{-1} e$ , i.e.  $\lambda_i = -\frac{\rho}{g_i(x)}$ ,  $\nabla_x l(x, \lambda) = L(x, \lambda) = \nabla f(x) + \lambda \nabla g(x) = \Phi(x, \rho)$ .

Similarly, for the quadratic penalty,  $\lambda = \frac{g(x)^t}{\rho}$  and  $\Phi(x, \rho) = \nabla f(x) + \frac{g(x)^t}{\rho} \nabla g(x)$ .

We are concerned with approximate solutions  $x(\rho, r)$  which satisfy  $\Phi(x(\rho, r), \rho) = r$ . In the sequel the residual  $r$  is assumed to satisfy  $\|r\| \sim \rho$ .

The basic predictor-corrector path following approach consists then in having an estimate  $x(\rho, r)$  which satisfy  $\Phi(x(\rho, r), \rho) = r$  and then iterate the following two steps:

pred	extrapolate $\hat{x}^1 = x + \frac{\partial x}{\partial \rho}(\rho^+ - \rho) + \frac{\partial x}{\partial r}(-r)$
corr	perform Newton corrections from $\hat{x}^1$ on the problem Eq.(6) or Eq.(3) for $\rho^+$ until $\ \Phi(x, \rho^+)\  \leq \rho^+$ .

For this basic scheme and the log barrier case, a single Newton correction asymptotically yields  $x(\rho^+, r^+)$  with  $\|\rho^+\| \leq \rho^+$  provided that  $\frac{\rho^+}{\rho^{\frac{4}{3}}} \rightarrow 0$  yielding a two-steps super-linear convergence of limiting order  $\frac{4}{3}$ . If one is prepared to perform two Newton corrections, then the limiting order is improved to  $\frac{\rho^+}{\rho^{\frac{5}{3}}} \rightarrow 0$  [4] yielding a three-steps super-linear convergence of limiting order  $\frac{5}{3}$ . Using a measure similar to Ostrovski efficiency, the optimal strategy for this family of algorithms is to aim for two Newton corrections following an extrapolation [4].

For the quadratic loss case, a single Newton correction asymptotically yields  $x(\rho^+, r^+)$  with  $\|\rho^+\| \leq \rho^+$  provided that  $\frac{\rho^+}{\rho^2} \rightarrow 0$  yielding a two-steps super-linear convergence of limiting quadratic order.

#### D. Terminology

First order extrapolations are usually named “predictor” steps. Higher order terms are sometimes named “corrections” to the predictor, but we will stick to the terminology “higher order”. Once a predictor (of arbitrary order) is computed, sometimes it is necessary to perform Newton iterations, referred to as a “corrector” steps. This is sometimes named “centrality correction” steps.

Predictor steps aim at changing the trajectory parameter  $\rho$  to a smaller value while corrector steps aim at improving the parametric solution for a given  $\rho$ -value.

### III. INEXACT VERSIONS

Since extrapolations (predictor steps) and Newton corrections are related to Newton steps, one may devise strategies to approximately compute the steps. We address in this section the asymptotic convergence order of such variants where both (predictors and correctors) steps are computed approximately.

#### A. Inexact SUMT

We now address inexact extrapolations and corrections. By solving approximately the equations defining the first order extrapolation  $\hat{x}^1$ , we get an extrapolate, denoted  $\hat{x}$  with  $\|\hat{x} - x(\rho^+, 0)\| \sim \rho^{a+1}$ . If  $a = 1$ , we get as good a prediction as  $\hat{x}^1$  while if  $\hat{x}$  is computed cheaply, we insist to at least obtain an order  $a + 1$  for some  $a > 0$ .

Assume that the Newton’s direction is computed approximately such that  $\nabla_x \Phi(\hat{x}, \rho^+) d_N + \Phi(\hat{x}, \rho^+) = R$  with  $\|R\| \leq \rho^{1+c} = \gamma$ .

*Lemma 3.1:* Let  $\hat{x}$  such that  $\|\hat{x} - x(\rho^+, 0)\| \sim \rho^{a+1}$ . Then,  $d_N \sim \mathcal{O}(\gamma + \rho^{a+1})$ .

1) *Details for the log barrier:* We first provide a proof of lemma 3.1

*Proof:* The primal-dual Newton’s direction is written

$$\begin{pmatrix} \nabla_x L(\hat{x}, \hat{\lambda}) & \nabla \hat{g} \\ \hat{\Lambda} \nabla \hat{g}^t & \hat{G} \end{pmatrix} \begin{pmatrix} d_x \\ d_y \end{pmatrix} = \begin{pmatrix} -L(\hat{x}, \hat{\lambda}) + R \\ 0 = \hat{G} \hat{\lambda} - \rho^+ e \end{pmatrix}, \quad (13)$$

where  $\hat{\lambda}_i = \frac{\rho^+}{\hat{g}_i}$ . Define also  $\lambda_i = \frac{\rho^+}{g_i}$ , where  $g_i = g_i(x(\rho^+, 0))$ . By defining  $\bar{d}_y = d_y - \lambda + \hat{\lambda}$ , we may rewrite Eq.(13) as

$$\begin{pmatrix} \nabla_x L(\hat{x}, \hat{\lambda}) & \nabla \hat{g} \\ \hat{\Lambda} \nabla \hat{g}^t & \hat{G} \end{pmatrix} \begin{pmatrix} d_x \\ \bar{d}_y \end{pmatrix} = \begin{pmatrix} -L(\hat{x}, \lambda) + R \\ \hat{G}(\lambda - \hat{\lambda}) \end{pmatrix}, \quad (14)$$

We now observe that  $L(\hat{x}, \lambda) = \mathcal{O}(\rho^{a+1})$  and  $\hat{G}(\lambda - \hat{\lambda}) = \rho^+ G^{-1}(\hat{g} - g) = \mathcal{O}(\rho^{a+1})$  and  $\|R\| \leq \gamma$  which concludes the proof.  $\blacksquare$

Now consider the effect of an inexact Newton correction.

*Lemma 3.2:* Let  $\hat{x}$  such that  $\|\hat{x} - x(\rho^+, 0)\| \sim \rho^{a+1}$ . Then,  $\Phi(\hat{x} + d_N, \rho^+) \sim \mathcal{O}\left(\frac{(\gamma + \rho^{(a+1)})^2}{\rho^+}\right)$ .

*Proof:* We write

$$\Phi(\hat{x} + d_N, \rho^+) = \Phi(\hat{x}, \rho^+) + \nabla_x \Phi(\hat{x}, \rho^+) d_N + \mathcal{O}\left(\frac{\|d_N\|^2}{(\rho^+)^2}\right), \quad (15)$$

noting the last denominator  $(\rho^+)^2$  which comes from derivating twice  $\frac{\rho^+}{g_i(\hat{x})}$ . The first two terms are bounded by  $\gamma$  and using the bound on  $d_N$  from the lemma 3.1 we get  $\frac{(\gamma + \rho^{(a+1)})^2}{\rho^+}$ .  $\blacksquare$

Consider  $\rho^+ = \rho^b$ . If we want to be as cheap as possible while ensuring super-linear convergence ( $b > 1$ ), we deduce that  $b < \frac{2(1+c)}{3}$  while  $c \leq a$ . Therefore, to get super-linear convergence, one has to pick  $0.5 < c \leq a$ .

2) *Details for quadratic loss:* The proof of lemma 3.1 is very similar to the proof of the Lemma 3 in [2].

Now consider the effect of an inexact Newton correction.

*Lemma 3.3:* Let  $\hat{x}$  such that  $\|\hat{x} - x(\rho^+, 0)\| \sim \rho^{a+1}$ . Then,  $\Phi(\hat{x} + d_N, \rho^+) \sim \mathcal{O}\left(\frac{(\gamma + \rho^{(a+1)})^2}{\rho^+}\right)$ .

*Proof:* We write

$$\Phi(\hat{x} + d_N, \rho^+) = \Phi(\hat{x}, \rho^+) + \nabla_x \Phi(\hat{x}, \rho^+) d_N + \mathcal{O}\left(\frac{\|d_N\|^2}{\rho^+}\right), \quad (16)$$

The first two terms are bounded by  $\gamma$  and using the bound on  $d_N$  from the lemma 3.1 we get  $\frac{(\gamma + \rho^{(a+1)})^2}{\rho^+}$ .  $\blacksquare$

Consider  $\rho^+ = \rho^b$ . If we want to be as cheap as possible while ensuring super-linear convergence ( $b > 1$ ), we deduce that  $b < (1 + c)$  while  $c \leq a$ . Therefore, to get super-linear convergence, one has to pick  $0 < c \leq a$ .

### IV. HIGH ORDER VARIANTS

Instead of solving approximately the predictor and-or corrector steps, we investigate here the effect of using higher order Taylor expressions of the central path. We reformulate slightly the equations to parametrize the path with the scalars  $\rho$  and  $\tau$ . At the current point,

$$\Phi(x, \rho) = \tau \bar{r} \quad (17)$$

for some residual vector  $r = \tau \bar{r}$  with  $\bar{r} = \frac{r}{\|\bar{r}\|}$ . Equation Eq.(17) induces a bi-parameter equation  $x(\rho, \tau)$  and the solution searched for is  $x^* = x(0, 0)$ .

$$\begin{aligned} \hat{x}^1 &= x + \frac{\partial x}{\partial \rho}(\rho^+ - \rho) + \frac{\partial x}{\partial \tau}(-r) \\ \hat{x}^2 &= \frac{1}{2} \left( \frac{\partial^2 x}{\partial \rho^2}(\rho^+ - \rho)^2 + \frac{\partial^2 x}{\partial \rho \partial \tau}(\rho^+ - \rho)(-r) + \frac{\partial^2 x}{\partial \tau^2}(-r)^2 \right) \\ \hat{x}^p &= \sum_{j=0}^p \binom{p}{j} \frac{\partial^p x}{\partial \rho^{p-j} \partial \tau^j} (\bar{\rho}^+ - \bar{\rho})^{p-j} (-\bar{r})^j \end{aligned}$$

We are now concerned with higher order extrapolations  $\sum_{i=1}^a \hat{x}^i$  for  $a > 1$ . Postponing the actual computations of such a  $\hat{x}^a$  for  $a \neq 1$ , we already may obtain the following.

*Lemma 4.1:* Let  $\hat{x}$  such that  $\|\hat{x} - x(\rho^+, 0)\| \sim \rho^{a+1}$  with  $\frac{\rho^{a+1}}{\rho^+} < \infty$ . Then,  $\nabla \phi(\hat{x}, \rho^+) \sim \mathcal{O}\left(\frac{\rho^{a+1}}{\rho^+}\right)$ .

This result allows to claim that by using  $(a > 1)$ -order extrapolations, we get a  $\frac{(a+1)}{2}$  order of convergence without even recourse to Newton corrections, and this both for the log barrier and the quadratic loss variants. Using a first order extrapolation is not enough, and requires a further Newton correction. Indeed, to reach the required approximation criterion,

$\nabla\phi(\hat{x}, \rho^+)$  has to be lower than  $\|r^+\| = \rho^+$ , which implies that  $\rho^{a+1} < \rho^{+2}$ .

Observe in particular that a second order extrapolation ( $\hat{x}^1 + \hat{x}^2$ ) yields a predictor only algorithm achieving the limiting order  $\frac{3}{2}$ . This improves the exterior variant and even more so the interior variant since in this context, both interior and exterior variants share the same improved asymptotic behavior.

#### A. Proof of Lemma 4.1 for the log-barrier

*Proof:* Denote  $x = x(\rho^+, 0)$ ; then  $\Phi(x, \rho^+) = 0$  and write:

$$\begin{aligned}\Phi(\hat{x}, \rho^+) &= \Phi(\hat{x}, \rho^+) - \Phi(x, \rho^+) \\ &= \nabla\hat{f} - \nabla f + \sum \frac{\rho^+}{\hat{g}_i} \nabla\hat{g}_i - \sum \frac{\rho^+}{g_i} \nabla g_i \\ &= \mathcal{O}(\rho^{a+1}) + \sum \frac{\rho^+(g_i - \hat{g}_i)}{\hat{g}_i g_i} \nabla\hat{g}_i \\ &\quad + \sum \frac{\rho^+}{g_i} (\nabla\hat{g}_i - \nabla g_i) \\ &= \mathcal{O}(\rho^{a+1}) + \sum \frac{\mathcal{O}(\rho^{a+1})\rho^+}{\hat{g}_i g_i} \nabla\hat{g}_i \\ &\quad + \sum \lambda_i \mathcal{O}(\rho^{a+1}).\end{aligned}$$

We have  $g_i \sim \Theta(\rho^+)$  and  $\hat{g}_i = g_i + \mathcal{O}(\rho^{a+1})$ , so that  $\hat{g}_i \sim \Theta(\rho^+)$  since  $\frac{\rho^{a+1}}{\rho^+}$  is bounded. ■

#### B. Proof of Lemma 4.1 for the quadratic loss

*Proof:* Denote  $x = x(\rho^+, 0)$ ; then  $\Phi(x, \rho^+) = 0$  and write:

$$\begin{aligned}\Phi(\hat{x}, \rho^+) &= \Phi(\hat{x}, \rho^+) - \Phi(x, \rho^+) \\ &= \nabla\hat{f} - \nabla f + \sum \frac{\hat{g}_i}{\rho^+} \nabla\hat{g}_i - \sum \frac{g_i}{\rho^+} \nabla g_i \\ &= \mathcal{O}(\rho^{a+1}) + \frac{(g_i - \hat{g}_i)}{\rho^+} \nabla\hat{g}_i \\ &\quad + \sum \frac{g_i}{\rho^+} (\nabla\hat{g}_i - \nabla g_i) \\ &= \mathcal{O}(\rho^{a+1}) + \frac{\mathcal{O}(\rho^{a+1})}{\rho^+} \nabla\hat{g}_i \\ &\quad + \sum \lambda_i \mathcal{O}(\rho^{a+1}).\end{aligned}$$

We have  $g_i \sim \Theta(\rho^+)$  and  $\hat{g}_i = g_i + \mathcal{O}(\rho^{a+1})$ , so that  $\hat{g}_i \sim \Theta(\rho^+)$  since  $\frac{\rho^{a+1}}{\rho^+}$  is bounded. ■

### V. COMPUTING EXTRAPOLATIONS

The actual extrapolation computations for the quadratic loss function is presented in details in [3]. To avoid derivatives of the  $\frac{1}{\rho}$  factor involved in the penalty term, we resorted to primal-dual equations in [3]. We develop in this section the details for the high order derivatives for the log barrier.

We rewrite equation Eq.(10) in a simplified notation:

$$\Phi(x, \rho) = c - \rho A^t G^{-1} e. \quad (18)$$

For linear programs,  $c$  and  $A$  are constant while otherwise,  $c = \nabla f(x)$  and  $A = \nabla g(x)$ .  $G = \text{diag}(g_i(x))$ , and for linear programs,  $g(x) = Ax - b$ . We note for the sequel

$$\nabla_x \Phi(x, \rho) = \rho A^t G^{-2} A + \nabla_{xx}^2 l(x, \rho G(x)^{-1} e) \quad (19)$$

$$\nabla_\rho \Phi(x, \rho) = -A^t G^{-1} e \quad (20)$$

and remark that the Lagrangian term vanishes for linear programs.

The implicit function theorem yields

$$\begin{aligned}\nabla_x \Phi(x, \rho) \dot{x}_\rho + \nabla_\rho \Phi(x, \rho) &= 0 \\ \nabla_x \Phi(x, \rho) \dot{x}_\tau - \bar{\mathbf{r}} &= 0.\end{aligned} \quad (21)$$

Thus, the combined extrapolation step reduces to

$$\begin{aligned}\nabla_x \Phi(x, \rho) (\dot{x}_\rho (\rho^+ - \rho)) + \dot{x}_\tau (-\tau) \\ + \nabla_\rho \Phi(x, \rho) (\rho^+ - \rho) + \tau \bar{\mathbf{r}} &= 0,\end{aligned} \quad (22)$$

which, for this first order candidate, simplifies to  $\nabla_x \Phi(x, \rho) \hat{x}^1 + \Phi(x, \rho^+) = 0$ .

In order to go further to the expressions of higher order extrapolates, we first note the following for the log barrier case:

$$\begin{aligned}\nabla_{x\rho}^2 \Phi(x, \rho) = \nabla_{\rho x}^2 \Phi(x, \rho) &= A^t G^{-2} A + \nabla_{xx}^2 l(x, G(x)^{-1} e) \\ \nabla_{\rho\rho}^2 \Phi &\equiv 0 \\ \nabla_{\tau}^2 \Phi = \nabla_{\tau}^2 \Phi &\equiv 0\end{aligned}$$

In a nutshell, any derivative of  $\Phi$  with respect to  $\tau$  vanishes since  $\Phi$  does not involve  $\tau$ , and any high order derivative of  $\Phi$  with respect to  $\rho$  more than once also vanishes since  $\Phi$  is linear in  $\rho$ .

Now, still using the implicit function theorem, this time to equations Eq.(21), we get the following, in which we use  $\Phi$  without arguments as a shorthand notation for  $\Phi(x, \rho)$ :

$$\begin{aligned}\nabla_{xx}^2 \Phi \dot{x}_\rho \dot{x}_\rho + (\nabla_{x\rho}^2 \Phi + \nabla_{\rho x}^2 \Phi) \dot{x}_\rho + \nabla_x \Phi \ddot{x}_{\rho\rho} &= 0 \\ \nabla_{xx}^2 \Phi \dot{x}_\tau \dot{x}_\rho + \nabla_{\rho x}^2 \Phi \dot{x}_\tau + \nabla_x \Phi \ddot{x}_{\tau\rho} &= 0 \\ \nabla_{xx}^2 \Phi \dot{x}_\rho \dot{x}_\tau + \nabla_{x\rho}^2 \Phi \dot{x}_\tau + \nabla_x \Phi \ddot{x}_{\rho\tau} &= 0 \\ \nabla_{xx}^2 \Phi \dot{x}_\tau \dot{x}_\tau + \nabla_x \Phi \ddot{x}_{\tau\tau} &= 0\end{aligned} \quad (23)$$

Observe that the four relations above all imply a linear system defined by the same matrix  $\nabla_x \Phi(x, \rho)$  and the following four right hand sides, conveniently expressed using  $\bar{x}_\tau$  which denotes a constant vector of value  $\dot{x}_\tau$ , and similarly  $\bar{x}_\rho$  is a constant vector of value  $\dot{x}_\rho$ :

$$\nabla_\rho (\nabla_x \Phi \bar{x}_\rho + \nabla_\rho \Phi) = \nabla_{xx}^2 \Phi \dot{x}_\rho \bar{x}_\rho + \nabla_{x\rho}^2 \Phi \bar{x}_\rho + \nabla_{\rho x}^2 \Phi \dot{x}_\rho \quad (24)$$

$$\nabla_\tau (\nabla_x \Phi \bar{x}_\rho + \nabla_\rho \Phi) = \nabla_{xx}^2 \Phi \dot{x}_\tau \bar{x}_\rho + \nabla_{\rho x}^2 \Phi \dot{x}_\tau \quad (25)$$

$$\nabla_\rho (\nabla_x \Phi \bar{x}_\tau - \bar{\mathbf{r}}) = \nabla_{xx}^2 \Phi \dot{x}_\rho \bar{x}_\tau + \nabla_{x\rho}^2 \Phi \bar{x}_\tau \quad (26)$$

$$\nabla_\tau (\nabla_x \Phi \bar{x}_\tau - \bar{\mathbf{r}}) = \nabla_{xx}^2 \Phi \dot{x}_\tau \bar{x}_\tau \quad (27)$$

Hereafter, we use the ‘‘bar’’  $\bar{\rho}$  and  $\bar{\tau}$  to represent actual extrapolation steps values, as opposed to variables within the equations. Now,  $\hat{x}^2 = \ddot{x}_{\rho\rho} (\bar{\rho}^+ - \bar{\rho})^2 + 2\ddot{x}_{\tau\rho} (\bar{\rho}^+ - \bar{\rho}) (-\bar{\tau}) + \ddot{x}_{\tau\tau} (\bar{\tau})^2$  so that the right hand sides involving the second derivatives may be combined into  $(\bar{\rho}^+ - \bar{\rho}) ((\bar{\rho}^+ - \bar{\rho}) Eq.(24) - \bar{\tau} Eq.(26))$



and  $-\bar{\tau}((\bar{\rho}^+ - \bar{\rho})Eq.(25) - \bar{\tau}Eq.(27))$  and, also using the notation that  $\hat{x}^1$  is a constant vector of value  $\hat{x}^1$ , is expressed:

$$\begin{aligned} & \nabla_{\rho} (\nabla_x \Phi(x, \rho) \hat{x}^1 + \nabla_{\rho} \Phi(x, \rho) (\bar{\rho}^+ - \bar{\rho}) - \bar{\mathbf{r}} \bar{\tau}) (\bar{\rho}^+ - \bar{\rho}) \\ & + \nabla_{\tau} (\nabla_x \Phi(x, \rho) \hat{x}^1 + \nabla_{\rho} \Phi(x, \rho) (\bar{\rho}^+ - \bar{\rho}) - \mathbf{r} \bar{\tau}) (-\bar{\tau}) \end{aligned} \quad (28)$$

To establish a recurrence relation to compute the  $\hat{x}^p$ , it is convenient to define a family of functions

$$\Phi^0(x, \rho, \tau) = \Phi(x, \rho) - \tau \bar{\mathbf{r}} \quad (29)$$

$$\Phi^p(x, \rho, \tau) = (\bar{\rho}^+ - \bar{\rho}) \nabla_{\rho} \Phi^{p-1}(x, \rho, \tau) - \bar{\tau} \nabla_{\tau} \Phi^{p-1}(x, \rho, \tau) \quad (30)$$

*Theorem 5.1:*

$$\hat{x}^p = \sum_{j=0}^p \binom{p}{j} \frac{\partial^p x}{\partial \rho^{p-j} \partial \tau^j} (\bar{\rho}^+ - \bar{\rho})^{p-j} (-\bar{\tau})^j$$

satisfies  $\nabla_x \Phi(x, \rho) \hat{x}^p + \Phi^p(x, \rho, \tau) = 0$ .

*Proof:* The inductive proof has its base verified by the relation Eq.(28). The induction step will use the relation:

$$\hat{x}^{p+1} = \frac{\partial \hat{x}^p}{\partial \rho} (\bar{\rho}^+ - \bar{\rho}) + \frac{\partial \hat{x}^p}{\partial \tau} (-\bar{\tau})$$

The equations  $\Phi^p$  includes a term  $\nabla_x \Phi(x, \rho) \hat{x}^p$  defining the linear system, the remaining of  $\Phi^p$  corresponding to the right hand side of the linear equation. ■

The recurrence  $\Phi^p$  may be explicitly written as

$$\nabla_x \Phi(x, \rho) \hat{x}^p + \hat{\Phi}^p$$

where  $\hat{\Phi}^p$  involves terms of the form  $\nabla_{x^j \rho^j \tau^j} \Phi(x, \rho) v_1^{i_1} v_2^{i_2} \dots v_l^{i_l}$  with  $\sum_{k=1}^l i_k = j$ ,  $j_x + j_{\rho} = j$  and  $1 < j \leq p$ . Moreover, each  $v_k$  is composed of partial derivatives of  $x$  with respect to  $\rho$  and/or  $\tau$  up to order  $j_x - 1$ . As it happens, the recurrence may be written using only the  $\hat{x}^p$  without explicit reference to the (mixed) partials derivatives of  $x$  wrt  $\rho$  or  $\tau$ :

$$\Phi^0(x, \rho, \tau) = \Phi(x, \rho) + \tau \bar{\mathbf{r}} \quad (31)$$

$$\Phi^1(x, \rho, \tau) = \nabla_x \Phi(x, \rho) \hat{x}^1 + (\bar{\rho}^+ - \bar{\rho}) \nabla_{\rho} \Phi(x, \rho) + \bar{\mathbf{r}} \bar{\tau} \quad (32)$$

$$\begin{aligned} \Phi^2(x, \rho, \tau) = & \nabla_x \Phi(x, \rho) \hat{x}^2 + 2(\bar{\rho}^+ - \bar{\rho}) \nabla_{x\rho}^2 \Phi(x, \rho) \hat{x}^1 \\ & + \nabla_{xx}^2 \Phi(x, \rho) \hat{x}^1 \hat{x}^1 \end{aligned} \quad (33)$$

$$\begin{aligned} \Phi^3(x, \rho, \tau) = & \nabla_x \Phi(x, \rho) \hat{x}^3 + 3(\bar{\rho}^+ - \bar{\rho}) \nabla_{x\rho}^2 \Phi(x, \rho) \hat{x}^2 \\ & + 3 \nabla_{xx}^2 \Phi(x, \rho) \hat{x}^1 \hat{x}^2 \\ & + 3(\bar{\rho}^+ - \bar{\rho}) \nabla_{xx\rho}^3 \Phi(x, \rho) \hat{x}^1 \hat{x}^1 \\ & + \nabla_{xxx}^3 \Phi(x, \rho) \hat{x}^1 \hat{x}^1 \hat{x}^1 \end{aligned} \quad (34)$$

$$\begin{aligned} \Phi^4(x, \rho, \tau) = & \nabla_x \Phi(x, \rho) \hat{x}^4 + 4(\bar{\rho}^+ - \bar{\rho}) \nabla_{x\rho}^2 \Phi(x, \rho) \hat{x}^3 \\ & + 3 \nabla_{xx}^2 \Phi(x, \rho) \hat{x}^2 \hat{x}^2 \\ & + 4 \nabla_{xx}^2 \Phi(x, \rho) \hat{x}^1 \hat{x}^3 \\ & + 12(\bar{\rho}^+ - \bar{\rho}) \nabla_{xx\rho}^3 \Phi(x, \rho) \hat{x}^1 \hat{x}^2 \\ & + 6 \nabla_{xxx}^3 \Phi(x, \rho) \hat{x}^2 \hat{x}^1 \hat{x}^1 \\ & + 4(\bar{\rho}^+ - \bar{\rho}) \nabla_{xxx\rho}^4 \Phi(x, \rho) \hat{x}^1 \hat{x}^1 \hat{x}^1 \\ & + \nabla_{xxxx}^4 \Phi(x, \rho) (\hat{x}^1)^4 \end{aligned} \quad (35)$$

*1) Implementation for linear programming:* By introducing the notation  $v^1 = A\hat{x}^1$ , and  $V = \text{diag}(v)$ , we may express the high order terms using the following lemma.

*Lemma 5.2:*

$$\nabla_x (v^t V_{\rho} G^{-p} u) = -p A^t V V_{\rho} G^{-(p+1)} u \quad (36)$$

This allows to write equation Eq.(33) as

$$\rho A^t G^{-2} A \hat{x}^2 - 2\rho A^t V^1 G^{-3} v^1 + 2(\bar{\rho}^+ - \bar{\rho}) A^t G^{-2} v^1 = 0 \quad (37)$$

Similarly, equation Eq.(34) leads to the following expression:

$$\begin{aligned} & -6\rho A^t V^1 G^{-3} v^2 - 6(\bar{\rho}^+ - \bar{\rho}) A^t V^1 G^{-3} v^1 \\ & + 3(\bar{\rho}^+ - \bar{\rho}) A^t G^{-2} v^2 + 6\rho A^t (V^1)^2 G^{-4} v^1 \end{aligned} \quad (38)$$

As we may observe, each term involves a single matrix–vector computation in addition to several  $\mathcal{O}(n)$  diagonal matrices and vector operations, overall yielding cheap right hand sides to compute higher order derivatives. This was to be expected.

*2) General implementation using automatic differentiation (AD):* Using AD tools, we may evaluate higher order derivative cheaply too. Assuming full dense Hessian’s—constraint jacobians, the linear system requires  $\mathcal{O}(n^3)$  arithmetic operations to factorizes, and further on, back–front substitutions together with right hand side computations reduce to  $\mathcal{O}(n^2)$  arithmetic operations. As it happens, we may get the high order right hand sides required for the Taylor coefficients in  $\mathcal{O}(n^2)$  complexity, leaving the main burden to obtain and factorize the Jacobian Matrix.

## VI. SHAMANSKII INSPIRED EXTRAPOLATIONS

In the context of unconstrained optimization using Newton’s method, reusing the Hessian matrix is closely related to Shamanskii’s method, sometimes refered to as composite Newton method. Shamanskii’s consists in reusing the Hessian two or more times; this is interesting since factorizing the Hessian has a much superior computational cost than using the factorization to solve a linear system. From an asymptotic point of view, then, high order extrapolations (reminiscent of Chebychev method) or Shamanskii method share the same improvement with respect to the convergence order. The simplicity of Shamanskii’s approach is appealing.

We will provide a Shamanskii approximation to the second order extrapolation. This yields an approximate second order predictor algorithm reaching the limiting convergence order  $\frac{3}{2}$ .

We analyze the following scheme.

$$\begin{aligned} \nabla_x \Phi(x, \rho) \tilde{x}^1 + \Phi(x, \rho^+) & = 0 \\ \nabla_x \Phi(x, \rho) \tilde{x}^2 + \Phi(x + \tilde{x}^1, \rho^+) & = 0 \end{aligned}$$

Thus, we reuse  $\nabla_x \Phi(x, \rho)$  and only change the right hand sides. We recognize that  $\tilde{x}^1 = \hat{x}^1$  previously defined.

*Proposition 6.1:* The second order  $\tilde{x}^2$  is an  $\mathcal{O}(\rho^3)$  approximation to the second order extrapolation  $\hat{x}^1 + \hat{x}^2$ .

*Proof:* We express the right hand side using a Taylor expansion. All the terms in the right hand side are functions evaluated at  $x$  and  $\rho$ , and thus the various  $\Phi(x, \rho)$  will be shorthanded to  $\Phi$ .

$$\begin{aligned} \Phi(x + \tilde{x}^1, \rho^+) &= \Phi \\ &+ \nabla_x \Phi \tilde{x}^1 + \nabla_\rho \Phi (\rho^+ - \rho) \end{aligned} \quad (39)$$

$$+ \nabla_{xx}^2 \Phi \tilde{x}^1 \tilde{x}^1 \quad (40)$$

$$+ 2 \nabla_{x\rho}^2 \Phi \tilde{x}^1 (\rho^+ - \rho) \quad (41)$$

$$+ \nabla_{\rho\rho}^2 \Phi (\rho^+ - \rho)^2 \quad (42)$$

$$+ \mathcal{O}(\max(\tau, \rho)^3)$$

Now, we already have seen that  $\nabla_{\rho\rho}^2 \Phi = 0$ , and by the definition of  $\tilde{x}^1$ , Eq.(39) vanishes, which yields that  $\hat{x}^2$  Eq.(33) and  $\tilde{x}^2$  Eq.(40) and Eq.(41) differ by  $\mathcal{O}(\rho^3)$ . ■

The second order Shamanskii direction is thus a suitable approximation of the second order extrapolation. The conclusion of the lemma 4.1 then still holds.

*Corollary 6.2:*  $\nabla \phi(\tilde{x}^2, \rho^+) \sim \mathcal{O}(\frac{\rho^3}{\rho^+})$

The process may be continued,

$$\nabla_x \Phi(x, \rho) \tilde{x}^3 + \Phi(x + \tilde{x}^1 + \tilde{x}^2, \rho^+) = 0,$$

and in general,

$$\nabla_x \Phi(x, \rho) \tilde{x}^{p+1} + \Phi(x + \sum_{i=1}^p \tilde{x}^i, \rho^+) = 0.$$

We conjecture that the  $\tilde{x}^p$  may be used and preserve the good properties of the  $\sum_{i=1}^p \hat{x}^i$ .

## VII. NUMERICAL ILLUSTRATION

We now provide a simple numerical example to exhibit the benefits of using a Shamanskii like extrapolation.

We consider the simple example

$$\begin{aligned} \min_{x \in \mathbb{R}^6} \quad & f(x) = \sum_{i=1}^6 ix^i \\ \text{s.t.} \quad & g_1(x) = (x_1 + x_3 + x_5)^2 - 1 = 0 \\ & g_2(x) = (x_2 + x_3 + x_4)^2 - 1 = 0 \\ & g_3(x) = x_1 x_6 - 1 = 0 \end{aligned}$$

We use the quadratic penalty function. Therefore, we hope two-steps superlinear convergence almost quadratic using a first order extrapolation and almost  $\frac{3}{2}$  convergence order using a Shamanskii variant. The two steps in the first order variant require factorization and solution of two distinct linear systems while the Shamanskii variant uses a single factorization to solve two related linear systems.

We compare a sub- $\frac{3}{2}$  sequence  $\rho_k$  using both a first order extrapolation and a Shamanskii-2 extrapolation. We may observe in table I that the extrapolation does not require any Newton correction for the last four two-steps extrapolations.

In the tables, the first 7 iterations are identical and thus are omitted. We observe that the first order variant requires a few more iterations to reach our (tight) tolerance.

The first order variant could be improved by considering a sub-quadratic sequence for  $\rho_k$ , and we exhibit the results in table III. The first 13 iterations are identical with those from table II, and we confirm that overall, this variant betters the slower sequence  $\rho_k$ , but the Shamanskii variant is still the most efficient. The limiting order when considering two factorizations is quadratic for the linear extrapolation and  $\frac{9}{4}$  for the Shamanskii version, which is coherent with our example.

As a final remark regarding this simple illustration, the quadratic penalty using Shamanskii strategy benefits much less than log-barrier algorithms. Nevertheless, our example suggests that it (Shamanskii) may improve upon the plain first order extrapolation.

TABLE I  
SECOND ORDER SUB  $\frac{3}{2}$  VARIANT

$\rho$	Iter	$\ \nabla p(x, \rho)\ $	$\ g(x)\ $	$\nabla L$
Ex two 1.0e-02	7	2.2e+00	5.8e-02	
Nwt	8	1.2e-02	5.5e-02	
Nwt	9	6.5e-05	5.5e-02	
Ex two 1.0e-03	10	2.7e-02	5.6e-03	
Nwt	11	1.1e-06	5.6e-03	
Ex two 1.0e-04	12	2.7e-04	5.6e-04	
Nwt	13	1.2e-09	5.6e-04	
Ex two 1.0e-05	14	2.7e-06	5.6e-05	
Ex two 1.0e-06	15	2.8e-08	5.6e-06	
Ex two 1.0e-08	16	9.8e-08	5.6e-08	5.5e-12
Ex two 1.0e-11	17	6.9e-05	5.6e-11	1.2e-15
Ex two	18	1.2e-01	5.6e-14	1.2e-15

## VIII. CONCLUSION AND FUTURE WORK

In this paper, we summarized known results about the asymptotic behavior of SUMT algorithms in non-linear optimization. We considered both interior and exterior penalty variants. Overall, exterior variants enjoy better asymptotic properties.

As can be seen from the table, interior variants suffer from poorer asymptotic convergence order. In particular, for the inexact predictor-corrector strategy, one has to impose  $c > 0.5$  i.e. the residual of the extrapolation and the Newton correction has to be reduced to an order at least 1.5 merely to provide an overall two-steps superlinear behavior. The exterior variant is somewhat more forgiving in this context.

High order predictors make interior and exterior approaches competitive. Actually, the use of higher order predictors allow to bypass any Newton corrector step asymptotically, and for both the interior and exterior variant, allow to reach the same order of convergence limit, namely  $\frac{k+1}{2}$  for order  $k > 1$  extrapolates.

TABLE II  
FIRST ORDER SUB  $\frac{3}{2}$  VARIANT

$\rho$	Iter	$\ \nabla p(x, \rho)\ $	$\ g(x)\ $	$\nabla L$
Ex one 1.0e-02	7	1.3e+01	1.1e-01	
Nwt	8	4.7e-01	5.3e-02	
Nwt	9	4.4e-04	5.5e-02	
Ex one 1.0e-03	10	1.7e+00	5.8e-03	
Nwt	11	7.4e-04	5.6e-03	
Ex one 1.0e-04	12	1.7e-01	5.6e-04	
Nwt	13	7.4e-07	5.6e-04	
Ex one 1.0e-05	14	1.8e-02	5.6e-05	
Nwt	15	7.0e-10	5.6e-05	
Ex one 1.0e-06	16	1.8e-03	5.6e-06	
Nwt	17	1.6e-09	5.6e-06	
Ex one 1.0e-08	18	2.2e-03	5.6e-08	7.2e-12
Ex one 1.0e-11	19	2.2e+00	5.4e-11	1.6e-12
Ex one 1.0e-14	20	2.2e+03	7.4e-12	1.2e-12
Nwt	21	3.1e-02	5.6e-14	1.5e-11
Nwt	22	3.1e-02	5.6e-14	3.4e-15

TABLE III  
FIRST ORDER SUB QUADRATIC VARIANT

$\rho$	Iter	$\ \nabla p(x, \rho)\ $	$\ g(x)\ $	$\nabla L$
Ex one 1.0e-06	14	2.2e-01	5.6e-06	
Nwt	15	1.0e-08	5.6e-06	
Ex one 1.0e-09	16	2.2e-02	5.6e-09	7.3e-12
Ex one 1.0e-14	17	2.2e+03	7.5e-12	1.7e-12
Nwt	18	1.8e-01	5.6e-14	1.7e-11
Nwt	19	3.1e-02	5.6e-14	4.8e-15

It should be recalled that order- $k$  predictor incur a computational cost of  $\mathcal{O}(n^3)$  arithmetic operation to factorize the jacobian matrix plus  $k$  times  $\mathcal{O}(n^2)$  to obtain the high order terms while the corrector steps involve the solution of a linear system, again  $\mathcal{O}(n^3)$ . Therefore, from a complexity per iteration point of view, high order predictors are far preferable to their first order predictor-corrector counterpart: they require only one  $\mathcal{O}(n^3)$  factorization and  $k$   $\mathcal{O}(n^2)$  substitutions while

the first order approach requires two  $\mathcal{O}(n^3)$  factorization and two  $\mathcal{O}(n^2)$  substitutions.

The results presented suggest that the use of the Shamanskii approximation to the higher order trajectory derivatives is a simple solution to reach good asymptotic convergence properties, equivalently good for interior and exterior variants of SUMT. This contrasts with the usage of a simple extrapolation, or approximate computations of the predictor and corrector steps.

The analysis may be combined into a mixed penalty approach to treat programs involving both equality constraints and inequalities. This new strategy outperforms (from an asymptotic analysis point of view) previous studies using mixed interior and exterior penalties. The analysis also may be applied to the exponential penalty as well as other variations.

Future works will involve implementation and comparisons with primal-dual methods. Primal-dual interior point methods have very good convergence properties, but require skill to ensure global convergence while exhibiting good asymptotic behavior.

TABLE IV  
LIMITING CONVERGENCE ORDERS FOR VARIANTS DISCUSSED IN THE PAPER

variant	interior	exterior
corrector	linear	linear
predictor-corrector[1]	2-steps $\frac{4}{3}$	2-steps quadratic
order- $k \geq 2$ pred	$\frac{k+1}{2}$	$\frac{k+1}{2}$
inexact pred-corr	2-steps- $\frac{2}{3}(1+c)$	2-steps $1+c$

REFERENCES

- [1] Jean-Pierre Dussault. Numerical stability and efficiency of penalty algorithms. *S.I.A.M. Journal on Numerical Analysis*, 32(1):296–317, February 1995.
- [2] Jean-Pierre Dussault. Augmented penalty algorithms. *IMA Journal on Numerical Analysis*, 18:355–372, 1998.
- [3] Jean-Pierre Dussault. High order Newton-penalty algorithms. *Journal of computational and applied mathematics*, 182(1):117–133, oct 2005.
- [4] Jean-Pierre Dussault and Abdelatif Elafia. On the convergence rate of the logarithmic barrier algorithm. *Computational Optimization and Applications*, 19(1):31–54, apr 2001.
- [5] Antony V. Fiacco and Garth P. McCormick. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. S.I.A.M., 1990.
- [6] Sanjay Mehrotra. Asymptotic convergence in a generalized predictor-corrector method. *Math. Program.*, 74(1):11–28, 1996.



# Numerical Assessment of Finite Difference Time Domain and Complex-Envelope Alternating-Direction-Implicit Finite-Difference-Time-Domain Methods

Gebriel A. Gannat  
 Engineering Department  
 Sharjah Colleges, HCT, UAE  
 P.O.Box 7947  
 E-mail ggannat@hct.ac.ae

**Abstract**—A thorough numerical assessment of Finite Difference Time Domain and Complex-Envelope Alternating-Direction-Implicit Finite-Difference-Time-Domain Methods has been carried out based on a basic single mode Plane Optical Waveguide structure. Simulation parameters for both methods were varied and the impact on the performance of both numerical methods is investigated.

## I. INTRODUCTION

DIFFERENT numerical modelling techniques have been proposed in literature to analyse optical and photonic devices, such as Beam Propagation method (BPM) [1], Finite-Element-Time-Domain (FETD) method [2] and Finite-Difference-Time-Domain [3] method. The Finite Difference Time Domain (FDTD) is a very popular method mainly because of its ability to simulate ultra-complicated structures in a very simple and straightforward manner. However, as main drawback, FDTD requires high computational resources due to Courant criterion which limits the time step size in order to preserve the numerical stability of the scheme. As an efficient alternative to FDTD, Complex-Envelope Alternating-Direction-Implicit Finite-Difference-Time-Domain (CE-ADI-FDTD) method [4] has been proposed in literature. The main advantage of CE-ADI-FDTD is the time step sizes not bounded by the Courant criterion making virtually possible to employ larger time step sizes with low impact on accuracy and great saving in terms of computational resources. But the approach proposed in [4] has been proven to suffer from numerical instability due to the Absorbing Boundary Conditions (ABCs) [4]-[7]. This instability affects the maximum time step size that is possible to use in order to maintain the scheme stable limiting the numerical efficiency of the method itself. In [5], an improved formulation of the ABCs has been implemented in the context of CE-ADI-FDTD making the numerical scheme stable even for very large time step sizes. The focus in this paper is to assess and investigate the performance of FDTD and CE-ADI-FDTD methods against Courant,

Friederich, Levy Criterion (CFL) and different parameters of Uniaxial Perfectly Matched Layer (UPML). Parameters such as geometric coefficient (g), number and size of PML cells were varied for a number of simulations and the obtained results are presented in this paper.

## II. ASSESSMENT OF FDTD METHOD

A simple waveguide structure shown in Fig. 1 is simulated using the developed FDTD code to investigate the performance of the FDTD method. As shown in this figure, the refractive indices of the core and cladding are 3.2 and 1, respectively. The structure is discretised into a uniform mesh and is terminated by 20 cells UPML to absorb the reflected power. To ensure the single mode propagation at 1.55 $\mu\text{m}$  wavelength, the width (W) of the waveguide chosen to be 0.2 $\mu\text{m}$ . All the results for the reflected power are obtained by injecting a source-field along the transverse x direction, modulated at wavelength 1.55 $\mu\text{m}$  and 5-fs wide Gaussian pulse. The source-fields for TE and TM used is presented by the equations given below;

$$E_{z_{i,j}}^{n+1} = E_{z_{i,j}}^n \phi_j \sin(2\pi f n \Delta t) e^{((n\Delta t - 3T)/T)^2} \quad (1a)$$

$$H_{z_{i,j}}^{n+1} = H_{z_{i,j}}^n \phi_j \sin(2\pi f n \Delta t) e^{((n\Delta t - 3T)/T)^2} \quad (1b)$$

As shown in Fig. 1, the detector point is labelled as D1 to record the incident and the reflected power inside the waveguide. Once the incident and reflected power recorded in D1, the ratio of FFT of the reflected to incident field is calculated to compute the spectrum variation of the reflected power coefficient.

As shown in Fig. 1, the reference point is labelled as D1 to record the incident field transmitted and reflected power from the PML inside the waveguide, respectively. Once the transmitted power reaches the output terminal of the waveguide, the ratio of FFT of the reflected to incident field is calculated to compute the spectrum variation of the reflected power coefficient.

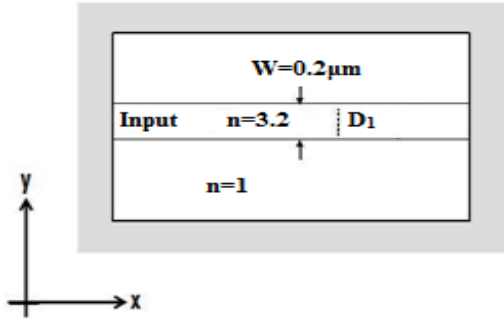


Fig. 1 Basic single mode waveguide, width ( $W$ ) =  $0.2\mu\text{m}$  and refractive index,  $n=3.2$

#### A. Assessment of FDTD against Courant, Friederich, Levy Criterion (CFL)

In order to investigate the effect of CFL criterion on the performance of the FDTD, the waveguide structure, shown in Fig. 1 is simulated using the developed FDTD code. The TE-mode and TM-mode source field given by equations (1a) and (1b) are injected inside the waveguide along the transverse  $x$  direction [8], modulated at wavelength  $1.55\mu\text{m}$  and  $5\text{-fs}$  wide. At the first simulation  $\Delta t$  chosen to be larger than the CFL criterion and the structure is discretised into a uniform mesh with cell size of  $20\text{nm}$ .  $\Delta t$  used in the simulation can be calculated as follows;

$$\Delta t > \frac{1}{c \sqrt{\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2}}} \quad (2)$$

Based on the parameters stated above, field profile shown in Fig. 2 is obtained. As it may be observed from the Figure, the field profile inside the waveguide is unstable and constantly increases as the time step increases. As a result of this stability problem, the propagation of the field profile inside the waveguide can not be simulated. The FDTD code has been tested when  $\Delta t > 2 * (\Delta t)_{\text{CFL}}$ , and  $\Delta t > 3 * (\Delta t)_{\text{CFL}}$  and it has been observed that field profile is extremely unstable when CFL criterion not applied. On other words FDTD is absolutely useless to simulate wave propagation if CFL criterion is not applied.

The simulation parameters modified and  $\Delta t$  chosen to be less than  $(\Delta t)_{\text{CFL}}$  and the obtained result is presented in Fig. 3. As it may be observed that the field profile presented in Fig. 3 represents a Gaussian pulse which is given by the equation (1) and therefore a stable field profile will propagate inside the waveguide and the transmitted and reflected power can be observed along the waveguide.

#### B. Assessment of UPML Reflection.

For further assessment of the FDTD scheme, the reflected power coefficient of the UPML is investigated for both TE-mode and TM mode. In order to investigate

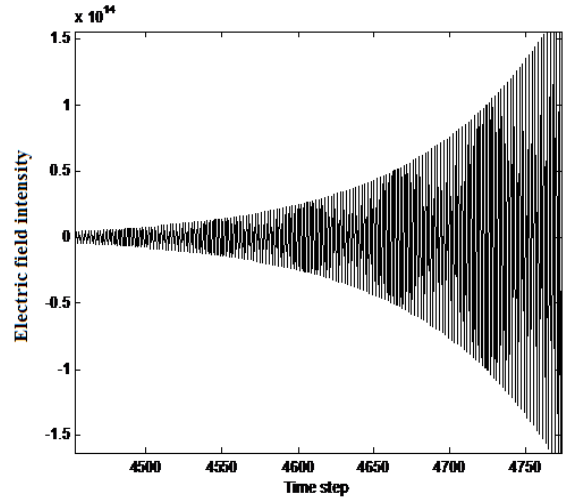


Fig. 2: Time variation of the electric field recorded at point detector D1 inside the waveguide presented in Fig. 1, when

$$\Delta t > (\Delta t)_{\text{CFL}}$$

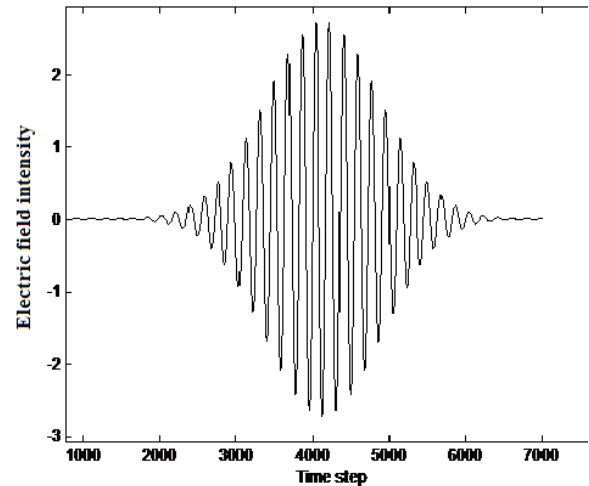


Fig. 3 Time variation of the electric field recorded at point detector D1 inside the waveguide presented in Fig. 1, when

$$\Delta t < (\Delta t)_{\text{CFL}}$$

the effect of the scaling factor ( $g$ ) on the reflected power coefficient, the structure presented in Fig.1 is simulated for three different values of scaling factor ( $g$ ) for both TE and TM mode of propagation. The central wavelength,  $\lambda$  is  $1.55\mu\text{m}$ , the size of discretisation cell ( $\Delta x$  and  $\Delta y$ ) are  $30\text{nm}$ , and the number of UPML cells is 20.

As it may be observed from Fig. 4a, the reflected power coefficient for TE-mode is obtained for a different scaling factor,  $g=1.5$ ,  $2.5$  and  $3.5$ . The reflected power coefficients obtained are about  $-37\text{dB}$ ,  $-25\text{dB}$  and  $-19\text{dB}$ , respectively. From the obtained results, it can be observed that minimum reflected power coefficient is obtained when  $g$  is equal to  $1.5$  and therefore this figure is considered for the rest of the simulations in this research study.

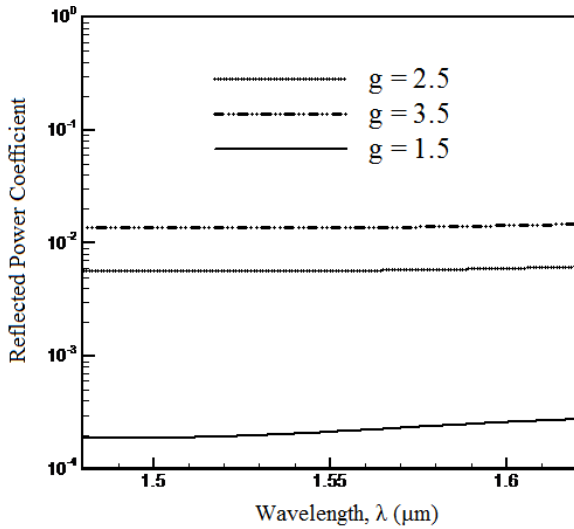


Fig. 4a: Variation of UPML reflected power coefficient for TE-Mode against the wavelength for three different values of “g” at central wavelength 1.55µm.

Similarly, the reflected power coefficients for TM-mode are obtained for the same values of (g) and as presented in Fig. 4b, the values of the reflected power coefficients obtained for TM-mode are about -40dB, -30dB and -27dB respectively. From both figures it can be observed that the reflected power coefficient for both propagation modes ( TE and TM ) is very low and the UPML gives a higher performance when the scaling factor (g) chosen around 1.5 and therefore this figure is considered for the rest of the simulations.

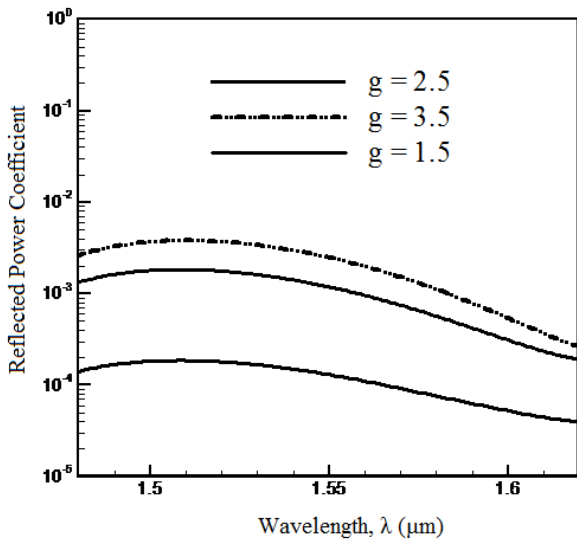


Fig. 4b: Variation of UPML reflected power coefficient for TM-Mode against the wavelength for three different values of “g” at central wavelength 1.55µm.

For further assessments of the reflected power coefficient of UPML for both TE and TM mode of propagation, the structure presented in Fig. 1 is simulated for 10, 20, 30 and 40 UPML cells. The central wavelength, λ is 1.55µm, the size of discretisation cell ( Δx = Δy=30nm), and the scaling factor ( g ) is 1.5. As it may be observed from Fig. 5a, the lowest reflected power

coefficient or TE-mode is obtained when number of UPML cells used is 20, 30 and 40. The reflected power coefficient obtained is about (-38dB). The highest reflected power coefficient is obtained when 10 UPML cells are used and the obtained is about (-20dB).

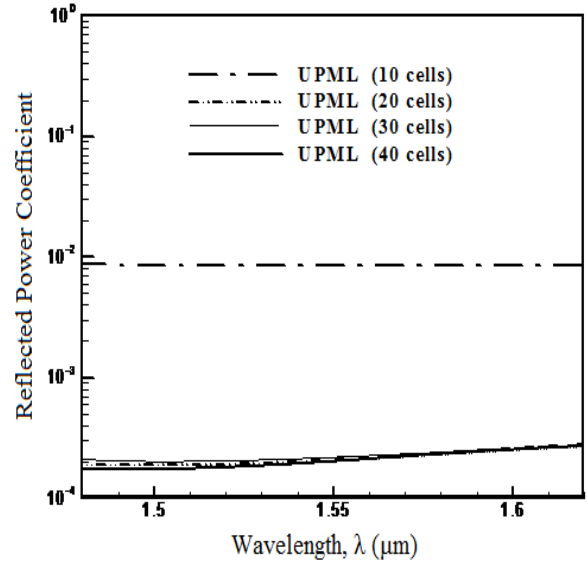


Fig. 5a: Variation of UPML reflected power coefficient of TE-Mode against wavelength, λ when the number of UPML cells is a parameter

When the number of UPML cells increased the UPML layers thickness increases and the ability to absorb incident power on the boundaries increases, this improves the performance of the UPML layer and minimise the reflected power from the UPML boundaries. Similarly, the reflected power coefficients for TM-mode are obtained for the same numbers of UPML cells. As presented in Fig. 5b, the values of the reflected power coefficient are slightly different from TE-mode. When the number of UPML cells used is 20, 30 and 40 reflected power coefficient is slightly below (-40db) and when UPML cells chosen to be 10, the reflected power coefficient obtained about (-35dB). Overall the reflection ratios for both TE and TM propagation mode are very low and this will help the accuracy of the results obtained by the simulations in more complex structures.

For further investigation the structure presented in Fig. 1 is simulated using different sizes of discretisation cell (Δx = Δy) at central wavelength, λ is 1.55µm and the scaling factor, g is 1.5. As it may be observed from Fig. 6a, Δx and Δy values used in the simulations are 10, 20 and 30 nm and the obtained reflected power coefficients are -28dB, -35dB and -39dB respectively. Experimentally Δx and Δy need to < 0.05a. Generally, for the TE-mode the reflected power coefficients for the three different values of Δx and Δy are very low. However it can be concluded that the reflected power coefficients are inverse proportional to the size of the discreti-

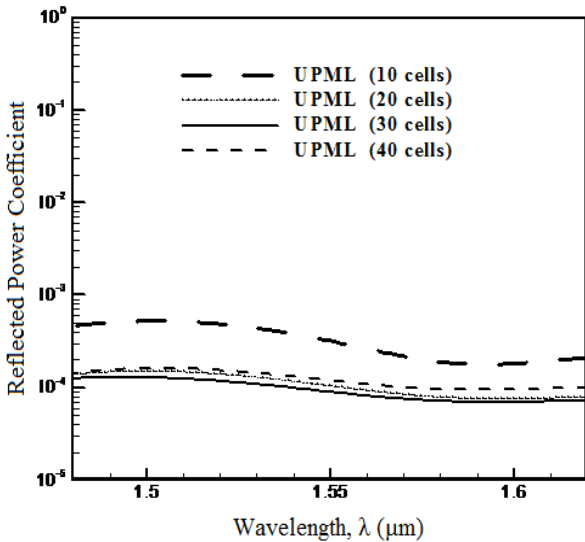


Fig. 5b: Variation of UPML reflected power coefficient for TM-Mode against the wavelength,  $\lambda$  when the number of UPML cells is a parameter.

sation cell, in other words the larger the cell the lowest the reflected power coefficients.

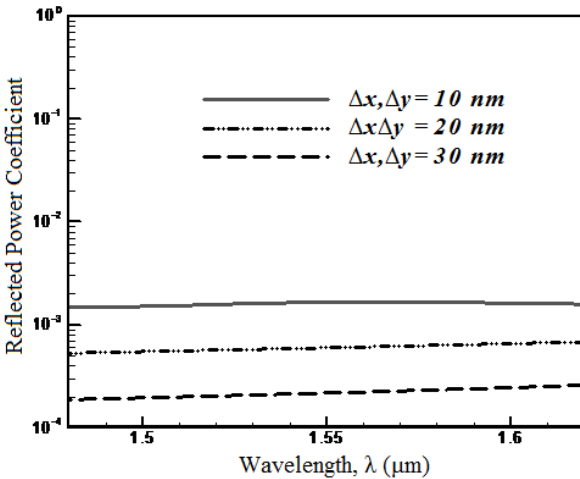


Fig. 6a: Variations of UPML reflected power coefficient of TE-Mode against wavelength,  $\lambda$  when  $\Delta x$  and  $\Delta y$  size is a parameter.

Fig. 6b presents the variation of reflected power coefficient with the wavelength for different values of discretisation cells. As it may be observed from the graphs, when  $\Delta x$  and  $\Delta y$  value is 10 nm, the reflected power coefficient reaches the lowest value (-40dB) at the central wavelength,  $\lambda$  is 1.55 $\mu\text{m}$ , when  $\Delta x$  and  $\Delta y$  value is 30 nm, the reflected power coefficient at the central wavelength is about - reflected power coefficient 40dB, however it reaches the lowest value (-50dB) at the wavelength,  $\lambda$  is 1.55 $\mu\text{m}$ . Meanwhile, when  $\Delta x$  and  $\Delta y$  value is 20 nm, the is about -36dB and similar cross the bandwidth.

### III. ASSESSMENT OF CE-ADI-FDTD METHOD.

The same waveguide structure presented in Fig. 1 is simulated using the developed CE-ADI-FDTD code to

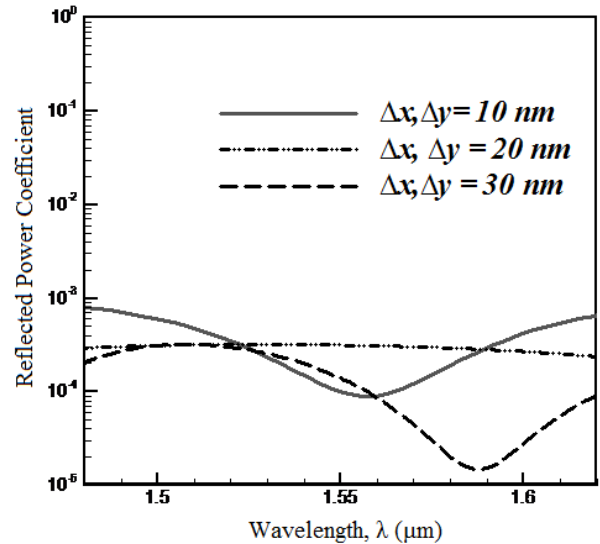


Fig. 6b: Variations of UPML reflected power coefficient of TM-Mode against the wavelength,  $\lambda$  when  $\Delta x$  and  $\Delta y$  size is a parameter.

investigate the performance of the CE-ADI-FDTD method. As shown in this figure, the refractive indices of the core and cladding are 3.2 and 1, respectively. The structure is discretised into a uniform mesh and is terminated by 20 cells PML to absorb the reflected power. To ensure the single mode propagation at 1.55 $\mu\text{m}$  wavelength, the width of the waveguide is chosen to be 0.2 $\mu\text{m}$ .

#### A. Assessment of CE-ADI-FDTD against CFL Criterion.

In order to investigate the effect of CFL criterion on the performance of the CE-ADI-FDTD, the waveguide structure presented in Fig. 1 is simulated using the developed FDTD code. The TE-mode source-field given by equation (1a) injected inside the waveguide along the transverse  $x$  direction, modulated at wavelength 1.55 $\mu\text{m}$  and 5-fs wide. The structure is discretised into a uniform mesh with cell size of 20nm.

For the first simulation,  $\Delta t$  chosen to be ten times larger than the CFL criterion, ( $\Delta t = 10(\Delta t)_{CFL}$ ) and the structure is discretised into a uniform mesh with cell size of 20nm. Based on the parameters stated above the CE-ADI-FDTD tested for ( $\Delta t = 10(\Delta t)_{CFL}$ ) and ( $\Delta t = 30(\Delta t)_{CFL}$ ) and the field profile shown in Fig. 7 is obtained. As it may be observed from the Figure, the field profile inside the waveguide is very stable and the CFL criterion is completely eliminated. Furthermore the same structure simulated with  $\Delta t$  chosen to be less than the CFL criterion, ( $\Delta t = 0.9(\Delta t)_{CFL}$ ) and the obtained field profile shown in Fig. 8 is very stable.



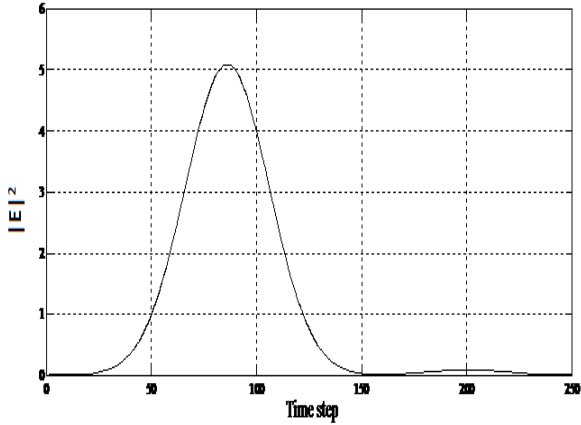


Fig. 7: Time variation of the envelopes of the electric field recorded at point detector D1 inside the waveguide presented in Fig. 1, when  $\Delta t > (\Delta t)_{CFL}$

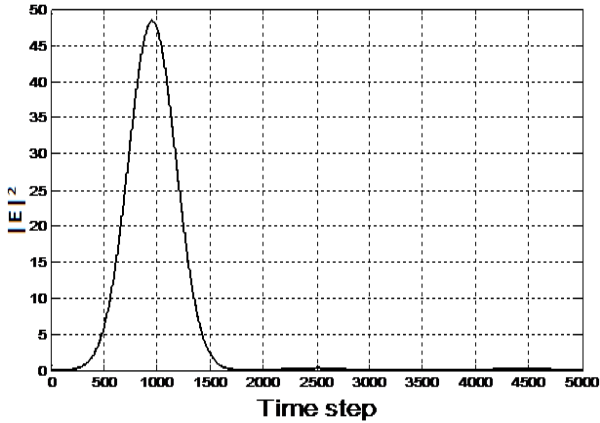


Fig. 8: Time variation of the envelopes of the electric field recorded at point detector D1 inside the waveguide presented in Fig. 1, when  $\Delta t < (\Delta t)_{CFL}$

*B. Assessment of PML Reflection*

For further assessment of the CE-ADI-FDTD scheme, the reflected power coefficient of the PML is investigated for TE-mode. In order to investigate the effect of the scaling factor ( $g$ ) on the reflected power coefficient, the structure presented in Fig. 1 is simulated for three different values of scaling factor ( $g$ ). The central wavelength,  $\lambda$  is  $1.55\mu\text{m}$ , the size of discretisation cell,  $\Delta x$  and  $\Delta y$  is  $30\text{nm}$ , and the number of PML cells is  $20$ . As it may be observed from Fig. 9, that the lowest reflected power coefficient obtained when the scaling factor ( $g$ ) is equal to  $1.5$ , it is about  $(-50\text{dB})$ .

In order to investigate the effect of time step ( $\Delta t$ ) on the reflected power coefficient, the structure presented in Fig. 1 simulated for three different values of  $\Delta t$ . The central wavelength,  $\lambda$  is  $1.55\mu\text{m}$ , the size of discretisation cell,  $\Delta x$  and  $\Delta y$  is  $30\text{nm}$ , and the number of PML cells is  $=20$ . It may be observed from Fig. 10 the reflected power coefficients obtained for three different values of  $\Delta t$  are about the same and therefore it may be

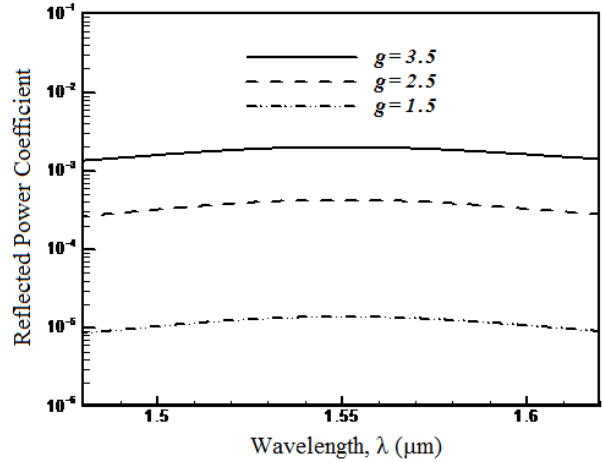


Fig. 9: Variations of PML reflected power coefficient of TE-Mode against the wavelength for three different values of “ $g$ ” at central wavelength  $1.55\mu\text{m}$ .

concluded that using a large  $\Delta t$  value it will have no major impact on the accuracy of the CE-ADI-FDTD scheme.

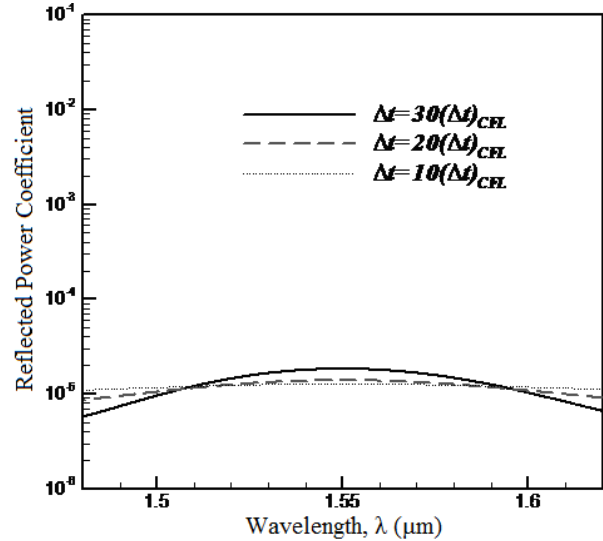


Fig. 10: Variations of PML reflected power coefficients with the wavelength for different values of  $\Delta t$  at central wavelength  $1.55\mu\text{m}$ .

Furthermore, the same structure presented in Fig. 1 is simulated to investigate the effect of the size of the discretisation cells on the reflected power coefficient. Discretisation cells in  $x$  and  $y$  direction are chosen to be equal and the structure simulated three times using cell size  $10, 20$  and  $30\text{nm}$ . As it may be observed from Fig. 11, the lowest value of reflected power coefficient obtained is about  $-55\text{dB}$  when the cell size is  $20\text{nm}$ .

IV. CONCLUSION

In this paper a number of simulations have been carried out in order to investigate the performance of the FDTD and CE-ADI-FDTD for different simulation parameters such as, Courant, Friederich, Levy Criterion, scaling factor ( $g$ ), size and number of discretisation cells. From the simulation results presented can be observed that the applying of Courant, Friederich, Levy

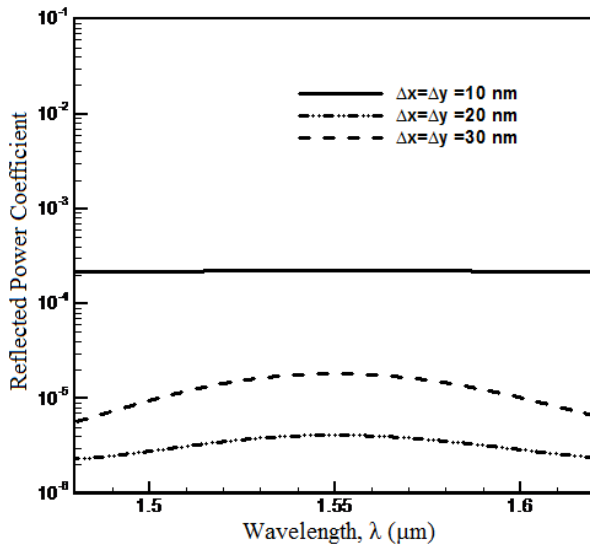


Fig. 11: Variations of PML reflected power coefficients with the wavelength for different values of  $\Delta x$ ,  $\Delta y$  at central wavelength  $1.55\mu\text{m}$

Criterion has a big impact on the stability of the simulation in FDTD method but has no impact on CE-ADI-FDTD in terms of stability. The scaling factor ( $g$ ), cell size and thickness of PML layer have an impact on the reflected power coefficient for both FDTD and CE-ADI-FDTD and have to be selected carefully.

## REFERENCES

- [1] Patrocínio da Silva J., Hernández-Figueroa H. E., and Ferreira Frasson A. M., "Improved Vectorial Finite-Element BPM Analysis for Transverse Anisotropic Media," *IEEE J. Lightwave Technol.*, 2003, 21, (2), pp. 567-576.
- [2] S. S. A. Obayya, B. M. Rahman, and H. A. El-Mikati, "New Full Numerical Efficient Propagation Algorithm Based on the Finite Element Method," *IEEE J. Quantum Electron.*, vol. 18, no. 3, pp. 409-415, March 2000.
- [3] M. De Pourcq, and C. Eng, "Field and Power-Density Calculations in Closed Microwave Systems by Three-Dimensional Finite Difference", *IEE Proc. H Microwaves, Antennas & Propagation*, Vol. 132, pp. 360-368, Oct. 1985.
- [4] Rao H., Scarmozzino R., and Osgood R. M.: 'An Improved ADI-FDTD Method and Its Application to Photonic Simulations', *IEEE Photon. Technol. Lett.*, 2002, 14, (4), pp. 477-479.
- [5] Pinto D., and Obayya S. S. A.: 'Improved Complex-Envelope Alternating-Direction-Implicit Finite-Difference-Time-Domain Method for Photonic-Bandgap Cavities', *IEEE J. Lightwave Technol.*, 2007, 25, (1), pp. 440-447.
- [6] T. Namiki, and K. Ito, "Investigation of Numerical Errors of The Two-Dimensional ADI-FDTD Method," *IEEE Trans. Microw. Theory Tech.*, Vol. 48, no. 11, pp. 1950 – 1956, Nov. 2000.
- [7] Sai-tak Chu and Sujeet K. Chaudhuri "A Finite-Difference Time-Domain Method for the Design and Analysis of Guided-Wave Optical Structure," *J. Lightwave Technol.*, vol. 7, no.12, pp. 2033-2038, Dec. 1989.
- [8] A. Taflove, *Computational Electrodynamics: The finite Difference Time Domain Method*. Boston, MA: Artech, 1995

# Performing Conjoint Analysis within a Logic-based Framework

Adrian Giurca, Ingo Schmitt

Daniel Baier

Dept. of Databases and Information Technology  
 Brandenburg University of Technology  
 P.O. 101344, 03013 Cottbus, Germany  
 Email: {giurca, schmitt}@tu-cottbus.de

Dept. of Marketing and Innovation Management  
 Brandenburg University of Technology  
 P.O. 101344, 03013 Cottbus, Germany  
 Email: daniel.baier@tu-cottbus.de

**Abstract**—Conjoint Analysis is heavily used in many different areas: from mathematical psychology, economics and marketing to sociology, transportation and medicine trying to understand how individuals evaluate products/services and as well as on predicting behavioral outcomes by using statistical methods and techniques. Nowadays is not much agreement about best practice, which in turn has led to many flavors of CA being proposed and applied. The goal of this paper is to offer a solution to perform Adaptive Conjoint Analysis inside CQQL, a quantum logic based information framework. We describe an algorithm to compute a logical CQQL formula capturing user preferences and use this formula to derive decision rules.

## I. INTRODUCTION AND MOTIVATION

NOWADAYS, the term *Conjoint Analysis* (CA) is used in many different ways. While in the past it was mostly on the interest of marketers and psychologists, today's variants are used in many fields including applied economics, sociology, transportation and medicine. The origins of Conjoint Analysis come from developments in several disciplines, most notably economics ([27], [26]) and mathematical psychology ([30], [31], [2]).

Much of the Conjoint Analysis work was used trying to understand how individuals evaluate products/services and form preferences (see, [18], [22], [33] and possibly others). In the last thirty years the CA literature focused more on predicting behavioral outcomes by using statistical methods and techniques ([3]) and this resulted in a widespread variation in CA practice. Recently, applications in innovation market were developed ([4]). The result today is that there is not much agreement about best practice, which in turn has led to many flavors of CA being proposed and applied.

The regular case involves a compositional model (see [17], [20], [16]) where the respondents provide the necessary information to be able to compute *local utilities*. Various ACA interpretations define a *conjoint*(or *utility*) function  $U$  that aggregates the local utilities to an *overall utility* i.e., if  $o$  is

This work is supported by German Federal Ministry of Education and Research, ForMaT project (Forschung für den Markt im Team), Phase II, Innovationslabor: Multimediale Ähnlichkeitssuche zum Matchen, Typologisieren und Segmentieren, <http://www.unternehmen-region.de/de/4818.php> and by DFG Project SQ-System: Entwicklung von Konzepten für ein quantenlogikbasiertes Retrieval-Datenbank-Anfragesystem: Anfragesprache, interaktive Suchformulierung sowie effiziente Anfragesauswertung

a product representation then  $U(o)$  denotes the overall utility of this product. The main condition for such a function is to be monotonic with respect of preferences i.e.,  $U(o_1) \geq U(o_2)$  whenever  $o_1 \succeq o_2$  (see [17], [8], [9]). There is a large literature concerning the way the overall score of a product should be computed (see [3], [13], [17], [19], [16], [20], and many other). A well known model for such a utility function is the additive linear model (see [3] for a survey of models). Basically, the overall utility is an additive linear combination on local utilities adjusted with attribute weights and compensated with a constant depending on interview:

$$U_p(o_j) = \mu + \sum_{k=1}^n \sum_{l=1}^{n_k} \beta_{kl} \cdot x_{jkl} + e_j$$

where  $\mu$  is the mean preference value across all profiles,  $U_p(o_j)$  – is the total score on product profile  $o_j$  with respect to respondent  $p$ ,  $\beta_{kl}$  – is the weight of value  $a_{kl}$  of attribute  $A_k$ ,  $x_{jkl} = \begin{cases} 1, & \text{if } o_j.A_k = a_{kl} \\ 0, & \text{otherwise} \end{cases}$  and  $e_j$  is the measurement error.

Therefore the problem reduces to find all  $\beta_{kl}$  and  $\mu$ . We need estimates and not crisp values because we cannot offer all possible product profiles in a user interview simply because they can be too many([22]). Traditional conjoint uses full profiles (complete product descriptions) as basis for user surveys, but such an approach suffers by description complexity in the presence of many attributes ([22], [23], [24] and [25]), the adaptive conjoint introduces a "partial evaluation" i.e., requesting respondents to evaluate only partial profiles, named *stimuli* by using trade-off matrix approaches, with emphasis on pair comparisons.

A solution of the above model will entitle experts to compute a measure of acceptance of a product by the consumer. However, such a solution is not able to provide explanations on product features and how they may influence the consumer final decision. Logical languages offers qualitative and symbolic methods to complement these standard approaches of economic decision theory.

The goal of this paper is to offer a heuristic to perform Adaptive Conjoint Analysis inside a logic based framework. We use the Commuting Quantum Logic Language,

TABLE I  
PRODUCT ATTRIBUTES AND USER RATINGS

AttrId	Attr Weight	Attr. Name	Attr. Value	ValueId	Value Weight
1	6	Operating system	Android	1	7
			Windows Phone	2	6
			other(proprietary)	3	3
2	7	WiFi	yes	1	6
			no	2	3
3	7	Screen size	less than 3.5"	1	3
			3.5"-4.0"	2	7
			greater than 4.0"	3	8
4	5	Battery life	12 hours	1	9
			6 hours	2	6
			4 hours	3	5
			2 hours	4	1
5	8	PriceLevel	less than 150 EUR	1	9
			150 EUR - 250 EUR	2	8
			250 EUR - 500 EUR	3	5
			greater than 500 EUR	4	0

CQQL([35]), to learn a logical formula  $U$  such that the CQQL evaluation of this formula against a specific set of database objects (product profiles) should monotonically correlate with the user preferences on objects i.e. if  $o_1$  and  $o_2$  are two product profiles  $eval(U, o_1) \geq eval(U, o_2) \Leftrightarrow o_1 \succeq o_2$ . Next section shows an illustrating example of our approach.

A. A simple scenario: Which smartphone they may prefer?

*Example 1 (Which smartphone do you prefer?):* A smartphone manufacturer aims understanding which smartphone is preferred by a specific target group. It performs adaptive conjoint analysis with CQQL using a product decomposition and user scores to compute a logical formula fulfilling the user preferences. This formula can be easily interpreted towards obtaining design decisions.

The initial data consists of a product decomposition into a set attribute values (e.g., "Operating System" is a product attribute with three possible values: "Android", "Windows Phone" and "other(proprietary)") and an initial user rating of attributes and values (e.g., attribute "WiFi" has score 7 and value "Android" has score 7) as shown in the Table I.

Product decomposition is an important stage in conjoint analysis. Many papers emphasize that the decomposition should be chosen carefully to satisfy a property known as *preferential independence*. Preferential independence is extremely important because if each set of attributes is preferentially independent of its complement set, then the attribute utility can be represented by an additive or multiplicative decomposition ([26]). If the attributes are not independent, according to [14], [15], and [5], the utility is more difficult to be estimated. It is not the goal of this paper to analyze methodologies and techniques for product decomposition, therefore we assume an initial a set of attributes together with their possible values.

The last step of our ACA-CQQL learning process yields the following DNF formula<sup>1</sup>:

$$U = (A_1 \wedge \overline{A_2} \wedge A_3 \wedge \overline{A_4} \wedge A_5) \vee (\overline{A_1} \wedge A_2 \wedge A_3 \wedge \overline{A_4} \wedge \overline{A_5})$$

<sup>1</sup>for sake of simplicity, in this example,  $U$  covers the two most important minterms.

After logical transformations such as distributivity, double negation, transforming to implication and dual implication the following set of decision rules are produced:

$$\begin{aligned} R_1 : \overline{A_2} \vee A_5 \rightarrow A_1; & R_2 : A_2 \rightarrow \overline{A_1}; & R_3 : A_1 \vee \overline{A_2} \rightarrow A_5; \\ R_4 : A_2 \rightarrow \overline{A_5}; & & \\ R_5 : \overline{A_2} \vee A_5 \rightarrow A_1; & R_6 : A_2 \rightarrow \overline{A_1} \wedge \overline{A_5}; & R_7 : A_5 \rightarrow A_1; \\ R_8 : \overline{A_2} \rightarrow A_1 \wedge A_5; & & \\ R_9 : A_3, & R_{10} : \overline{A_4} \end{aligned}$$

These rules are interpreted according with the user's level of importance of attribute values, e.g.,  $R_9$  and  $R_{10}$  requests that your product should definitely have a high level of  $A_3$  (i.e. "large screen") and may have a low level of  $A_4$  ("2 hour").  $R_1$  means "whenever you have a low level of  $A_2$  and a high level of  $A_5$ , you should have a high level of  $A_1$  too" that is "if the phone has no WIFI and a low price then it should have OS Android".

In addition, each minterm of the the learned formula  $U$  defines a logical interpretation satisfying  $U$ . Each such interpretation recommends product profiles. For example,  $\mathcal{I} = \{\overline{A_1}, A_2, A_3, \overline{A_4}, \overline{A_5}\}$  (introduced by the second minterm in our example) recommends

("proprietary OS", "WIFI", "> 4.0\"", "2 hours", "high price") While these rules are suitable to be processed by rule engines, we can also derive logically equivalent representations tailored to human interpretation:

$$\begin{aligned} F_1 : A_3 \wedge \overline{A_4}; \\ F_2 : A_1 \oplus \overline{A_2}; \\ F_3 : A_1 \leftrightarrow A_5; \\ F_4 : \overline{A_2} \oplus A_5; \end{aligned}$$

## II. OVERVIEW OF OUR APPROACH

Let  $\mathcal{A} = \{A_1, \dots, A_n\}$  be a set of mutually independent attributes. Let  $dom(A_i)$  the value space of attribute  $A_i$ . Let  $\mathcal{O} = \{(a_1, \dots, a_n) | a_k \in dom(A_k), k = 1, \dots, n\}$  a set of possible objects built over  $\mathcal{A}$ . Let  $\mathcal{S}$  be a finite set of *incomplete objects* or *stimuli* i.e.  $s \in \mathcal{S}$  is a database tuple with nulls. The *null* interpretation related to these objects is in the sense of missing information, e.g., considering 6 attributes a

possible stimuli is the tuple  $s = (a_1, null, null, a_4, null, null)$  where  $a_i$  are specific attribute values.  $\mathcal{S}$  is built by extracting incomplete objects from  $\mathcal{O}$ . Let  $\mathcal{P}$  be a set of respondents. The *ACA-CQQL conjoint representation* is the weighted full DNF<sup>2</sup>,  $U = \bigvee_{w_k} m_k$  where  $m_k$  denotes the  $k$ -minterm and  $w_k \in [0, 1]$  is the weight of minterm  $m_k$ . Each minterm is a conjunction of exactly  $n$  literals (positive attribute occurrence or negative attribute occurrence) each corresponding to one of the  $n$  attributes of ACA-CQQL problem.

The overall approach we use is briefly described below:

- 1) For each respondent  $p \in \mathcal{P}$ , use an interview to derive a preference relation  $P_{\mathcal{S},p} : \mathcal{S} \times \mathcal{S} \rightarrow \{0, 1, unknown\}$ ;
- 2) Use a stimuli preference relation  $P_{\mathcal{S},p}$  to compute a rank  $\rho_{\mathcal{S},p}$  on  $\mathcal{S}$ . A stimuli  $s$  is better ranked by  $\rho_{\mathcal{S},p}$  when the user likes  $s$  better than other one (with a lower rank);
- 3) Derive an induced rank on  $\mathcal{O}$ ,  $\rho_{\mathcal{O},p}$ ; This step computes a rank on full objects by considering the computed rank in Step 2.
- 4) Use CQQL evaluation to obtain  $P_{\mathcal{M},p}$  a preference relation on minterms of  $U$ ; This step allows to express a respondent preference relation on minterms and by consequence to compute an overall preference as described in Step 5.
- 5) Compute an overall minterm preference  $P_{\mathcal{M}}$  by aggregation of all  $P_{\mathcal{M},p}$ ;
- 6) Use  $P_{\mathcal{M}}$  to compute,  $\rho_{\mathcal{M}}$ , a rank on CQQL formula minterms; This is one of the core steps of our solution. Ranking minterms of the full DNF allows to select significant minterms and as such to derive an approximation of interests.
- 7) Create and interpret decision rules considering the best ranked minterms.

Methods for learning and predicting preferences are addressed by the machine learning community [21] and recommender systems [1]. These communities provides solutions such as approximating the scoring function by using interviews (*preference elicitation*) to *collaborative filtering*, where the user preferences are estimated from the preferences of other users. The steps 1–3 are described in Section III while steps 4–7 are described in Section V.

### III. ORDERING FROM PREFERENCES ON INCOMPLETE INFORMATION

Conjoint Analysis associates each respondent  $p \in \mathcal{P}$ , with a set of *pairwise comparisons* (2–stimuli questions), each comparison being rated on a Likert scale [28]. Each question may have its own scale. Likert scales are bipolar scaling methods therefore we can straightforward derive a preference from ratings.

Let  $p \in \mathcal{P}$  be a respondent. Let  $(s_i, s_j)$  be a comparison rated

<sup>2</sup>Any logical formula can be converted to a disjunctive normal form (DNF) by using logical equivalences, such as the double complement elimination, De Morgan’s laws, and the distributive law. Recall that CQQL provides rewriting rules to translate from weighted operations to into CQQL Boolean calculus. Therefore we are going to learn a formula directly in the DNF form.

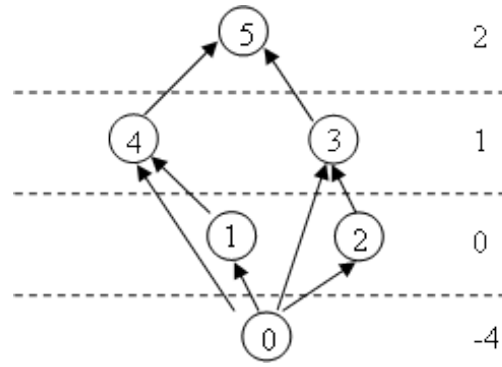


Fig. 1. Graph of preferences

with  $r$  on a Likert scale  $1..K_p$ ,  $K_p \in \mathbb{N}$ . One traditional scale is 1..9 but other choices can be used too (e.g. 1..5). On scale 1..9,  $r = 1$  means "I like very much  $s_i$ ",  $r = 5$  is neutral ("I equally like both  $s_i$  and  $s_j$ ") while  $r = 9$  means "I like very much  $s_j$ ". Then

$$P(s_i, s_j) = \begin{cases} 0, & r < \lfloor K_p/2 \rfloor + 1, (s_i \succ s_j) \\ unknown, & r = \lfloor K_p/2 \rfloor + 1 \\ 1, & r < \lfloor K_p/2 \rfloor + 1, (s_j \succ s_i) \end{cases}$$

The first step is to use the given initial preference  $P_{\mathcal{S}}$  to learn an ordering function  $\rho_{\mathcal{S}}$ . We use the Algorithm 1, a greedy ordering introduced by [6].

This algorithm is based on viewing preferences as a directed

**Data:**  $\mathcal{S}$ , binary preference  $P_{\mathcal{S}}$

**Result:**  $\rho_{\mathcal{S}}$

Let  $V = \mathcal{S}$ ;

**foreach**  $v \in V$  **do**

$\pi(v) = \sum_{u \in V} P_{\mathcal{S}}(v, u) - \sum_{u \in V} P_{\mathcal{S}}(u, v)$ ;

**end**

**while**  $V \neq \emptyset$  **do**

    Let  $t = \text{argmax}_{u \in V} \pi(u)$ ;

    Let  $\rho_{\mathcal{S}}(t) = |V|$ ;

$V = V - \{t\}$ ;

**foreach**  $v \in V$  **do**

$\pi(v) = \pi(v) + P_{\mathcal{S}}(t, v) - P_{\mathcal{S}}(v, t)$ ;

**end**

**end**

**Algorithm 1:** *orderByPrefs*( $\mathcal{S}, P$ ), [6]

weighted graph where the initial set of vertices  $V$  is equal of set of stimuli  $\mathcal{S}$  and each edge  $u \rightarrow v$  has weight  $P_{\mathcal{S}}(u, v) = 1$  ( $v$  is preferred to  $u$ ). Each vertex  $v$  gets a "potential"  $\pi(v)$  which is the sum of the incoming edges minus the sum of outgoing edges. The Algorithm 1 picks some node  $t$  that has a maximum potential, assigns it a rank  $\rho_{\mathcal{S}}(t) = |V|$  and then ordering in the same way the remaining nodes, after updating the nodes potentials.

The potentials define a stratification of the directed weighted graph into node sets of the same potential. The stratum

with the highest potential is processed first. However when processing the nodes in the same stratum all of them have the same potential, therefore the algorithm must choose between them. For example, considering the preferences as in Figure 1 (the initial potential of each stratum is shown too) it is easy to see that node 5 will be processed first (it gets a rank of 4) but then we have a nondeterministic selection because either 4 or 3 can be a choice (both will have the same updated potential). Therefore the output ordering provided by Algorithm 1 depends on the *argmax* implementation<sup>3</sup>. Among others, orderings  $\{0, 1, 2, 3, 4, 5\}$  and  $\{0, 2, 1, 4, 3, 5\}$  can be produced.

*Definition 1 (Incomplete Objects Equality and Membership):*

Two incomplete objects  $s_1, s_2 \in \mathcal{S}$  are *equal*, and we write  $s_1 = s_2$ , when all their correspondent values are the same i.e. if  $s_1 = (a_1, \dots, a_n)$  and  $s_2 = (b_1, \dots, b_n)$  then  $s_1 = s_2$  iff  $a_k = b_k$  for all  $k = 1, \dots, n$ . The values equality interpretation is defined as in the below table:

=	<i>v</i>	<i>null</i>
<i>v</i>	true	false
<i>null</i>	false	true

Let  $s = (a'_1, \dots, a'_n) \in \mathcal{S}$  and  $o = (a_1, \dots, a_n) \in \mathcal{O}$ . We say that  $s$  is *contained by*  $o$  (or  $o$  *contains*  $s$ ) and we denote  $s \subseteq o$  iff  $a'_k \neq null$  implies  $a'_k = a_k$  for all  $k = 1, \dots, n$ . Therefore an incomplete object is contained by an object when its non-null values are the same in the related object.

The following definition introduces an extension from an order on incomplete objects to an order on complete objects.

*Definition 2 (Ordering Extension):* Let  $\mathcal{O}$  be a set of complete objects,  $\mathcal{S}$  be a set of incomplete objects and  $P_{\mathcal{S}}$  a binary preference. Let  $\rho_{\mathcal{S}}$  obtained by Algorithm 1. Then, the ordering  $\rho_{\mathcal{O}}$  defines the *extension of*  $\rho_{\mathcal{S}}$  from  $\mathcal{S}$  to  $\mathcal{O}$ :

$$\rho_{\mathcal{O}}(o) = \sum_{s_i \in \mathcal{S}, s_i \subseteq o} u_{s_i} \rho_{\mathcal{S}}(s_i)$$

where  $u_{s_i}$  is the stimuli utility. If no stimuli belong to object  $o$  then  $\rho_{\mathcal{O}}(o) = 0$ .

Stimuli utility is an additive linear combination of attribute value weight i.e. if  $m_s$  is the number of distinct attribute values of  $s \in \mathcal{S}$ , then  $u_s = \frac{1}{m_s} \sum_{i=1}^N \theta_i \cdot (\sum_{k=1}^{n_i} \theta_{ik} \cdot x_{ik})$  where  $x_{ik} = 1$  when  $A_i$  is present (with value  $a_{ik}$ ) in stimuli  $s$ , otherwise is 0 (null values are ignored).  $\theta_i \in [0, 1]$  is the weight of attribute  $A_i$  and  $\theta_{ik} \in [0, 1]$  is the weight of value  $a_{ik} \in \text{dom}(A_i)$ . The reader may notice that  $\theta_i$  and  $\theta_{ik}$  are obtained from the respondent initial ratings of attributes and attribute values as requested by traditional ACA typically by using a  $[0, 1]$  mapping from ratings on Likert scales.

*Proposition 1:* Let  $o_1, o_2 \in \mathcal{O}$  such that  $s_1, \dots, s_k \subseteq o_1$  and  $s_1, \dots, s_k, s_{k+1} \subseteq o_2$ . Then  $\rho_{\mathcal{O}}(o_2) > \rho_{\mathcal{O}}(o_1)$ .

*Proof:* Easy to see that  $\rho_{\mathcal{O}}(o_2) = \rho_{\mathcal{O}}(o_1) + u_{s_{k+1}} \rho_{\mathcal{S}}(s_{k+1})$ . ■

<sup>3</sup>While Algorithm 1 successfully applies when we have a binary preference as input (see [7] for an optimality proof), in practical cases we work with non binary preferences i.e., there are pairwise comparisons  $(x, y)$  which are rated neutral, therefore  $P_{\mathcal{S}}(x, y) = \text{unknown}$ . In this work we will ignore neutral rated questions.

Any ordering induce a preference as below:

$$P_{\rho}(x, y) = \begin{cases} 1, & \rho(x) < \rho(y) \\ 0, & \rho(x) > \rho(y) \\ \text{unknown}, & \rho(x) = \rho(y) \end{cases}$$

When  $\rho$  is strict then,  $P_{\rho}$  is binary (the *unknown* case does not occur).

In the rest of the paper we assume as given  $\rho_{\mathcal{O}}$  and  $P_{\mathcal{O}}$ . The next step is to use CQQL minterm evaluation to obtain a preference on formula minterms. The goal of the next section is to give basic information about CQQL language, particularly on CQQL evaluation rules.

#### IV. COMMUTING QUANTUM QUERY LANGUAGE (CQQL)

Introduced in [35], CQQL is an extension of the relational calculus using quantum logic paradigm which defines *metric* (or *similarity*) predicates, weighted conjunction ( $\wedge_{\theta_1, \theta_2}$ ), weighted disjunction ( $\vee_{\theta_1, \theta_2}$ ) and quantum negation. CQQL extends relational calculus by allowing for complex logical formulas mixing classical first-order logic predicates with metric predicates. Score values from  $[0, 1]$  results from the evaluation of metric predicates on data objects. On the other hand, traditional database predicates (*non-metric* predicates) provide 1 for true and 0 for false. A significant advantage of the language derives from the capabilities of quantum measurement results to be interpreted as probability values. Therefore conjunction, disjunction and negation conforms with the probability calculus. As a consequence, many concepts of information retrieval, already embedded into linear algebra and probability theory, can be addressed.

Processing *non-metric attributes* i.e. attributes not requiring any degree of neighborhood between their values, therefore we follow the classical database query processing. When processing *metric attributes* i.e. data for which we are interested in distinguishing comparisons between two values which are close neighbors from those which lie far away from each other we follow the CQQL approach of similarity evaluation based on quantum measurement. An example of a *non-metric attribute* (or *database attribute*) is "Operating System" as shown in Section I-A.

Attributes such as "Price" comes naturally as *metric attributes* (or *similarity attributes*) because the users are definitely interested in low prices, therefore they are evaluated by means of similarity predicates: it becomes easy to derive an order of the values of these attributes and to use a continuous, and monotonic predicate  $p_{Price} : \text{dom}(Price) \rightarrow [0, 1]$  to associate their values with truth degrees.

*Definition 3 (Database Values Evaluation):* Let  $o$  be a database object. If  $A$  is a non-metric attribute such that  $o.A = a$ , then

$$\text{eval}(A = v, o) = \begin{cases} 1, & \text{if } v = a \\ 0, & \text{otherwise} \end{cases}$$

whenever  $a = null$  or  $v = null$  we take

$$\text{eval}(A = v, o) = \text{unknown}.$$

Let  $A$  be a metric attribute. Any value  $a \in \text{dom}(A)$ , is mapped by quantum encoding into a vector state  $\bar{a}$ . This give the

user a means to assign its own semantics to the resulting proximity values. The CQQL framework allows evaluation of similarity values by using any similarity measure  $s$  satisfying the following conditions:

- 1) for all  $a, b \in \text{dom}(A)$ ,  $s(\bar{a}, \bar{b}) \in [0, 1]$ .
- 2) for all  $a, b \in \text{dom}(A)$ ,  $s(\bar{a}, \bar{b}) = 1 \Leftrightarrow \bar{a} = \bar{b}$ .
- 3) for all  $a, b \in \text{dom}(A)$ ,  $s(\bar{a}, \bar{b}) = s(\bar{b}, \bar{a})$ .

A widely used such similarity measure is the cosine similarity  $s(\bar{a}, \bar{b}) = \frac{\bar{a} \cdot \bar{b}}{\|\bar{a}\| \|\bar{b}\|}$ .

*Definition 4 (Retrieval Values Evaluation):*

$$\text{eval}(A \approx v, o) = \begin{cases} s(\bar{v}, \bar{o.a}), & \text{if } v \neq \text{null and } a \neq \text{null} \\ \text{unknown}, & \text{otherwise} \end{cases} \quad (1)$$

Before a CQQL formula can be evaluated it has to be normalized. The normalization requires a special syntactical form starting with a formula that is in the prenex and disjunctive normal form. Then all common atoms  $\varphi$  are removed by applying the rule  $(\varphi \wedge \varphi_1) \vee (\varphi \wedge \varphi_2) = \varphi \wedge (\varphi_1 \vee \varphi_2)$  (derived from distributivity and absorption). The normalization algorithm is based on Boolean transformation rules and is described in [35].

*Definition 5 (Formula Evaluation):* Let  $\varphi_1 \wedge \varphi_2$ ,  $\varphi_1 \vee \varphi_2$  and  $\varphi$  normalized formulas. Then

$$\begin{aligned} \text{eval}(\varphi_1 \wedge \varphi_2, o) &= \text{eval}(\varphi_1, o) * \text{eval}(\varphi_2, o) \\ \text{eval}(\varphi_1 \vee \varphi_2, o) &= \text{eval}(\varphi_1, o) + \text{eval}(\varphi_2, o) - \\ &\text{eval}(\varphi_1, o) * \text{eval}(\varphi_2, o) \\ \text{eval}(\neg \varphi, o) &= 1 - \text{eval}(\varphi, o) \end{aligned}$$

Applying CQQL evaluation against a set of database objects we get a rank of all objects according to their score value. Integrating weights into CQQL can be achieved simply. The core idea is the direct transformation of a weighted conjunction or disjunction into a logical expression in which weight values are converted into 0-ary predicates using the rewriting rules below:

$$\begin{aligned} \varphi_1 \wedge_{\theta_1, \theta_2} \varphi_2 &\Rightarrow (\varphi_1 \vee \neg \theta_1) \wedge (\varphi_2 \vee \neg \theta_2) \\ \varphi_1 \vee_{\theta_1, \theta_2} \varphi_2 &\Rightarrow (\varphi_1 \wedge \theta_1) \vee (\varphi_2 \wedge \theta_2) \end{aligned}$$

These rules provide a way to transform weighted conjunction, weighted disjunctions into CQQL Boolean calculus. More technical details on these transformations and examples can be found in [37]. An intuitive interpretation of rewriting rules when considering only Boolean weights (0 and 1) is described in the below table.

$\theta_1$	$\theta_2$	$\varphi_1 \wedge_{\theta_1, \theta_2} \varphi_2$	$\varphi_1 \vee_{\theta_1, \theta_2} \varphi_2$	Explanation
0	0	1	0	none of $\varphi_1$ and $\varphi_2$ counts
0	1	$\varphi_2$	$\varphi_2$	only $\varphi_2$ counts
1	0	$\varphi_1$	$\varphi_1$	only $\varphi_1$ counts
1	1	$\varphi_1 \wedge \varphi_2$	$\varphi_1 \vee \varphi_2$	both $\varphi_1$ and $\varphi_2$ counts

Existential quantification and universal quantification in a CQQL query are evaluated by computing maximum respectively minimum of the weight values of the appropriate objects.

## V. CONJOINT ANALYSIS WITH CQQL

This scenario is centered on a weight learning algorithm, an adaptation of the weighted majority algorithm proposed in [29] and discussed in [6] and [7]. The main assumption, introduced by [29] and conforming with conjoint analysis principles ([22], [23]) is the compositional approach to overall preference as a weighted sum of individual preferences.

Let  $\mathcal{A} = \{A_1, \dots, A_n\}$  be a set of mutually independent attributes. Let  $\text{dom}(A_i)$  the value space of attribute  $A_i$ . Let  $\mathcal{O} = \{(a_1, \dots, a_n) | a_k \in \text{dom}(A_k), k = 1, \dots, n\}$  a set of possible objects built over  $\mathcal{A}$ . Let  $\mathcal{S}$  be a set of stimuli and  $\mathcal{M}$  be the set of all minterms over  $\mathcal{A}$ . The overall idea of the conjoint process is to obtain an aggregated preference on possible minterms and use this preference to compute an ordering on  $\mathcal{M}$ . Given minterm preferences  $P_{\mathcal{M}, p}$  for all respondents  $p \in \mathcal{P}$  the algorithm learn optimal  $\{w_p \in [0, 1] | p \in \mathcal{P}\}$  such that

$$P_{\mathcal{M}}(m_1, m_2) = \sum_{p \in \mathcal{P}} w_p \cdot P_{\mathcal{M}, p}(m_1, m_2), \quad m_1, m_2 \in \mathcal{M}$$

such that a specific loss function is minimized. This solution uses a loss function defined as the [0,1] normalization of the number of discordant preferences between  $\hat{P}_{\mathcal{M}}$  – a preference function computed from experts evaluations, and  $P_{\mathcal{M}}$  – the preference computed by the learner:

$$\text{Loss}(P_{\mathcal{M}}, \hat{P}_{\mathcal{M}}) = \frac{\sum_{\hat{P}_{\mathcal{M}}(x, y)=1} (1 - P_{\mathcal{M}}(x, y))}{|\hat{P}_{\mathcal{M}}|}$$

As largely discussed in [6] and [7], this loss function has a probabilistic interpretation: if  $P_{\mathcal{M}}(x, y)$  is interpreted as the probability that  $y$  is preferred to  $x$  then  $\text{Loss}(P_{\mathcal{M}}, \hat{P}_{\mathcal{M}})$  is the probability of disagreement of  $P_{\mathcal{M}}$  with the feedback on  $(x, y)$  from  $\hat{P}_{\mathcal{M}}$ .

As we use this algorithm on minterm preferences, at first, we need a mechanism to compute minterm preferences from object preferences.

Recall that the goal is to learn a formula as a disjunction of most important minterms. It defines a logical representation of the respondent interests and offers an explanation on user's interests. To compute it we compute a ranking of formula minterms from preferences on database objects and take the most dominant minterms.



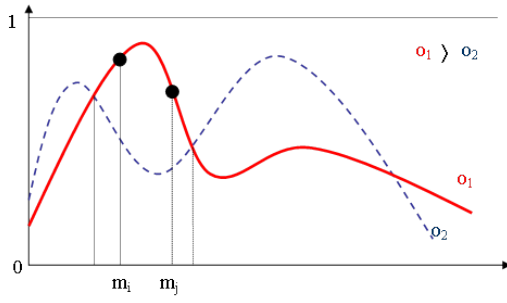


Fig. 2. Geometric Interpretation of Rule 1 on minterm preferences

### A. Computing Minterm Preferences

**Definition 6 (Minterm Preference):** Let  $P_{\mathcal{O}}$  be a preference. Let  $\mathcal{M}$  be the set of minterms of  $U$ . Let  $o_1, o_2 \in \mathcal{O}$  and  $m_i, m_j \in \mathcal{M}$ . The minterms preference  $P_{\mathcal{M}}$  is computed according with the following rules:

$$\left. \begin{array}{l} P_{\mathcal{O}}(o_1, o_2) = 0 \\ eval(m_i, o_1) > eval(m_j, o_2) \\ eval(m_i, o_1) > eval(m_j, o_1) \end{array} \right\} \Rightarrow P_{\mathcal{M}}(m_i, m_j) = 0$$

$$\left. \begin{array}{l} P_{\mathcal{O}}(o_1, o_2) = 0 \\ eval(m_j, o_1) > eval(m_i, o_2) \\ eval(m_j, o_1) > eval(m_i, o_1) \end{array} \right\} \Rightarrow P_{\mathcal{M}}(m_i, m_j) = 1$$

All remaining unassigned preferences are considered *unknown*.

Basically, "minterms that are more similar in CQQL evaluation against preferred objects are preferred". This heuristic has a geometric interpretation. Figure 2 shows minterms  $m_i$  and  $m_j$  such that  $P_{\mathcal{M}}(m_i, m_j) = 0$ . We consider only the regions where the CQQL minterm evaluation of the preferred object ( $o_1$  in Figure 2) is better than minterm evaluation on the less preferred object ( $o_2$ ) and inside these regions we compute preferences induced by the the ordering produced by CQQL minterm evaluation.

**Proposition 2:** The minterm preference and object preference are monotonic with respect of CQQL evaluation i.e.,

$$P_{\mathcal{O}}(o_1, o_2) = P_{\mathcal{M}}(m_i, m_j) = 0 \text{ iff } eval(m_i, o_1) > eval(m_j, o_2)$$

and

$$P_{\mathcal{O}}(o_1, o_2) = P_{\mathcal{M}}(m_i, m_j) = 1 \text{ iff } eval(m_j, o_2) > eval(m_i, o_1).$$

**Proof:** The first relation yields directly using the first rule from Definition 6. The second comes from the preference symmetry ( $P(x, y) = 0 \Leftrightarrow P(y, x) = 1$ ) and second rule. ■

### B. Performing Conjoint Analysis

The conjoint process takes place in a sequence of rounds, one for each interview question and for all respondents. The complete description is depicted in Algorithm 2. Basically, on the  $q$ -th iteration we have  $\mathcal{M}$  the set of minterms and  $P_{\mathcal{M},p}^{(q)}$  the (computed) respondents preferences on  $\mathcal{M}$ . Then, the learner receives feedback from the environment. We assume that this feedback comes as a preference function on minterms, previously computed using the same techniques from experts

**Data:**  $\mathcal{O}, \mathcal{S}, \mathcal{Q}$  interview

**Data:**  $r \in [0, 1]$ . Initial  $w^{(1)} \in [0, 1]^{|P|}$  with

$$\sum_{j=1}^{|P|} w_j^{(1)} = 1$$

**Result:**  $U, R_1, \dots, R_l$

**foreach**  $q \in \mathcal{Q}$  **do**

Update preferences  $\{P_{\mathcal{S},p}^{(q)} | p \in \mathcal{P}\}$ ;

Compute minterm preferences  $\{P_{\mathcal{M},p}^{(q)} | p \in \mathcal{P}\}$ ;

Aggregate

$$P_{\mathcal{M}}(m_i, m_j) = \sum_{p \in \mathcal{P}} w_p^{(q)} \cdot P_{\mathcal{M},p}^{(q)}(m_i, m_j);$$

$$L_{(q)} = Loss(P_{\mathcal{M}}^{(q)}, \hat{P}_{\mathcal{M}});$$

$$\text{Update } w_p^{(next(q))} = \frac{1}{C^{(q)}} w_p^{(q)} \cdot r^{L_{(q)}}, \text{ for all } p \in \mathcal{P};$$

**end**

Compute  $k\_means\_Lloyd(P_{\mathcal{M}}, [0, 1],$

$\{0, 1, unknown\}$ );

Compute ordering  $\rho_{\mathcal{M}} = orderByPrefs(\mathcal{M}, P_{\mathcal{M}})$ ;

Compute  $U$  as a disjunction of the first  $k$  minterms ordered by  $\rho_{\mathcal{M}}$ ;

Compute decision rules  $R_1, \dots, R_l$ ;

**Algorithm 2:** CA inside a Logical Framework

inputs. At each step we update the new weight vector as  $w_p^{(next(q))} = \frac{1}{C^{(q)}} w_p^{(q)} \cdot r^{Loss(P_{\mathcal{M}}^{(q)}, \hat{P}_{\mathcal{M}})}$  for all  $p \in \mathcal{P}$  where  $r \in [0, 1]$  is a calibration constant and  $C^{(q)}$  is an normalization constant chosen so that  $\sum_{p \in \mathcal{P}} w_p^{(next(q))} = 1$ .

The "majority weighted" solution for preference aggregation is largely used by machine learning community. It obtains a partial preference  $P_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \rightarrow [0, 1]$ . However, Algorithm 1 works with binary preferences, therefore we have to perform a clustering from  $P_{\mathcal{M}}(m_1, m_2) \in [0, 1]$  to  $P_{\mathcal{M}}(m_1, m_2) \in \{0, 1, unknown\}$ . There is large literature on clustering methods ([34] is a recent survey). By now, because our clustering space is Euclidean and one dimensional we considered standard k-means clustering procedure (Lloyd's algorithm) to produce three clusters: "0", "0.5" ("unknown") and "1".

### C. Creating and Interpreting Decision Rules

Following the above learning process we get a formula  $U$  as a disjunction of the most  $K$  dominant minterms,  $m_{i_1}, \dots, m_{i_K}$ . To obtain a ruleset based on  $U$ :

- 1) *Compute a conjunctive normal form.* Applying the distributivity laws to  $U$ , then eliminate duplicates ( $L \vee L \Rightarrow L$ ) and tautology (delete  $L \vee \bar{L}$ ). As a result of this step we obtain  $U$  in the CNF form.
- 2) *Find unit clauses.* (i.e. the clauses containing only one literal). All these unit clauses (or facts) are mandatory rules of our ruleset.
- 3) *Simplify unit clauses.* All unit clauses must be true therefore we replace the corresponding literals from  $U$ , then apply the usual Boolean computation. As a result all clauses of  $U$  do not contain any literals from unit clauses. Let  $\mathcal{C}$  be the set of all clauses of  $U$ .
- 4) *Transform each clause to a rule.* For each  $C \in \mathcal{C}$ , let  $C = L_1 \vee \dots \vee L_j$  and let  $L_1$  be the desired conclusion of



the rule<sup>4</sup>. Then the rule corresponding to  $C$  is obtained by simple transformation to implication i.e.

$$\overline{((L_2 \vee \dots \vee L_j))} \vee L_1 \Rightarrow \overline{(L_2 \wedge \dots \wedge L_j)} \vee L_1 \Rightarrow R_C : \overline{L_2} \wedge \dots \wedge \overline{L_j} \rightarrow L_1.$$

Eliminate possible double negation of the rule literals (apply  $\overline{\overline{L}} \Rightarrow L$ ).

## VI. EXPERIMENTAL RESULTS

Our experiments were related to an use case considering 15 attributes. We considered one scale (1..9) for all attribute ratings and the experiment uses a slightly modified variant of logistic function  $f(n) = \frac{1}{(1+e^{-n+5})}$  for rating mapping. This maps the scale 1..9 into 0.0179, 0.0474, 0.1192, 0.2689, 0.5, 0.7310, 0.8808, 0.9525, 0.9820.

The training data had 16 objects previously ranked by experts, from which we derived 32 stimuli. Each interview had 15 pairwise comparisons containing stimuli of similar utility, i.e., a pair comparison  $(s_1, s_2)$  was generated iff  $|u_{s_1} - u_{s_2}| < \varepsilon_p$  where  $\varepsilon_p$  is a threshold depending on the respondent. All questions used the same evaluation scale, 1..9. We computed a CQQL formula based on 1,2,3,5, and 10 most dominant minterms. The respondents scores and preferences were set up at random and we performed 100 simulations.

The result quality was measured by using Spearman's correlation, to compare the training data rank obtained by CQQL evaluation with the experts evaluation as shown in the below table:

No. of minterms	Best Spearman	Worse Spearman	Average
1	0.4814	0.1231	0.4002
2	0.7871	0.3612	0.72
3	0.91	0.7713	0.8724
5	0.9159	0.7023	0.8813
10	0.9921	0.7718	0.8763

The actual results show that using the first 3 or 5 dominant minterms give a correlation comparable with the case when the first 10 most dominant minterms were used. When creating decision rules the main complexity parameter is the number of minterms to be used: using the first two dominant minterms as a starting base will produce the simplest rules, as the ones described in Section I-A<sup>5</sup>. Such rules can be easily interpreted both by a human expert or a rule engine. However, when four or more dominant minterms were used then a much larger ruleset is computed and the complexity of each rule increase too. Typically such a ruleset will be processed by means of a rule engine although if a human expert is interested only in partial decisions he can use only a subset of these rules ([12]). The settings of the decision rules creation process considered

<sup>4</sup>This choice is related to the human expert interests on one or other attribute. Other solutions may consider the user's high rated attributes as a choice.

<sup>5</sup>Notice that the rules produced as described in Section V-C can also be grouped by head towards a much compact writing. This is easy when two or three minterms were used in the CQQL formula, but becomes not so much useful when more minterms were considered.

the first two dominant minterms of the obtained full DNF formula and the first three user high rated attributes as rule conclusions.

## VII. CONCLUSIONS AND FUTURE RESEARCH

This article has shown how conjoint analysis can be modeled using the tools offered by CQQL, a logic-based similarity query language. The advantage of the approach lies in the expressive power and flexibility of logic to encode the conjoint model. We obtained an ACA-CQQL interpretation as ranking of potential products according with some preexistent pattern (CQQL query) and used some learning algorithms to compute the solution. Doing conjoint analysis inside a logic-based framework creates opportunities to apply such data analysis strategy to other kind of problems such as delivering recommendations inside social networks, and deriving user's profiles from mining social activities.

We plan to extend our approach. The model formulation offers the opportunity to tune the conjoint problem by performing a choice on a number of parameter as discussing below.

*Attribute classification and preferences normalization.* CQQL supports both metric and non-metric attributes therefore both crisp and non-crisp data can be handled. In addition, each metric attribute may come with his own rating scale. The length of the scale might have to be considered while creates much more granularity of weights. The actual experiment uses classical exponential utility but other normalization approach should be considered in the future research.

*Stimuli and pairwise comparisons.* While in our experiments we've extracted stimuli on a heuristic base, possibly a more systematic approach (using techniques already provided in economics research) may yield to better results. Stimuli with two conjuncts are very easy to be understood by respondents, but using 3 or more conjuncts in the stimuli will improve the quality of the extended ordering on the training objects (see Definition 2). Secondly, CA also considers questions with more than 2 stimuli choices: in this case, we cannot use bipolar scales, therefore the preference order induced by question rating is not unique<sup>6</sup>. Finally, in the present work we ignore neutral rated comparisons but we aim to investigate other approaches in further research.

*The interview creation.* This work does not impose any restrictions on the interview creation, therefore the list of questions composing the respondent interview may support various orderings such as "by question importance" or "by question difficulty". In all cases a strategy should not violate the transitivity property of preferences: non-transitive pairs creates inconsistent problems (not all preferences can be simultaneously satisfied). An interview creation strategy

<sup>6</sup>Receiving a question  $q = (s_1, s_2, s_3)$  after scoring  $r_1 < r_2 < r_3$  we get  $s_1 \preceq s_2 \preceq s_3$  but we have to understand how to derive ratings for pairs such as  $s_1 \preceq s_2$ .

should detect this issue in the interview creation stage (immediately after the user rated a new pair comparison) and not during the learning stage. In addition, because during an interview, traditional ACA may also ask respondents to rate individual products (complete objects) on a purchase likelihood scale, we aim to generate these by considering intermediary learned formulas for tuning the preferences set towards fasten the convergence to an acceptable solution.

*The learning strategy.* Actually, the algorithm uses a loss function previously tested by other research. However we intend to investigate the performance of other loss functions such as

$$Loss(P, \hat{P}) = \frac{\sum_{(x,y) \in X \times X} (\hat{P}(x,y) - P(x,y))}{|X \times X|}$$

which is similar with Kendall's  $\tau$  rank correlation coefficient. The learning strategy implementation supports parallelization by distributing processing on persons or person groups, therefore an ACA-CQQL application should be able to offer fast answers. We also intend to investigate other learning algorithm such as ones described by [21]. Obtaining various ACA-CQQL conjoint representation that can be interpreted and explained is one of the powerful achievements of the proposed solution. We get explanations both in terms of user's most preferred attributes/values, and preferred products/services. In future research alternative solutions to derive minterm preferences have to be examined.

*Decision rules.* The obtained rule set can be subject of interpretation in various ways from simpler (as seen in Example 1) to solutions involving different semantics such as incomplete/imprecise information, [36], probabilistic models [32], or plausibility-based models [10], [11]). We plan to address some of these alternatives in subsequent research.

## REFERENCES

- [1] G. Adomavicius, and A. Tuzhilin, Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 6, June 2005, pp. 734-749.
- [2] N.H. Anderson. Foundations of information integration theory. Acad Press, 1981.
- [3] D. Baier and M. Brusch (Eds.) Conjointanalyse, Methoden - Anwendungen - Praxisbeispiele, Springer, Berlin, 2009.
- [4] D. Baier. Conjoint Measurement in der Innovationsmarktforschung, in: Baaken, Thomas; Höft, Uwe; Kesting, Tobias (Hrsg.), Marketing für Innovationen - Wie innovative Unternehmen die Bedürfnisse ihrer Kunden erfüllen, Harland Media, Münster, ISBN-13 978-3-938363-42-3.
- [5] F. J. Carmone and P. E. Green. Model Misspecification in Multiattribute Parameter Estimation. Journal of Marketing Research, 18, 1981, pp.87-93.
- [6] W. Cohen, R.E. Schapire and Y. Singer. Learning to Order Things. Advances in Neural Information Processing Systems 10, Morgan Kaufmann, 1998.
- [7] W. Cohen, R.E. Schapire and Y. Singer. Learning to Order Things. Journal of Artificial Intelligence Research 10 (1999), pp. 213-270.
- [8] P.C. Fishburn. Methods of Estimating Additive Utilities. Management Science, Vol. 13, 1967, pp.435-453.
- [9] P.C. Fishburn. Utility Theory. Management Science, Vol. 14, 1968, pp. 335-367.
- [10] N. Friedman, and J.Y. Halpern. Plausibility measures and default reasoning. Journal of the ACM, 48:648-685, 2001.
- [11] A. Giurca. A Logic with Plausibility. Annales of Craiova University, Mathematics and Computer Science Series, XXVII, pp.105-115, 2000.
- [12] A. Giurca, D. Gasevic, and K. Taveter Handbook of Research on Emerging Rule-Based Languages and Technologies: Open Solutions and Approaches, Vol 1-2, IGI Publishers, 2009, ISBN13: 978-1-60566-402-6.
- [13] P. E. Green. Hybrid Models for Conjoint Analysis: An Expository Review. Journal of Marketing Research, Vol. 21, 1984, pp. 155-169.
- [14] P. E. Green and M. T. Devita. A Complementary Model of Consumer Utility for Item Collections. Journal of Consumer Research, 1, (December), 1974, pp.56-67.
- [15] P. E. Green and M. T. Devita. An Interaction Model of Consumer Utility. Journal of Consumer Research, 2, (September), 1975, pp.146-153.
- [16] P. E. Green, S. M. Goldberg, and M. Montemayor. A Hybrid Utility Estimation Model for Conjoint Analysis. Journal of Marketing, Vol. 45, Winter 1981, pp.33-41.
- [17] P. E. Green and A. M. Krieger. Conjoint Analysis with Product-Positioning Applications. J. Eliashberg, and G.L. Lilien (Eds.): Handbook in Operations Research and Management Science, Vol. 5, North-Holland, 1993, pp. 467-515.
- [18] P. E. Green and V. Rao. Conjoint measurement for quantifying judgmental data. Journal of Marketing Research, 8, 1971, pp.355-363.
- [19] P. E. Green and V. Srinivasan. Conjoint Analysis in Consumer Research: Issues and Outlook. Journal of Consumer Research, Vol. 5, September 1978, pp. 103-123.
- [20] R.T. Hoepfl and G. P. Huber. A Study of Self-Explicated Utility Models. Behavioral Science, Vol. 15, 1970, pp. 408-414.
- [21] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker. Label ranking by learning pairwise preferences, Artificial Intelligence, Vol. 172:16-17, Nov. 2008, pp. 1897-1916.
- [22] R. M. Johnson. Tradeoff Analysis of Consumer Values. J. of Marketing Research, 1974, pp. 121-127.
- [23] R. M. Johnson. Accuracy of Utility Estimation in ACA. Working Paper, Sawtooth Software, Sequim, WA, April 1987.
- [24] R. M. Johnson. Comment on Adaptive Conjoint Analysis: Some Caveats and Suggestions. Journal of Marketing Research, 28, 1991, pp. 223-225.
- [25] R. M. Johnson. Comments on Studies Dealing With ACA Validity and Accuracy, With Suggestions for Future Research, 1991 published by Sawtooth Software.
- [26] R.L. Keeney and H. Raiffa. Decisions with multiple objectives: Preferences and value tradeoffs. Wiley Series in Probability and Mathematical Statistics. NY: John Wiley & Sons, 1976.
- [27] K. Lancaster. A new approach to consumer theory. J. of Political Economy, 74, 1966, pp.132-157.
- [28] R. Likert. A Technique for the Measurement of Attitudes. Archives of Psychology 140, 1932, pp.1-55.
- [29] N. Littlestone and M. Warmuth. The weighted majority algorithm. Information and Computation, 108 (2), 1994, pp. 212-261.
- [30] R.D. Luce and J. W. Tukey. Simultaneous Conjoint Measurement: A New Type of Fundamental Measurement. J. of Mathematical Psychology, 1, 1964, pp.1-27.
- [31] R.D. Luce and P. Suppes. Preference, utility and subjective probability, in Luce, R.D., Bush, R.R., and Galanter, E. (Eds.), Handbook of Mathematical Psychology, III, New York: Wiley, 1965, pp.235-406.
- [32] N. J. Nilsson. Probabilistic logic. Artificial Intelligence 28(1):1986, pp.71-87.
- [33] K.L. Norman and J.J. Louviere. Integration of attributes in public bus transportation: two modeling approaches. Journal of Applied Psychology, 59, 6, 1974, pp.753-758.
- [34] L. Rokach. A survey of Clustering Algorithms. Data Mining and Knowledge Discovery Handbook, 2nd ed. Springer 2010, pp. 269-298, ISBN 978-0-387-09822-7.
- [35] I. Schmitt. QQL: A DB&IR Query Language. VLDB J., 17(1):39-56, 2008.
- [36] G. Wagner. Logic Programming with Strong Negation and Inexact Predicates. Journal of Logic and Computation 1(6): 835-859 (1991).
- [37] D. Zellhöfer and I. Schmitt. A preference-based approach for interactive weight learning: learning weights within a logic-based query language. Distributed and Parallel Databases (2010) 27: 31-51, DOI 10.1007/s10619-009-7049-4.

# Extending the definition of $\beta$ -consistent biclustering for feature selection

Antonio Mucherino<sup>‡</sup>

<sup>‡</sup>CERFACS, Toulouse, France.  
 mucherino@cerfacs.fr

**Abstract**—Consistent biclusterings of sets of data are useful for solving feature selection and classification problems. The problem of finding a consistent biclustering can be formulated as a combinatorial optimization problem, and it can be solved by the employment of a recently proposed VNS-based heuristic. In this context, the concept of  $\beta$ -consistent biclustering has been introduced for dealing with noisy data and experimental errors. However, the given definition for  $\beta$ -consistent biclustering is coherent only when sets containing non-negative data are considered. This paper extends the definition of  $\beta$ -consistent biclustering to negative data and shows, through computational experiments, that the employment of the new definition allows to perform better classifications on a well-known test problem.

## I. INTRODUCTION

CLASSIFICATION problems in data mining aim at finding a suitable partition of the samples contained in a certain set of data. Various classification techniques have been proposed over the last years, and they have been applied to various problems arising in applied fields [8], [13]. Recently, a new approach for classification have been proposed in [2], [3], which is based on the concept of *consistent biclustering*. Samples and features of a set of data are organized on the columns and on the rows, respectively, of a matrix, and a partition in biclusters of this matrix (the so-called *biclustering*) can be found with the aim of performing supervised classifications. If the biclustering is *consistent*, then the knowledge acquired by finding the biclustering can be exploited for performing good-quality classifications (see Section II for more details).

When real data are considered, i.e. data obtained by experimental techniques, the matrix representing the whole set of data does not usually admit any consistent biclustering. This may be due to the fact that some of the features which are considered for describing the samples are actually not pertinent to the problem. A way to overcome to this issue is then to remove all these features from the set of data. During this process, however, useful features should not be discarded [2].

Since experimentally obtained data are usually noisy, even small errors introduced in the set of data may cause the loss of the consistency of the found biclusterings. This issue has been firstly addressed in [2] and, successively, the concepts of  $\alpha$ -consistent biclustering and  $\beta$ -consistent biclustering have been introduced in [15] with the aim of efficiently managing noisy data and errors. When looking for biclusterings satisfying the  $\alpha$ -consistency or the  $\beta$ -consistency property, a larger number of features (depending on the parameters  $\alpha$  and  $\beta$ ) need to be discarded from the set of data, because all the features that are

sensitive to experimental errors are supposed to be identified and removed.

While the  $\alpha$ -consistency property helped in the management of errors and noise, biclusterings satisfying the  $\beta$ -consistency property showed instead a weird behavior [12], [15]. While larger  $\alpha$  values allowed for finding  $\alpha$ -consistent biclusterings in which *better* features were selected, so that better-quality classifications were actually possible by exploiting the biclustering,  $\beta$ -consistent biclusterings with larger  $\beta$  values did not follow this general trend. In fact, the definition of  $\beta$ -consistent biclustering given in [15] is coherent only if the considered set contains non-negative data only. The aim of this paper is therefore to extend the definition of  $\beta$ -consistent biclustering to negative data. The definition given in this paper actually allows for finding correct  $\beta$ -consistent biclusterings even when negative data are available, as it is usually the case when real-life problems are considered.

The rest of the paper is organized as follows. Section II will provide some more details about how to find consistent biclusterings and how to use this knowledge for performing supervised classifications. Section III will briefly describe a recently proposed heuristic for the solution of the combinatorial optimization problem for the identification of consistent biclusterings of training sets. Then, the concept of  $\beta$ -consistent biclustering will be deeply discussed. In Section IV, an extended version of its formal definition will be presented. In Section V, a well-known set of data will be scaled so that it only contains non-negative entries, and an existing algorithm will be employed for finding  $\beta$ -consistent biclusterings. The experiments show that better classifications can be obtained by exploiting these new biclusterings. Results show that the new  $\beta$ -consistent biclusterings allow to obtain classifications with no misclassified elements, as it was instead the case in previous works. Section VI concludes the paper.

## II. CONSISTENT BICLUSTERING FOR SUPERVISED CLASSIFICATIONS

Let  $A \equiv \{a_{ij}\}$  be an  $m \times n$  matrix representing a set of data. The matrix  $A$  contains  $n$  samples (column by column) and  $m$  features (row by row). A *bicluster* is a submatrix of  $A$ , which is able to group together a subset of samples (a class  $S_r$ ) and a subset of features (a class  $F_r$ ) of the set of data. Finally, a *biclustering*

$$B = \{(S_1, F_1), (S_2, F_2), \dots, (S_k, F_k)\}$$

is a partition of  $A$  in disjoint biclusters which covers  $A$ , i.e. the following conditions must be satisfied:

$$\bigcup_{r=1}^k S_r \equiv A, \quad S_\zeta \cap S_\xi = \emptyset \quad 1 \leq \zeta \neq \xi \leq k,$$

$$\bigcup_{r=1}^k F_r \equiv A, \quad F_\zeta \cap F_\xi = \emptyset \quad 1 \leq \zeta \neq \xi \leq k,$$

where  $k \leq \min(n, m)$  is the considered number of biclusters [2].

Biclusterings of sets of data are usually searched by unsupervised techniques, where it is supposed that no information about the data is available. The interested reader can refer to [10] for a recent survey. In this approach, it is instead supposed that the set of data  $A$  is a training set, i.e.  $A$  is a set for which the classification of its samples is already known. The corresponding biclustering is therefore computed by employing a supervised technique, and the found biclustering is then exploited for classifying samples having no known classification.

Let us suppose that  $A$  is a training set. Therefore, the classification of its samples in  $k$  classes is known. From this classification, it is possible to construct a classification of its features in  $k$  classes. The basic idea is to assign each feature to the class  $F_{\hat{r}}$  (with  $\hat{r} \in \{1, 2, \dots, k\}$ ) such that it is mostly expressed (i.e. *it has higher value*), in average, in the class of samples  $S_{\hat{r}}$ . The reader is referred to [2], [12], [15] for details about this supervised procedure. Note that the same procedure can be inverted and it can be used for finding a classification of the samples of  $A$  from a known classification of its features.

By combining the two classifications, the one for the samples of  $A$  and the other one for the features of  $A$ , a biclustering can be defined for the matrix  $A$ . The supervised procedure mentioned above can construct classifications of the samples from classifications of the features, and vice versa. If the biclustering remains unchanged when the supervised procedure is applied, then it is said to be *consistent*. In other words, the biclustering is consistent if the classification of the samples (the features) suffices for correctly reconstructing the biclustering.

Consistent biclusterings can be very useful for performing supervised classifications. Let  $\hat{A}$  be a set of data which is not a training set and that it is related to the same classification problem as the set  $A$ . No information regarding the classification of the samples in  $\hat{A}$  is available, but  $\hat{A}$  contains the same features of  $A$  and a classification of these features is known because a biclustering for  $A$  is available. By using the supervised procedure, then, a classification for the samples of  $\hat{A}$  can be found from the known classification of its features. Since the biclustering of  $A$  is consistent, the procedure is able to find the correct classification for the samples in  $A$ , and therefore it should be able to do most likely the same for the samples in  $\hat{A}$  [2].

Let  $f_{ir}$  be a binary parameter which indicates if the  $i^{th}$  feature belongs to the class  $F_r$  of features ( $f_{ir} = 1$ ) or not

( $f_{ir} = 0$ ). Let  $x \equiv \{x_1, x_2, \dots, x_m\}$  be a binary vector of variables, where  $x_i$  is 1 if the  $i^{th}$  feature of  $A$  is selected, and it is 0 otherwise. Let us also indicate with the symbol  $A[x]$  the submatrix of  $A$  obtained by selecting only the features (rows) of  $A$  for which  $x_i = 1$ .

### II.1 Definition

A biclustering for  $A[x]$  is consistent if and only if,  $\forall \hat{r}, \xi \in \{1, 2, \dots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$ , the following inequality is satisfied [2]:

$$\frac{\sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^m f_{i\hat{r}} x_i} > \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i}. \quad (1)$$

Note that the two fractions in (1) are used for computing the *centroids* of the considered biclusters (for each sample in  $S_{\hat{r}}$ , the average over the features belonging to the same class is computed). On the left hand side of (1), the  $j^{th}$  component of the centroid of the bicluster  $(S_{\hat{r}}, F_{\hat{r}})$  is computed. On the right hand side of (1), the  $j^{th}$  component of the centroid of the bicluster  $(S_{\hat{r}}, F_\xi)$  is computed. In order to have a consistent biclustering, all components of the centroid of  $(S_{\hat{r}}, F_{\hat{r}})$  must have a larger value.

In order to overcome issues related to sets of data containing noisy data, the concepts of  $\alpha$ -consistent biclustering and  $\beta$ -consistent biclustering have been introduced in [15]. The basic idea is to artificially increase the margin between the centroids of the different biclusters in the constraints (1). In this way, small variations due to noisy data and errors should not be able to spoil the classifications performed by exploiting the found biclusterings.

### II.2 Definition

A biclustering for  $A[x]$  is  $\alpha$ -consistent if and only if,  $\forall \hat{r}, \xi \in \{1, 2, \dots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$ , the following inequality is satisfied [15]:

$$\frac{\sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^m f_{i\hat{r}} x_i} > \alpha + \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i}, \quad (2)$$

where  $\alpha > 0$ .

The additive parameter  $\alpha > 0$  is used to guarantee that the margin between the centroid of  $(S_{\hat{r}}, F_{\hat{r}})$  and any other bicluster concerning  $S_{\hat{r}}$  is at least greater than  $\alpha$ , independently from the considered data. Similarly, in the case of  $\beta$ -consistent biclustering, a multiplicative parameter  $\beta$  is used.

### II.3 Definition

A biclustering for  $A[x]$  is  $\beta$ -consistent if and only if,  $\forall \hat{r}, \xi \in \{1, 2, \dots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$ , the following inequality is

satisfied [15]:

$$\frac{\sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^m f_{i\hat{r}} x_i} > \beta \times \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i}, \quad (3)$$

where  $\beta > 1$ .

Note that different values for the parameters  $\alpha$  and  $\beta$  could be used for different components of the centroids. Usually, however, only one value is set up for all the components. More details about  $\alpha$ -consistent and  $\beta$ -consistent biclustering are given in Section IV.

In real-life applications, there are usually no biclusterings which are consistent,  $\alpha$ -consistent or  $\beta$ -consistent if all the features are selected (this situation corresponds to a binary vector  $x$  with all its components equal to 1). As already mentioned in the Introduction, this is consequence of the fact that some of the considered features may not be pertinent. Such features must therefore be removed from the set of data, while the total number of considered features must be maximized in order to lose the minimum amount of information.

To this aim, the following combinatorial optimization problem can be considered:

$$\max_x \left( f(x) = \sum_{i=1}^m x_i \right), \quad (4)$$

subject to constraints (1), (2) or (3) depending on the fact that a consistent,  $\alpha$ -consistent or  $\beta$ -consistent biclustering, respectively, is searched. These three optimization problems are all NP-hard [9], and different heuristic algorithms have been proposed in order to solve such problems [2], [12], [15]. In previous works, consistent biclusterings have been found for sets of data related to:

- gene expressions of human tissues from healthy and sick (affected by cancer) patients [12], [16];
- patients diagnosed with acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML) diseases [2], [4], [12], [15];
- the Human Gene Expression (HuGE) Index [2], [6], [15];
- wine fermentations [14], [17].

### III. A VNS-BASED HEURISTIC FOR FINDING BICLUSTERINGS

The optimization problems (4)-(1), (4)-(2) and (4)-(3) are combinatorial problems with fractional constraints and binary decision variables. In order to solve these optimization problems, we employ a recently proposed heuristic [12], which is based on a reformulation of these problems as bilevel programs.

Let us introduce new continuous variables

$$y_r, r = 1, 2, \dots, k.$$

Each variable  $y_r$  is related to the bicluster  $(S_r, F_r)$  of a possible biclustering, and it represents the percentage of features

that are selected in that bicluster. The bilevel reformulation can be written for problem (4)-(1) as follows:

$$\min_y \left( g(x, y) = \sum_{r=1}^k \left[ (1 - y_r) + \sum_{\xi=1: \xi \neq r}^k c(x, r, \xi) \right] \right)$$

s.t.:

$$x = \arg \max_x \left( f(x) = \sum_{i=1}^m x_i \right)$$

$$\text{s. t. } \begin{cases} \sum_{i=1}^m f_{ir} x_i = \lfloor y_r \sum_{i=1}^m f_{ir} \rfloor \quad \forall r \in \{1, \dots, k\} \\ \frac{1}{y_{\hat{r}}} \sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i > \frac{1}{y_{\xi}} \sum_{i=1}^m a_{ij} f_{i\xi} x_i \end{cases}$$

$$\sum_{r=1}^k y_r \leq 1,$$

where

$$c(x, \hat{r}, \xi) = \sum_{j \in S_{\hat{r}}} \left| \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i} - \frac{\sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^m f_{i\hat{r}} x_i} \right|_+,$$

and the symbol  $|\cdot|_+$  represents the function which returns its argument if it is positive, and it returns 0 otherwise. Solving this bilevel program is equivalent to solving the original problem (4)-(1). For more details, the reader is referred to [12].

Reformulating a single optimization problem in a bilevel program may not seem convenient, because the complexity of the problem might increase. However, this reformulation allowed the development of an efficient heuristic for the solution of the problem. The inner problem of this bilevel program is linear, and therefore it can be solved by standard methods for linear optimization. The heuristic is inspired by the Variable Neighborhood Search (VNS) [5], [11] and it only acts on the new introduced variables  $y_r$ , with  $r = 1, 2, \dots, k$ . At each iteration of the algorithm, the inner problem is exactly solved and a set of values for the original variables  $x_i$ , with  $i = 1, 2, \dots, m$ , is obtained. The main intuition is that the exact solution of the inner problem helps the heuristic in converging towards the desired biclusterings.

Algorithm 1 gives a sketch of this VNS-based heuristic [12]. It is composed by two VNS's which are nested. An adaptive value for the percentage of unselected features, *unsel*, is kept small at the beginning (*unsel*  $\simeq$  0) of the heuristic, and then it increases when no better solutions can be found. In this way, the algorithm firstly tries to find solutions where the number of selected features is high. After, solutions where fewer features are selected are also allowed. For each neighbor of the first VNS, there is a full execution of the second VNS. The neighbors of the second VNS are generated so that the set of variables  $y_r$  can be slightly perturbed at the beginning (*range* = *starting\_range*), and larger perturbations can be performed only when no better solutions can be found by

**Algorithm 1** A VNS-based heuristic for feature selection.

---

```

1: let  $iter = 0$ ;
2: let  $x_i = 1, \forall i \in \{1, 2, \dots, m\}$ ;
3: let  $y_r = \sum_i f_{ir}/m, \forall r \in \{1, 2, \dots, k\}$ ;
4: let  $y_r^{best} = y_r, \forall r \in \{1, 2, \dots, k\}$ ;
5: let  $range = starting\_range$ ;
6: let  $unsel = 0$ ;
7: while ((1) unsatisfied and  $unsel \leq max\_unsel$ ) do
8:   while ((1) unsatisfied and  $range \leq max\_range$ ) do
9:     let  $iter = iter + 1$ ;
10:    solve inner optimization problem (linear & cont.);
11:    if (constraints (1) unsatisfied) then
12:      increase  $range$ ;
13:      if ( $g$  has improved) then
14:        let  $y_r^{best} = y_r, \forall r \in \{1, 2, \dots, k\}$ ;
15:        let  $range = starting\_range$ ;
16:      end if
17:      let  $y_r = y_r^{best}, \forall r \in \{1, 2, \dots, k\}$ ;
18:      let  $r' = \text{random in } \{1, 2, \dots, k\}$ ;
19:      choose randomly  $y_{r'}$  in  $[y_{r'} - range, y_{r'} + range]$ ;
20:      let  $r'' = \text{random in } \{1, 2, \dots, k\} : r' \neq r''$ ;
21:      set  $y_{r''}$  so that  $1 - unsel \leq \sum_r y_r \leq 1$ ;
22:    end if
23:  end while
24:  if (constraints (1) unsatisfied) then
25:    increase  $unsel$ ;
26:  end if
27: end while

```

---

considering the current neighbor. As for all heuristics, there is no guarantee that the biclusterings that are found by the heuristic are the ones with the largest number of selected features. However, multi-start techniques may be for example used for improving the quality of the found solutions.

#### IV. EXTENDING THE DEFINITION OF $\beta$ -CONSISTENT BICLUSTERING

The constraints (1) guarantee that all the components of the centroid of  $(S_{\hat{r}}, F_{\hat{r}})$  are larger than their homologous components in any other bicluster  $(S_{\hat{r}}, F_{\xi})$ , for any  $\xi \in \{1, 2, \dots, k\}$ , with  $\xi \neq \hat{r}$ . In the case of  $\alpha$ -consistent biclustering (see constraints (2)), this requirement is strengthened by introducing a minimum margin between pairs of homologous components of the centroids. This prevents to have the constraints unsatisfied after small variations in the data, and it leads to the following immediate result:

##### IV.1 Proposition

Any  $\alpha$ -consistent biclustering of  $A[x]$  (see Definition II.2) is also a consistent biclustering of  $A[x]$  (see Definition II.1).

The basic idea behind the  $\beta$ -consistent biclustering is the following. Instead of using an additive parameter  $\alpha$ , the multiplicative parameter  $\beta$  is employed, which must be greater than 1. In this case (see constraints (3)), each component of the centroid of  $(S_{\hat{r}}, F_{\hat{r}})$  must be larger than  $\beta$  times the

value of its homologous component in any other bicluster  $(S_{\hat{r}}, F_{\xi})$ . However, if some of these components are negative, the multiplication by  $\beta$  can give an undesired effect. When this happens, even if the constraints (3) are all satisfied, the original set of constraints (1) may not be satisfied. Therefore, even though the found biclustering may be  $\beta$ -consistent by Definition II.3, it might actually be not even consistent.

The incorrectness of Definition II.3 for  $\beta$ -consistent biclustering is also reflected in the results of the experiments reported in [12], [15]. While the rate of correct classifications constantly increased when  $\alpha$ -consistent biclusterings with larger  $\alpha$  values were searched, this rate had a *strange* behaviour in correspondence with  $\beta$ -consistent biclusterings with larger  $\beta$  values. After a certain threshold for  $\beta$ , the number of correct classifications performed with the found biclustering started to decrease. Since the considered matrix  $A$  contains both positive and negative elements, this phenomenon can most likely be explained by the fact that the found  $\beta$ -consistent biclusterings were actually not consistent.

Consider for example this simple matrix:

$$A = \begin{pmatrix} -2 & 2 \\ -1 & 8 \end{pmatrix}.$$

Suppose that there are two classes of samples and features. Suppose that the two biclusters of the considered biclustering contain, respectively, the element -2 and the element 8 of  $A$ . It is very easy to verify that this biclustering is  $\beta$ -consistent with  $\beta = 3$ , because  $-2$  is greater than  $\beta \times (-1)$ . However, it is not consistent, because  $-2$  is not greater than  $-1$ . This incoherence is due to the fact that  $A$  contains negative entries.

The following is the definition of  $\beta$ -consistent biclustering that extends the one given in [15] to matrices  $A$  containing negative elements.

##### IV.2 Definition

A biclustering for  $A[x]$  is  $\beta$ -consistent if and only if,  $\forall \hat{r}, \xi \in \{1, 2, \dots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$ , the following condition is satisfied:

$$\left\{ \begin{array}{l} \frac{\sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^m f_{i\hat{r}} x_i} > \beta \times \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i} \quad \text{if } c > 0 \\ \frac{\sum_{i=1}^m a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^m f_{i\hat{r}} x_i} > (2 - \beta) \times \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i} \quad \text{if } c < 0 \end{array} \right. \quad (5)$$

where

$$c = \frac{\sum_{i=1}^m a_{ij} f_{i\xi} x_i}{\sum_{i=1}^m f_{i\xi} x_i}$$

and  $\beta > 1$ .

If  $c$  is positive, then Definition II.3 is coherent. If  $c$  is instead negative, its product by  $\beta$  would produce a decrease of its value. Suppose that  $\beta = 1 + \gamma$ , with  $\gamma > 0$ . Then,  $\beta \times c$  can be divided in two parts:  $c$  itself, and  $\gamma \times c$ , which represents the *variation* on the original value of  $c$  obtained by performing the product. Definition IV.2 is able to correct this variation when  $c$  is negative by inverting the sign of  $\gamma \times c$ . Indeed, when  $-\gamma \times c$  is added to  $c$ , the multiplicative factor is actually  $1 - \gamma$ , which corresponds to  $2 - \beta$ . The following theoretical result can now be stated:

### IV.3 Proposition

Any  $\beta$ -consistent biclustering of  $A[x]$  (see Definition IV.2) is also a consistent biclustering of  $A[x]$  (see Definition II.1).

## V. COMPUTATIONAL EXPERIMENTS

The heuristic algorithm discussed in Section III is employed in the following experiments for finding  $\beta$ -consistent biclusterings of sets of data (Definition IV.2) containing a maximal number of selected features. The heuristic has been implemented in AMPL [1], where the ILOG CPLEX11 solver [7] has been used for solving the linear inner problem at each iteration of the VNS-based heuristic. The set of data which is considered in the experiments is related to patients diagnosed with acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML) diseases [4]. All the experiments are carried out on an Intel Core 2 CPU 6400 @ 2.13 GHz with 4GB RAM, running Linux.

The set of data is divided in a training set, which can be used for finding the biclusterings, and a validation set, which can be used for checking the quality of the classifications performed by the exploiting the knowledge acquired by finding the biclusterings (see Section II). The training set contains 38 samples: 27 ALL samples and 11 AML samples. The validation set contains 34 samples: 20 ALL samples and 14 AML samples. The total number of features is 7129. This is a well-known test problem, and experiments on this problem can be found, for example, in [12], [15], where found biclusterings have been employed for performing supervised classifications. In these experiments, the number of misclassifications on the validation set gets higher when the parameter  $\beta$  reaches a certain threshold.

In order to use the heuristic described in Section III without any modification, the training set is scaled in the experiments so that it only contains non-negative data. This practically solves the issue related to negative data, but it does not allow for directly comparing the experiments presented in this paper to those in [12], [15]. In fact, the degree of magnitude of the data changes with scaling, and the values for the parameter  $\beta$  are not comparable anymore. However, the important result which is presented in this paper is that completely correct classifications can be performed with the newly found biclusterings. The  $\beta$  value that allows for obtaining correct classifications in the original scaling of the training set is only a marginal information.

Table I shows the experiments. The total number  $f(x)$  of features that are selected in each experiment is reported,

TABLE I  
COMPUTATIONAL EXPERIMENTS ON A SET OF SAMPLES FROM PATIENTS DIAGNOSED WITH ALL AND AML DISEASES. THESE ARE THE FIRST EXPERIMENTS IN WHICH THE NUMBER OF MISCLASSIFIED SAMPLES IS 0.

$\beta$	$f(x)$	$err$	$mis. samples$
1.001	7011	2	{3,31}
1.002	6984	2	{3,31}
1.003	6946	1	31
1.004	6702	1	31
1.005	5914	1	31
1.006	5072	1	31
1.007	4524	0	-
1.008	3932	0	-
1.009	3443	0	-
1.010	3033	0	-

together with the number  $err$  of misclassifications occurring when the samples of the validation set are classified accordingly with the found  $\beta$ -consistent biclusterings. Moreover, the labels of the misclassified samples (if any) are reported in the last column (each sample is labeled by an integer number between 1 and 34 and by following the ordering in which they are stored in the set of data [4]). Each experiment took no more than 10 minutes of CPU time.

When  $\beta$  is rather small, two samples of the validation set, the one labeled by 3 and the one labeled by 31, are misclassified even though the found biclustering is  $\beta$ -consistent. When  $\beta$  increases, fewer samples are misclassified. In particular, when  $\beta$  is equal to 1.007 (notice that this value is strongly related to the new scaling of the training set), and even for larger values, there are no misclassifications on the validation set when the found  $\beta$ -consistent biclustering is employed for performing the classifications. For the very first time, there are no misclassifications on the validation set when its samples are classified accordingly to  $\beta$ -consistent biclusterings of this training set.

## VI. CONCLUSIONS

This paper extends the definition of  $\beta$ -consistent biclustering previously given in [15]. More accurate classifications can be performed by using this extended definition. The paper presents a theoretical study, which includes a new definition for the  $\beta$ -consistency, as well as an experimental study. A subset of features was found for a well-known classification problem related to gene expression data that allowed for performing classifications with no mistakes.

## ACKNOWLEDGMENTS

I am grateful to Sonia Cafieri for the fruitful discussions.

## REFERENCES

- [1] AMPL, <http://www.ampl.com/>
- [2] S. Busygin, O. A. Prokopyev, P. M. Pardalos, *Feature Selection for Consistent Biclustering via Fractional 0-1 Programming*, Journal of Combinatorial Optimization **10**, 7-21, 2005.
- [3] S. Busygin, O. A. Prokopyev, P. M. Pardalos, *Biclustering in Data Mining*, Computers & Operations Research **35**, 2964-2987, 2008.

- [4] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, E. S. Lander, *Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring*, *Science* **286**, 531–537, 1999.
- [5] P. Hansen and N. Mladenovic. *Variable Neighborhood Search: Principles and Applications*, *European Journal of Operational Research* **130**(3), 449–67, 2001.
- [6] L.-L. Hsiao, F. Dangond, T. Yoshida, R. Hong, R. V. Jensen, J. Misra, W. Dillon, K. F. Lee, K.E. Clark, P. Haverty, Z. Weng, G. L. Mutter, M. P. Frosch, M.E. MacDonald, E. L. Milford, C.P. Crum, R. Bueno, R. E. Pratt, M. Mahadevappa, J. A. Warrington, Gr. Stephanopoulos, Ge. Stephanopoulos, S.R. Gullans, *A Compendium of Gene Expression in Normal Human Tissues*, *Physiological Genomics* **7**, 97-104, 2001.
- [7] ILOG, CPLEX, <http://www.ilog.com/products/cplex/>
- [8] W. Klosgen, J.M. Zytow, *Handbook of Data Mining and Knowledge Discovery*, Oxford University Press, 2002.
- [9] O. E. Kundakcioglu, P. M. Pardalos, *The Complexity of Feature Selection for Consistent Biclustering*, In: *Clustering Challenges in Biological Networks*, S. Butenko, P. M. Pardalos, W. A. Chaovalitwongse (Eds.), World Scientific Publishing, 2009.
- [10] S. C. Madeira and A. L. Oliveira, *Biclustering Algorithms for Biological Data Analysis: a Survey*, *IEEE Transactions on Computational Biology and Bioinformatics* **1** (1), 24–44, 2004.
- [11] M. Mladenovic and P. Hansen, *Variable Neighborhood Search*, *Computers and Operations Research* **24**, 1097–1100, 1997.
- [12] A. Mucherino, S. Cafieri, *A New Heuristic for Feature Selection by Consistent Biclustering*, arXiv e-print, arXiv:1003.3279v1, March 2010.
- [13] A. Mucherino, P. Papajorgji, P. M. Pardalos, *Data Mining in Agriculture*, Springer, 2009.
- [14] A. Mucherino, A. Urtubia, *Consistent Biclustering and Applications to Agriculture*, Ibal Conference Proceedings, Proceedings of the Industrial Conference on Data Mining (ICDM10), Workshop on Data Mining and Agriculture (DMA10), Berlin, Germany, 105-113, 2010.
- [15] A. Nahapatyan, S. Busygin, and P.M. Pardalos, *An Improved Heuristic for Consistent Biclustering Problems*, In: *Mathematical Modelling of Biosystems*, R.P. Mondaini and P.M. Pardalos (Eds.), *Applied Optimization* **102**, Springer, 185–198, 2008.
- [16] D. A. Notterman, U. Alon, A.J. Sierk, A. J. Levine, *Transcriptional Gene Expression Profiles of Colorectal Adenoma, Adenocarcinoma, and Normal Tissue Examined by Oligonucleotide Arrays*, *Cancer Research* **61**, 3124-3130, 2001.
- [17] A. Urtubia, J. R. Perez-Correa, A. Soto, P. Pszczolkowski, *Using Data Mining Techniques to Predict Industrial Wine Problem Fermentations*, *Food Control* **18**, 1512–1517, 2007.



# International Workshop on Advances in Business ICT

**A**BICT focuses on Advances in Business ICT approached from a multidisciplinary perspective. It will provide an international forum for scientists/experts from academia and industry to discuss and exchange current results, applications, new ideas of ongoing research and experience on all aspects of Business Intelligence.

We kindly invite contributions originating from any area of computer science, information technology and computational solutions for different applications areas, data integration and organizational implementation of ABICT, as well as practical ABICT solutions.

Topics include (but are not limited to):

- Advanced Technologies of Data Processing, Content Processing and Information Indexing
- Business Applications of Social Networks
- Business Data Mining and Knowledge Discovery
- Business Intelligence, Business Analytics
- Business Rules
- Business-oriented Time Series Data Mining, Analysis, and Processing
- Data Warehousing
- Information Forensics and Security, Information Management, Risk Assessment and Analysis
- Information Systems in Enterprise Management
- Information Technologies in Enterprise Logistics
- Information Technologies in Enterprise Management, Information Systems,
- Service Oriented Architectures (SOA)
- Knowledge Management
- Recommender Systems
- Semantic Web and Ontologies in Business ICT
- Virtual Enterprise
- Web-Based Data Management Systems

## PROGRAM COMMITTEE

**Rainer Alt**, University of Leipzig, Germany

**Amelia Badica**, University of Craiova, Romania

**Giuseppe Berio**, Universite de Bretagne Sud, France

**Witold Bielecki**, Kozminski University, Poland

**Witold Byrski**, AGH-University of Science and Technology, Poland

**Gerardo Canfora**, University of Sannio, Italy

**Miriam Capretz**, University of Western Ontario, Canada

**Dickson K. W. Chiu**, Dickson Computer Systems, Hong Kong

**Flavio Corradini**, University of Camerino, Italy

**Petr Dostal**, Brno University of Technology, Czech Republic

**Marek Druzdzel**, University of Pittsburgh, Biaystok Technical University, United States

**Jan T. Duda**, AGH-University of Science and Technology, Poland

**Ewa Dudek-Dyduch**, AGH-University of Science and Technology, Poland

**Schahram Dustdar**, Vienna University of Technology, Austria

**Yamna Ettarres**, Virtual University of Tunis, Tunisia

**Bogdan Franczyk**, University of Leipzig, Germany

**Jozef Goetz**, University of La Verne, United States

**Adam Grzech**, Wrocław University of Technology, Poland

**Ryszard Janicki**, McMaster University, Canada

**Stanisław Jarzabek**, National University of Singapore, Singapore

**Joanna Józefowska**, Poznań University of Technology, Poland

**Janusz Kacprzyk**, Institute of Computer Science, Polish Academy of Sciences, Poland

**Pawel J. Kalczynski**, California State University, United States

**Waldemar Koczkodaj**, Laurentian University, Canada

**Mieczysław Kokar**, Northeastern University, United States

**Beata Konikowska**, Institute of Computer Science, Poland

**Michael L. Korwin-Pawlowski**, Universite du Quebec en Outaouais, Canada

**Piotr Kulczycki**, Systems Research Institute, Polish Academy of Sciences, Poland

**Maurizio Lenzerini**, Sapienza Università di Roma, Italy

**Antoni Ligęza**, AGH-University of Science and Technology, Poland

**Peri Loucopoulos**, Loughborough University, United Kingdom

**Lech Madeyski**, Wrocław University of Technology, Poland

**Yannis Manolopoulos**, Aristotle University of Thessaloniki, Greece

**Mieczysław Muraszkievicz**, Warsaw University of Technology, Poland

**Markus Nuettgens**, University of Hamburg, Germany

**Andreas Oberweis**, Karlsruhe Institute of Technology (KIT), Germany

**Mitsunori Ogihara**, University of Miami, United States

**Vassilios Peristeras**, National University of Ireland, Ireland

**Lyubomyr Petryshyn**, AGH-University of Science and Technology, Poland

**Carlos Andre Reis Pinheiro**, Dublin City University, Ireland

**T. V. Prasad**, Lingaya's University, India

**Elke Pulvermueller**, University Osnabrueck, Germany

**Ulrich Reimer**, University of Applied Sciences St. Gallen, Switzerland

**Gustavo Rossi**, National University of La Plata, Argentina

**Massimo Ruffolo**, University of Calabria, Italy

**Jurgen Sauer**, University of Oldenburg, Germany

**Douglas C. Schmidt**, Vanderbilt University, United States

**Iwona Skalna**, AGH-University of Science and Technology, Poland

**Andrzej Sluzek**, Nanyang Technological University, Singapore

**Marcin Szpyrka**, AGH University of Science and Technology, Poland

**Ryszard Tadeusiewicz**, AGH University of Science and Technology, Poland

**Stephanie Teufel**, University of Fribourg, Switzerland

**Hans Weigand**, Tilburg University, Netherlands

**Stanisław Wrycza**, University of Gdansk, Poland

**Sławomir Zadrozny**, Polish Academy of Sciences, Poland

**John Zeleznikow**, Victoria University, Australia

**Jerzy S. Zieliński**, University of Łódź, Poland

#### ORGANIZING COMMITTEE

**Mieczysław L. Owoc**, Wrocław University of Economics, Poland

**Maria Antonina Mach** (Chairperson), Wrocław University of Economics, Poland [maria.mach@ue.wroc.pl](mailto:maria.mach@ue.wroc.pl)

**Tomasz Pelech-Pilichowski**, AGH-University of Science and Technology, Poland [tomek@agh.edu.pl](mailto:tomek@agh.edu.pl)

# Formal Verification of Business Processes as Role Activity Diagrams

Amelia Bădică\* and Costin Bădică†

\*University of Craiova, Romania, Email: ameliabd@yahoo.com

†University of Craiova, Romania, Email: costin.badica@software.ucv.ro

**Abstract**—Business process modeling is performed during the requirements analysis and specification of business software systems. Checking qualitative aspects of business processes is required for quality assurance, as well as for compliance with non-functional requirements. We show how business process models represented as Role Activity Diagrams can be formally checked using process algebras and temporal logics.

## I. INTRODUCTION

MODERN organizations are process-centered and the truth of the statement “process precedes information” [1] is now widely recognized. This explains why most of the frameworks for organizational modeling and design emphasize the role of processes and process modeling from the high level description of the organization to the underlying IT support systems. Nevertheless, most often, the mapping of business process models to IT applications is defined in an ad-hoc way and the support for managing this mapping is poor. Even if the importance of a separate modeling stage was recognized and a preliminary business process modeling activity is carried out, the resulting model does not have a formal computational semantics and thus it is difficult to map it onto an IT language. On the other hand, the language spoken by IT specialists is too technical, so it is hard to link it up to higher level goals of the organization stated by business analysts.

Consider for example the requirements analysis for an IT system supporting a process-centered business organization. The set of resulting requirements usually contains functional, non-functional, as well as domain specific requirements. According to [2] (i) non-functional requirements may be very hard to verify, as the customers describe them using high-level and informal goals that usually apply to the system or process as a whole, rather than to a specific functionality, and (ii) domain requirements are usually very difficult to understand and specify by non-experts of the application domain. Recent works proposed methods to link non-functional requirements to business process models via goal analysis [3], although the support for automated verification is lacking.

Second, during the last decade, several formal approaches for business process modeling and design were proposed, relying on sound logical approaches. We classified them into (i) lightweight formalizations that use classic first-order logics [4], (ii) heavyweight formalizations based on temporal logics [5], and (iii) hybrid approaches that combine first-order and more advanced (e.g. dynamic or temporal) logics [6].

Third, software engineering provides sound formal modeling and verification techniques for the modeling and verification of software specifications. These methods are now well-supported by model checking technologies [5] and their practical application was improved by introduction of property verification patterns that can be expressed using temporal logics [7]. Recently, the application of formal methods was extended to business processes [8], [9]. A significant step ahead was achieved by introduction of property verification patterns for business process models [10], recently enhanced with visual notations [11], [12]. In particular, model checking can be applied for quality assurance of business processes [13], [9].

We conclude that business process modeling is necessary during the requirements analysis and specification of a business software system for a process-centered organization. Checking qualitative aspects of business processes is required for quality assurance, as well as for compliance with non-functional or domain specific requirements. However, although the software technology that could help to automate this verification exists, the main difficulty is the semantic gap between the languages “spoken” by the business analysts and the IT people. While business analysts are using high-level diagrammatic notations with an intuitive meaning and closer to the business world, IT people are using low-level computational languages that are closer to the computing world. We claim that the missing bit that hinders their clean interaction is the lack of, broadly understood, formal rigor of the current notations used by both business and IT communities.

We propose our contribution for bridging this gap by focusing on formal modeling and verification of business processes represented as Role Activity Diagrams (RAD) [1]. We are aware of the large variety of modeling languages that were proposed for the business and software engineering communities, including the most recent Business Process Modeling Notation (BPMN)<sup>1</sup>. Our chose RAD because: (i) RAD was used for modeling requirements of business processes [12], [14], [15]; (ii) RAD has formal semantics that we introduced in [8]; thus the extension of this work to formal verification is quite natural and straightforward; moreover RAD is at the core of the formalism for knowledge-based modeling of organizations proposed in [6]; (iii) RAD is intuitive and more appropriate for business analysts [1]; (iv) while the focus of other process

<sup>1</sup><http://www.bpmn.org/>

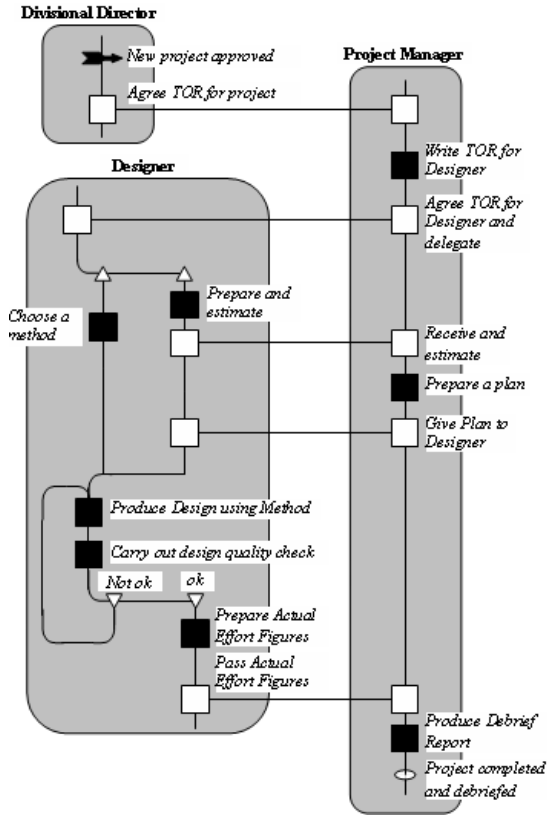


Fig. 1: Sample RAD model for a design project

notations, like UML activity diagrams and BPMN is set on the technical aspects including standardization, interoperability, and integration with software modeling languages, the focus of RAD is set on addressing the high-level needs of a process modeler from business rather than technical perspective.

## II. AN EXAMPLE OF FORMAL VERIFICATION

RAD is a visual notation for business process modeling [1]. We consider the Finite State Process algebra (FSP) model of a RAD process and show how it can be checked against qualitative properties represented using temporal logics. FSP is introduced in [16] and the mapping of RAD to FSP is presented in [8].

**Sample Formal Model.** Let us consider the RAD model of the design process shown in Figure 1. Its FSP model is shown below. Note that the meaning of the actions used in this FSP model is given in Table I.

$$\begin{aligned} & \|DD_0 = SS/\{npa/o\}. \\ & \|DD_1 = L/\{npa/i, atp/o\}. \\ & \|DD_2 = E/\{atp/i\}. \\ & \|DivisionalDirector = (DD_0 \parallel DD_1 \parallel DD_2). \end{aligned}$$

$$\begin{aligned} & \|D_0 = SS/\{atdd/o\}. \\ & \|D_1 = L/\{atdd/i, dp/o\}. \\ & \|D_2 = L/\{dp/i, cm/o\}. \\ & \|D_3 = L/\{dp/i, pe/o\}. \\ & \|D_4 = L/\{cm/i, ds/o\}. \\ & \|D_5 = L/\{pe/i, re/o\}. \\ & \|D_6 = L/\{re/i, gpd/o\}. \\ & \|D_7 = L/\{gpd/i, ds/o\}. \\ & \|D_8 = L/\{ds/i, pdm/o\}. \\ & \|D_9 = L/\{\{nok, pdm\}/i, cdqc/o\}. \\ & \|D_{10} = L/\{cdqc/i, \{nok, ok\}/o\}. \\ & \|D_{11} = L/\{ok/i, paef/o\}. \\ & \|D_{12} = L/\{paef/i, psaeef/o\}. \\ & \|D_{13} = E/\{psaeef/i\}. \\ & \|Designer = (D_0 \parallel D_1 \parallel D_2 \parallel \dots \parallel D_{13}). \\ & \|PM_0 = SS/\{atp/o\}. \\ & \|PM_1 = L/\{atp/i, wtd/o\}. \\ & \|PM_2 = L/\{wtd/i, atdd/o\}. \\ & \|PM_3 = L/\{atdd/i, re/o\}. \\ & \|PM_4 = L/\{re/i, pp/o\}. \\ & \|PM_5 = L/\{pp/i, gpd/o\}. \\ & \|PM_6 = L/\{gpd/i, psaeef/o\}. \\ & \|PM_7 = L/\{psaeef/i, pdr/o\}. \\ & \|PM_8 = E/\{pdr/i\}. \\ & \|ProjectManager = (PM_0 \parallel PM_1 \parallel PM_2 \parallel \dots \parallel PM_8). \\ & \|System = (DivisionalDirector \parallel Designer \parallel ProjectManager). \end{aligned}$$

**Sample Verification.** Formal modeling of business processes has the advantage that models can be systematically checked against user-defined properties. A property is defined by a statement that should be true for all the possible execution paths of the process. A property is used to describe a desirable feature of the system behavior. Formal definition of business process properties has the advantage that it enables their concise, rather than speculative analysis.

Properties of software systems are expressed as temporal logic formulas [5]. Temporal logics are used for declarative specification of properties of dynamic systems defined as labeled transition systems, including business processes. A property holds if the associated formula is true for all the possible executions of the system, as it is described by the system model. For system models captured using FSP it was shown that a convenient logic for property specification is *fluent linear temporal logic* (FLTL) [16].

In FLTL primitive properties are expressed using *fluents*. A fluent is a property whose truth is triggered by an initiating event and that holds until the signalling of a terminating event. In FSP it is natural to model initiating and terminating events by execution of specific actions. Every action  $a$  defines a singleton fluent  $F(a)$  having  $a$  as the single initiating action and the rest of all actions as terminating actions. A singleton fluent  $F(a)$  is usually written as  $a$  in FLTL formulas.

FLTL formulas are built over fluent propositions using the logical operators  $\wedge, \vee, \rightarrow, \neg$  and temporal operators **X** (next), **U** (until), **W** (weak until), **F** (eventually) and **G** (always) [16]. A property  $P$  is specified using an FLTL formula  $\Phi$ .

At the core of a verification task is the activity of property specification. This activity is recognized as very difficult, because on one side it requires special skills in formal specification using temporal logics, while on the other side it requires a good understanding of the target application domain. Nevertheless, some steps have been made in order to help the human modeler to produce specifications of properties by the introduction of specification patterns [7], [10], [11]. However,

TABLE I: Mapping of RAD elements onto FSP actions.

Action	RAD entity name	RAD entity type	Role
<i>npa</i>	<i>New project approved</i>	External event	<i>Divisional Director</i>
<i>atp</i>	<i>Agree TOR for project</i>	Interaction	<i>Divisional Director, Project Manager</i>
<i>wtd</i>	<i>Write TOR for Designer</i>	Activity	<i>Project Manager</i>
<i>atdd</i>	<i>Agree TOR for Designer and delegate</i>	Interaction	<i>Designer, Project Manager</i>
<i>dp</i>	<i>n/a</i>	Part splitting	<i>Designer</i>
<i>cm</i>	<i>Choose a method</i>	Activity	<i>Designer</i>
<i>pe</i>	<i>Prepare and estimate</i>	Activity	<i>Designer</i>
<i>re</i>	<i>Receive and estimate</i>	Interaction	<i>Designer, Project Manager</i>
<i>ds</i>	<i>n/a</i>	Parts synchronization point	<i>Designer</i>
<i>gpd</i>	<i>Give Plan to Designer</i>	Interaction	<i>Designer, Project Manager</i>
<i>pdm</i>	<i>Produce Design using Method</i>	Activity	<i>Designer</i>
<i>cdqc</i>	<i>Carry out design quality check</i>	Activity	<i>Designer</i>
<i>pacf</i>	<i>Prepare Actual Effort Figures</i>	Activity	<i>Designer</i>
<i>psacf</i>	<i>Pass Actual Effort Figures</i>	Interaction	<i>Designer, Project Manager</i>
<i>pp</i>	<i>Prepare a plan</i>	Activity	<i>Designer</i>
<i>gpd</i>	<i>Give Plan to Designer</i>	Interaction	<i>Designer, Project Manager</i>
<i>nok</i>	<i>Not ok</i>	Case refinement	<i>Designer</i>
<i>ok</i>	<i>Ok</i>	Case refinement	<i>Designer</i>
<i>pcd</i>	<i>Project completed and debriefed</i>	State description	<i>Project Manager</i>

in our opinion even with the availability of visual specification patterns [12] we are still far from bridging the gap between requirements analysis and verification, as pointed out in the introduction of this paper. Nevertheless, we found very useful the use of patterns to specify simple properties for the process example considered in this paper.

In our case we considered two sample properties:

- P1 Each project approved must be eventually completed and debriefed.
- P2 Each project cannot be completed and debriefed without being approved by a design quality check.

Based on a rigorous analysis of the business domain, [10] proposed four classes of property specification patterns for business process models; tracing, consequence, combined occurrence, and precedence. Those two properties were formalized using consequence and precedence patterns by reformulating them as follows:

- P1 The action “New project approved” leads to reaching the state “Project completed and debriefed”.
- P2 The state “Project completed and debriefed” requires the action “Carry out design quality check” to return a positive response.

Their formal description using FLTL is as follows:

$$\begin{aligned} \text{assert } P1 &= \mathbf{G} (npa \rightarrow \mathbf{F} pcd) \\ \text{assert } P2 &= ((\neg pcd \mathbf{U} ok) \vee \mathbf{G} \neg pcd) \end{aligned}$$

We have checked the sample process model against these two properties with the help of Labelled Transition System Analyser (version 3.0)<sup>2</sup>. Both P1 and P2 were found as not violated by the process model. However, the analysis of P2 revealed a deadlock that can be explained as follows. A property check assumes three steps: (i) construction of a new process corresponding to the given property; (ii) computation of the parallel composition of the original process with the property process; (iii) performing a graph analysis of the resulting parallel composition. In our case, the resulting parallel composition has a deadlock because the property P2 does not

explicitly check that action *pcd* actually occurs, while in the original process this action will always eventually occur.

This simple experiment revealed a number of difficulties with the application of formal specification to requirements verification of business process models.

- The formal business process model is very difficult to understand and maintain. Without a proper management of the links between the formal model and the initial RAD model, it is impossible to manage large and complex process models.
- The formalization of requirements is very difficult to realize, even with the availability of property specification patterns. Additional support, beyond patterns, is needed to better manage the requirements and their mapping to formal properties.
- The results of the verification process are very difficult to interpret. Special support to link them back to the RAD model, as well as to explain them to the human modeler is lacking.

### III. CONCLUSIONS

We introduced a method for checking qualitative properties of business processes represented as RADs using the formal specification languages of process algebras and temporal logics. We presented our initial analysis of the problems encountered during the application of formal verification for checking requirements of business processes. In our opinion the core technologies already exist. However, the necessary links between them are lacking. We think that the main cause is the gap between the languages spoken by the business and computing communities. We plan to address this issue as medium term future work. In the short term we plan to enhance our results by considering more complex processes and properties.

<sup>2</sup><http://www.doc.ic.ac.uk/~jnm/book/ltsa/download.html>

## ACKNOWLEDGMENT

The work reported here was partly supported by (i) the research project “SCIPA: Servicii software semantice de Colaborare si Interoperabilitate pentru realizarea Proceselor Adaptive de business” with the National Authority for Scientific Research, Romania and partly by (ii) the research project “Agent-Based Service Negotiation in Computational Grids” between Systems Research Institute, Polish Academy of Sciences, Poland and Software Engineering Department, University of Craiova, Romania.

## REFERENCES

- [1] M. A. Ould, *Business Process Management: A Rigorous Approach*. British Computer Society, 2005.
- [2] I. Sommerville, *Software Engineering, 9/E*. Addison-Wesley, 2011.
- [3] F. Aburub, M. Odeh, and I. Beeson, “Modelling non-functional requirements of business processes,” *Information and Software Technologies*, vol. 49, no. 11-12, pp. 1162–1171, 2007.
- [4] Y.-H. Chen-Burger and D. Robertson, *Automating Business Modelling*. Springer, 2004.
- [5] E. M. Clarke, O. Grumberg, and D. A. Peled, *Model Checking*. The MIT Press, 1999.
- [6] M. Koubarakis and D. Plexousakis, “A formal framework for business process modelling and design,” *Information Systems*, vol. 27, no. 5, pp. 299–319, 2002.
- [7] M. B. Dwyer, G. S. Avrunin, and J. C. Corbett, “Patterns in property specifications for finite-state verification,” in *Proc. 21<sup>st</sup> international conference on Software engineering (ICSE 1999)*. IEEE Computer Society Press, 1999, pp. 411–420.
- [8] A. Bădică, C. Bădică, and V. Lițoiu, “Role activity diagrams as finite state processes,” in *Proc. 2<sup>nd</sup> International Symposium on Parallel and Distributed Computing (ISPDC 2003)*. IEEE Computer Society, 2003, pp. 15–22.
- [9] B. B. Anderson, J. V. Hansen, P. B. Lowry, and S. L. Summers, “Model checking for e-business control and assurance,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 35, pp. 445–450, 2005.
- [10] W. Janssen, R. Mateescu, S. Mauw, P. Fennema, and P. v. d. Stappen, “Model checking for managers,” in *Proceedings of the 5th and 6th International SPIN Workshops on Theoretical and Practical Aspects of SPIN Model Checking*, ser. Lecture Notes in Computer Science, vol. 1680. Springer-Verlag, 1999, pp. 92–107.
- [11] A. Forster, G. Engels, T. Schattkowsky, and R. Van Der Straeten, “Verification of business process quality constraints based on visual process patterns,” in *Proceedings of the First Joint IEEE/IFIP Symposium on Theoretical Aspects of Software Engineering*. IEEE Computer Society, 2007, pp. 197–208.
- [12] A. Awad, M. Weidlich, and M. Weske, “Visually specifying compliance rules and explaining their violations for business processes,” *Journal of Visual Languages and Computing*, vol. 22, no. 1, pp. 30–55, 2011.
- [13] W. Wang, Z. Hidvégi, A. D. Bailey, and A. B. Whinston, “E-process design and assurance using model checking,” *Computer*, vol. 33, no. 10, pp. 48–53, 2000.
- [14] N. V. Patel, “Healthcare modelling through role activity diagrams for process-based information systems development,” *Requirements Engineering*, vol. 5, no. 2, pp. 83–92, 2000.
- [15] S. Bleistein, K. Cox, J. Verner, and K. Phalp, “Requirements engineering for e-business advantage,” *Requirements Engineering*, vol. 11, no. 1, pp. 4–16, 2006.
- [16] J. Magee and J. Kramer, *Concurrency. State Models and Java Programs, 2/E*. John Wiley & Sons, 2006.

# Virtualization as an approach in the development of IT system implementation process

Iwona Chomiak-Orsa, Wiesława Gryniewicz, Maja Leszczyńska  
University of Economics ul. Komandorska 118/120, 53-345 Wrocław, Poland  
Email: {iwona.chomiak, wieslawa.gryniewicz, maja.leszczyńska}@ue.wroc.pl

**Abstract**—Virtual administration of IT system implementation processes is now possible in small and micro-companies, characterized by relative simplicity and marked recurrence of business processes. Popularity of such approach to implementation is largely due to the wide availability of IT solutions offering remote administration of authorized IT resources. Virtual form of implementation offers significant reduction of both cost and time, compared with traditional approach. Consequently, it seems reasonable to expect further development of this trend, addressing larger economic entities and servicing more complex IT systems.

## I. INTRODUCTION

COMPANIES competing on modern markets face increased dynamics of changes, fierce competition and the need of making fast decisions. To meet those challenges, it is necessary to make good use of the available Information Technology (IT) solutions. IT instruments open up new potential for company operation and are a driving force of transformation. By implementing modern technological solutions, companies re-organize their activities not only in the B&C context, but also in relation to other companies, through value-creation chain [1]. The potential offered in this respect by the Internet and networking is widely and readily employed for optimization of business processes, namely the minimization of process cost and maximization of profit. Processes conducted via information and communications technologies (most notably, the Internet), due to the nature of the medium, are subject to potential virtualization. Through virtualization, both the entrepreneurs and their customers can profit from delocalization of business processes, i.e. freeing them up from the geographic constraints and focusing on customer needs and key competences of process supervisors [2].

The potential of virtualization can be readily deduced from the etymology of the term. Virtualization is a word derived from Latin *virtus*, *virtuti* standing for proficiency, efficiency, courage, fortitude and *virtualis* – effective [3]. The aim of this paper is to offer an insight into potential of virtualization of business processes through the evaluation of the IT system implementation process. Determinants, possibilities and tools for virtualization of this process are analyzed. Definition of system implementation process virtualization is presented, together with benefits implied by the use of this method. Deliberations presented in this paper refer to the practice of IT system implementation, with main focus on

the recent trend to virtualize the co-operation between the provider and the client and employ remote implementation procedures and processes based on large potential of modern IT solutions – a trend observed and not yet sufficiently addressed in professional literature.

## II. DETERMINANTS OF VIRTUALIZATION OF THE IT SYSTEM IMPLEMENTATION PROCESS

At present, straight majority of companies, especially large and medium-sized, employ consolidated application suites to service the main areas of their business activities, mainly in the sphere of accounting, personnel and wages, and sales [4]. Due to good saturation of IT solutions in this area, software providers seek to extend their offer to cover the sector of small and micro-companies. This interest takes the form of adapting the IT solutions to the specificity and needs of this particular segment as well as development of trade-specific IT instruments, evident even on the part of the largest software providers. Moreover, IT providers offer a range of supplementary services (business consultancy, support in the acquisition of EU structural funds), and their applications, when properly implemented, warrant increased effectiveness of business processes. This trend is manifested in the increased interest of IT providers in commercialization of their knowledge and implementation expertise, by offering services that complement IT systems' functionality, such as best practices in organization of specialized business processes, process maps for trade-specific activities and supplying predefined sets of procedures for their IT systems. This range of services is particularly attractive for small companies seeking to improve their market standing by implementing IT solutions combined with extensive knowledge of best practices in their line of trade [4].

However, solutions tailored to the needs and requirements of small and micro-companies, from the viewpoint of IT providers, offer significantly lower per-unit profit. Financial resources available to small and micro-companies for IT system implementation (need analysis, product modification and configuration, training, assistance, etc.) are decidedly sub-par compared to those of larger companies. Moreover, their business processes are of significantly lower complexity, coupled with trade-specific large-scale recurrence of procedures. IT systems addressed to this sector are definitely cheaper. Consequently, for optimization of business activi-

ties in this sector to be profitable, the IT providers need to minimize the cost and maximize the number of recipients. Cost minimization requires, on the one hand, standardization and simplification of system functionality and, on the other hand, simplification and time-effectiveness of implementation procedures.

The search for cost-minimization of implementation procedures has led to the present market trend, observed in this sector. In general, the trend is manifested in virtualization of implementation procedures or their constituents. This approach allows for delocalization of implementation, regardless of geographic location of both parties – in practice, the whole implementation process can be accomplished remotely, with significant reduction of implementation time. In addition, the IT system provider can service a large group of clients in a relatively short time, and at marginal cost.

Virtualization of IT system implementation process can be defined as remote realization of individual implementation procedures using modern telecommunications and IT solutions as well as the Internet potential (detailed analysis below). Thus, the need of personal contact between IT provider and the customer is effectively eliminated. The consultant, using a set of IT instruments and the Internet connection, communicates in real time with the customer. Both parties not only communicate with each other, but are also linked with the same physical machine, sharing the desktop view on their respective monitors and working hand in hand. It must be noted that this particular form of cooperation is made possible by the user-friendly, intuitive applications tailored to the requirements of virtual IT system implementation. Those instruments typically do not require specialized IT knowledge; the only requirement is the efficiency of communication via the Internet.

It should be noted that virtualization of IT system implementation processes is available and applicable mainly to small companies operating within a standard set of business activities or those that seek to modify their operation in accordance with best practices of the trade. IT systems dedicated to this sector are characterized by simple, straightforward functionality, intuitive and user-friendly interface and a range of predefined, typically trade-specific standard business processes. End users of such systems, for the majority of tasks, adapt their processes to the knowledge and expertise provided within the system, rather than vice versa. This is in clear opposition to the practice of system implementations in medium and large companies, where IT systems are often adapted to particular requirements of the user, resulting in costly modifications.

The interest in virtualization of implementation processes on the part of small and micro-companies stems from their need to minimize the cost involved. In this respect, both customers and system providers are motivated by similar aims. Moreover, the typically low cost of such projects allow the client to take the risk of virtualization and accept the lack of direct contact with the provider. Such constraints would be considered unacceptable in the case of large projects.

It must be noted that virtualization, among other things, is aimed at building the implementation process on the particular needs of the client. In addition, the client is an active participant of the process, taking part in realization of individual implementation tasks. Consequently, virtualization of IT system implementation processes obliges the provider to offer a suitable level of security and work comfort to the client. This should be reflected in proper organization of implementation tasks and – most of all – a reliable hot-line service offering fast and effective solutions for most of the client's problems and inquiries. In virtual context, this is a substitute for direct contact between clients and consultants.

It must also be remembered that virtualization of implementation processes for standard IT products dedicated to small and micro-companies carries a large potential for development, since the procedures and methodologies developed in the course of system implementation may help streamline future implementation processes targeted to medium and large companies.

### III. POTENTIAL FOR VIRTUALIZATION OF IT SYSTEM IMPLEMENTATIONS

As already mentioned above, the owners of small and micro-companies typically purchase trade-specific or highly standardized IT solutions. Such decisions result from the lack of funds to carry out pre-implementation analyses, offering detailed evaluation of organizational needs, information needs of individual user groups and elaborate design of business processes. In such cases, the implementation becomes a key stage in the system life-cycle. In the sector of small and micro-companies, the awareness of the need to implement IT solutions is a first step in the implementation process. Identification of user needs is carried out from "within" the organization itself – typically through involving company employees and owners in identification and evaluation of operational areas that may benefit from IT support.

The self-induced awareness typically leads the potential users to independently penetrate the market of trade-specific IT products. Users seek products that are not only adequate to their needs, but – most of all – products that place within the reasonable price range. After initial selection, the potential clients contact individual IT system developers or distributors. Nowadays, such contact is accomplished via predefined, interactive contact forms made available on IT providers' web sites. This contact constitutes the first stage of remote client-provider communication. More often than not, the interactive forms include questions that offer initial verification of the client's expectations towards the system's functionality.

The classic approach to implementation process identifies the following sequence of activities [5]:

- preparatory proceedings – involving analysis and preparing the way for the organization to adopt the system, preparation of the system itself and ensuring proper technical infrastructure for future use of the system,



- testing the system – involving trial runs and elimination of errors,
- system exploitation.

In traditional approach, preparatory proceedings required frequent on-site visits, with consultants preparing specifications of user requirements in terms of system functionality [6]. In modern approach, the initial evaluation of user requirements is verified via specific questions included in the interactive contact form. Hence, virtualization of this stage of system implementation process, in the case of small and micro-companies, may limit the number of direct contacts to only one pre-implementation meeting, to clarify user expectations and settle the financial conditions of the contract. In the case of small and micro-companies, preparing the organization for adopting the system is typically reduced to appointing the client representative to supervise the implementation procedures and remotely co-operate with the consultant representing the IT system provider.

Since the IT market at present offers a large number of IT solutions for remote automation of the installation process, the prospective user may choose to open the technical resources of the company to the IT provider and have the system installed remotely. The IT instruments also offer trial run assistance as well as verification of data structure and correctness of implemented algorithms. The functionality of selected IT instruments offering virtualization of system implementation process is presented in the next section of this paper.

One of the key stages of IT system implementation is employee training. This area can also benefit from remote cooperation between system provider and end user. Modern software is typically equipped with elaborate help modules and detailed user manuals with detailed presentation of system functionality features. The increased focus on user self-improvement during standard system operation effectively reduces the time needed to familiarize the user with the system. The end users (employees) effectively take over parts of the implementation procedures but, at the same time, are made responsible for the progress [7]. By limiting or eliminating the number of training sessions supervised by IT provider, the company can largely reduce the cost of system implementation.

Moreover, virtualization of IT system implementation offers the prospect of remote assistance. Since the end user cannot benefit from direct contact with the IT consultant, the latter is often equipped with remote desktop instruments to better support the user during the initial trial runs and help eliminate errors and problems. The on-line consultants can also remotely address any errors found in data structure or business algorithms implemented in the system.

#### IV. SELECTED IT TOOLS OFFERING VIRTUALIZATION OF IMPLEMENTATION PROCESS

Virtual accomplishment of the aforementioned stages of IT system implementation is made possible by the dynamic development of information and communications technolo-

gies. At present, the market of IT products features a large number of solutions for remote administration of shared computers. Those applications vary in terms of operating system support, functionality features, ease of use, built-in security level and licensing fees.

Companies intent on using virtual approach in implementing new software need only satisfy the requirement of leased line Internet connection, with bandwidth playing a major part in the efficiency and facility of remote cooperation. The Internet in this process is perceived as global channel, providing real-time information exchange between the parties [Jurga 2010, pp. 49-53]. There are also dedicated IT solutions for remote administration via LAN and WAN (local and wide area networks), but – since software providers are outside the reach of such networks, the Internet remains a fundamental communication medium.

The most popular applications used for the purpose of virtual implementation are presented in the table 1. They were divided into two groups, depending on the type of license: freeware and commercial software. Many of them are available for use at no cost or for an optional fee. All of the commercial solutions have also trial versions, which can be used for free for a predetermined time.

TABLE 1.  
THE MOST POPULAR SOFTWARE USED BY IT PROVIDERS FOR THE PURPOSE OF VIRTUAL IMPLEMENTATION

Type of license	Software
Freeware	<ul style="list-style-type: none"> <li>• TeamViewer<sup>1</sup></li> <li>• CrossLoop</li> <li>• TightVNC</li> <li>• Remote Desktop Connection</li> </ul>
Commercial	<ul style="list-style-type: none"> <li>• pcAnywhere</li> <li>• NetOp Remote Control</li> <li>• Radmin</li> <li>• Atelier Web Remote Commander</li> <li>• YuuGuu</li> </ul>

TeamViewer offers facilitated connectivity without the need of installing client/server applications on the remote machine. The provider needs access to full version of Team Viewer application, but the customer needs only to install a Team Viewer QuickSupport module. The client-side module is user-friendly and does not require advanced skills nor knowledge. Moreover, TeamViewer allows for generation of customized client modules, with provider's logo and welcome message, thus offering optimal presentation of contact details. In the case of non-commercial use Team Viewer is free [<http://www.teamviewer.com>].

Another example of remote administration software based on the Internet connection is CrossLoop. The package offers access and/or administration of remote computers, with the administrator having unrestrained view of the remote desktop with mouse and keyboard functionality. The CrossLoop

<sup>1</sup> It is free only for non-commercial use

package is an ideal tool for specialized technical assistance services during system implementation. It is available for free, but requires a certain degree of IT knowledge to operate. There is also CrossLoop Pro – the commercial version of this application with more complex functionality [<http://www.crossloop.com>].

TightVNC is another free remote control package based on the VNC software. It allows the user to take over remote desktop functionality. Compared with RealVNC (based on similar code), TightVNC offers improved image compression, which helps perform standard operations on remote systems with low-bandwidth connection. Similarly to RealVNC, the package includes two modules: Server (image generation and transfer) and Viewer (image reception) [<http://www.tightvnc.com>].

Remote Desktop Connection, a client software for remote administration offered by Microsoft, is less popular due to limited functionality, such as the lack of shared control (mouse, keyboard, desktop) and OS restrictions (the software offers connectivity with systems working in Windows Server 2003 or Windows XP Professional environment).

Symantec's pcAnywhere package is the market leader of remote administration software. By using efficient data encryption and authentication mechanisms, the software offers a high degree of security during remote access sessions. The most recent version of pcAnywhere offers improved directory search capabilities and AutoTransfer for automated batch transfers of files [<http://www.symantec.com/business/pcanywhere>].

NetOp is a family of software products for remote computer administration, with cross-platform capabilities, i.e. the potential to administer computers working under different operating systems. From the remote location, the provider can control workstations and servers, with remote desktop access, keyboard and mouse control, chat functionality (both text and audio), bi-directional file transfer, session recording, etc. NetOp Remote Control offers administration of any remote corporate network, with support for over 20 various operating systems at minimal system resource load. This is the only remote administration package with a centralized security system. This means that the user can control not only access authorization, but also set individual authorization rights for any remote operation – all from a single location. The package includes two basic modules: Host – for sharing computer resources; and Guest – for remote connectivity. Similarly to Time Viewer QuickSupport, the Guest module is easy to install and very user-friendly [<http://www.netop.pl>].

Other packages offering fast, reliable and secure administration of remote systems include Radmin, Atelier Web Remote Commander and Yuuguu.

Radmin is one of the safest, fastest and most popular remote access software solutions designed for Windows. A remote computer screen can be viewed on a local monitor in either a window or a full-screen display. All mouse movements and keystrokes are transferred directly to the remote

computer. Files can be transferred to and from the remote computer, and communication with the remote computer's user is possible by either Text Chat or Voice Chat [<http://www.radmin.com/products/radmin>].

Atelier Web Remote Commander lets manage servers and workstations from local computer and does not require to install any software on the remote machine. This turns the software particularly useful for accessing remote computer without any previous preparation. This application provides lots of powerful tools for remote management and audit [<http://www.atelierweb.com/rcomm/index.htm>].

Yuuguu is the simple to use service for screen sharing, remote support and collaboration. It is a little different than the options outlined above, because it allows to share screen via instant messenger. It also supports all of the major communication platforms such as Yahoo, MSN, AIM, GTalk and more. It also doesn't require to download and install anything, which should save lots of time [<http://www.yuuguu.com/home>].

Regardless of the IT solution chosen for the purpose of remote implementation of systems, both parties can also communicate via standard channels used in traditional implementation processes, such as e-mail, instant messengers, sets of frequently asked questions and answers (FAQ), helpdesk and hotline. Direct forms of contact in virtual implementation sessions are typically limited to minimum.

## V. CONCLUSIONS

Virtualization of business processes represents an effective use of information technologies, as one of the key determinants of success and a strategic resource of modern company. Information technologies open up new potential for operation and transform both means and methods of economic activities [3]. The use of virtualization potential in the process of IT system implementation offers ways for improving the operational capabilities of software providers and profitability of services addressed to small and micro-companies. In this respect, the IT providers, despite certain financial restrictions of their potential customers, can offer efficient implementation of IT solutions which would prove unprofitable for both parties if they were to employ traditional implementation methodology based on direct contact and carried out on-site. Through remote accessibility, the customer can benefit from consultant services, and the IT provider can carry out the implementation at the lowest possible cost. Another benefit of virtualization lies in the fact that the consultants can fully focus on using their key competences to address and satisfy the needs of the customer.

For obvious reasons, virtual implementation is typically limited to simple systems with low complexity and a range of predefined standard business processes. However, the skill and expertise resulting from such virtual implementations, coupled with best practices and methodologies developed in the course of servicing the small and micro-companies sector can bring profits in the foreseeable future, when

the IT providers decide to carry the virtual approach over to more advanced projects targeted to larger companies. Standardization of internal procedures on the part of IT providers, resulting from financial constraints and virtualization of services rendered to small companies, may effectively lead to optimization of procedures that take effect regardless of the scope of implementation processes and the targeted sectors. This approach, in effect, can bring about general reduction of implementation cost as well as considerable time savings. In this respect, the trend to virtualize IT system implementations seems a good direction, offering high potential for development on the part of IT providers as well as improvement and facilitation of the process from the viewpoint of future customers.

Case study of the virtualization of the IT system implementation process will be published in the AITM 2011 proceedings.

## REFERENCES

- [1] W. Szpringer: *Wpływ wirtualizacji przedsiębiorstw na modele e-biznesu*. Szkoła Główna Handlowa w Warszawie, Warszawa 2008, pp. 58–60.
- [2] D. Kisperska–Moroń: *Świat organizacji wirtualnych*. Zeszyty Naukowe Wyższej Szkoły Zarządzania Ochroną Pracy w Katowicach, Wydawnictwo WSZOP, Katowice 2008
- [3] M. Brzozowski: *Organizacja wirtualna*. Polskie Wydawnictwo Ekonomiczne, Warszawa 2010
- [4] P. Waszczuk: *Bliżej klienta i bardziej kompleksowo*. [In]: Top 10. Pierwsza dziesiątka producentów najpopularniejszych systemów ERP, Special report of Computerworld weekly magazine, March 2010
- [5] J. Kisielnicki: *MIS. Systemy informatyczne zarządzania*. Placet, Warszawa 2008, p. 203-205
- [6] J. Kisielnicki, H. Sroka: *Systemy informacyjne biznesu*. Placet, Warszawa 2009, pp. 135-137
- [7] K. Frączkowski: *Zarządzanie projektem informatycznym*, Oficyna wydawnicza Politechniki Wrocławskiej, Wrocław 2003, pp. 89-91
- [8] A. Jurga: *Technologia teleinformatyczna w organizacji wirtualnej*. Wydawnictwo Politechniki Poznańskiej, Poznań 2010, pp. 49-53
- [9] Webpage <http://www.crossloop.com>.
- [10] Webpage <http://www.netop.pl>
- [11] Webpage <http://www.symantec.com>
- [12] Webpage <http://www.teamviewer.com>
- [13] Webpage <http://www.tightvnc.com>



# An architecture of a Web recommender system using social network user profiles for e-commerce

Damian Fijałkowski  
 Wrocław University of Technology  
 ul. Wybrzeże Stanisława  
 Wyspiańskiego 27,  
 50-370 Wrocław, Poland  
 Email:  
 damian.fijalkowski@pwr.wroc.pl

Radosław Zatoka  
 Wrocław University of Economics  
 ul. Komandorska 118/120,  
 53-345 Wrocław, Poland  
 Email:  
 radoslaw.zatoka@ue.wroc.pl

**Abstract**—In this paper we propose a concept of a web e-commerce system that collects and uses, in the process of making recommendations, data obtained from social network profiles of its users. This architecture modeling approach was developed within the project of a mashup Web application that integrates with Facebook API. We describe which data could be obtained from Facebook, propose the way to store it and suggest how the information from user profile could improve the effectiveness of a e-commerce recommender system.

## I. INTRODUCTION

NOWADAYS more and more e-commerce platforms use recommender modules to propose their clients products that they would presumably buy, therefore expecting to increase their sales and incomes. Web e-commerce systems have access only to limited scope of demographic data, which users provide when registering.

These data are insufficient to use most of methods of recommendation. To widen this area, web platforms provide advanced user tracking, gathering all information about user activity in the system like searched phrases and browsed products. With the growing need of knowing more about their clients, e-commerce extends social network marketing and pursues to stay in touch with their clients on social services. Analyzing clients activities on social network services gives e-commerce an opportunity to create more personalized offer and help their clients to cope with huge informational overload problems continuously occurring in Internet shopping.

## II. RECOMMENDATION METHODS

Among many approaches to recommendation problems authors of [6] list most popular methods as follows:

- demographic filtering (DF),
- content-based filtering (CBF),
- collaborative filtering (CF),
- hybrid approach (HA),
- case-based reasoning (CBR),
- and rule-based filtering (RBF).

Modern web e-commerce systems mostly compute recommendations with the use of a hybrid strategy which for the most part is a mixture of three basic strategies called demographic, content-based and collaborative filtering [1, 2].

Demographic filtering is the group of least precise methods [13] that aim to find regularity among profiles of users who like particular object.

In content-based filtering strategy, an object is classified as relevant to a user, if it is similar to objects that, in the past, were recommended to him and accepted by him. System implementing CBF approach builds recommendations by analyzing a set of data previously rated by a given user.

Collaborative recommenders differ from content-based ones in that user opinions are used instead of content. CF approach assumes that an object should be suggested to a user, if it was rated as relevant by a group of users (neighbours) with a profile similar to the given user, provided that it has not yet been rated by him.

Despite the widespread use, above-named methods has two major limitations related to sparsity and scalability [5]. Sparse data environment causes that recommendation methods can't properly identify the products to recommend. Data derived from social network services can extend and enhance input data sets for demographic, content-based and collaborative filtering strategy.

## III. CONCEPT OF THE SYSTEM

Fig. 1 presents the overview of the e-commerce system integrated with the social network service.

### A. E-commerce

E-commerce platform (EC) contains core modules for implementing functionalities supporting the transaction of goods or services through electronic communication. These features include product catalogue, category browsing, product searching, basket and checkouts, online payments, order tracking, etc.

### B. Middleware

The architecture of proposed system is obviously distributed, therefore we distinguish the special tier that links core e-commerce module with Facebook web services. This middleware provide an uniform interface for the system to request for data from social network service and send them back to e-commerce system in the proper format, facilitating collaboration of both parts. Acquired data are stored in the profile system within the e-commerce platform.

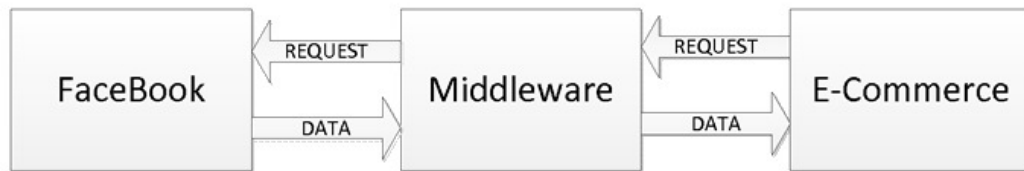


Fig. 1 Architecture of a Web system integrating e-commerce platform with Facebook

The role of middleware and example of its part in communication between both layers of the system is presented on Fig. 2.

### C. Social network service

Facebook, like most of Web 2.0 applications, moves towards Representational State Transfer (REST) based communications. Last year (2010) Facebook introduced new changes within its Open API, implementing the Open Graph Protocol [15] and the next generation of programming interface - Graph API.

Graph API provides developers a set of objects, on which they can operate to search data. This list includes: [14]

- all public posts,
- people,
- pages,
- events,
- groups,
- places,
- checkins.

The API is RESTful, therefore, when the agent has sufficient permissions, it can access to those objects simply by using HTTP request:

```

https://graph.facebook.com/search?
q=QUERY&type=OBJECT_TYPE
  
```

## IV. PROFILE SYSTEM

### A. Data structure

Fig. 3 presents the data structure model for the profile system. Clients of e-commerce platform who have Facebook account (and decided to provide us access data at registration or later, editing their preferences) and use social network service are additionally stored in *User* table in the profile system. Profile system collects keywords used by client and/or his friends and tries to determine the proper fields of his interests. Gathered keywords can have various context based on user activity in social network service or activity of his friends. All keywords are also categorized by the source of appearance on Facebook (e.g. post, link, comment). Profile

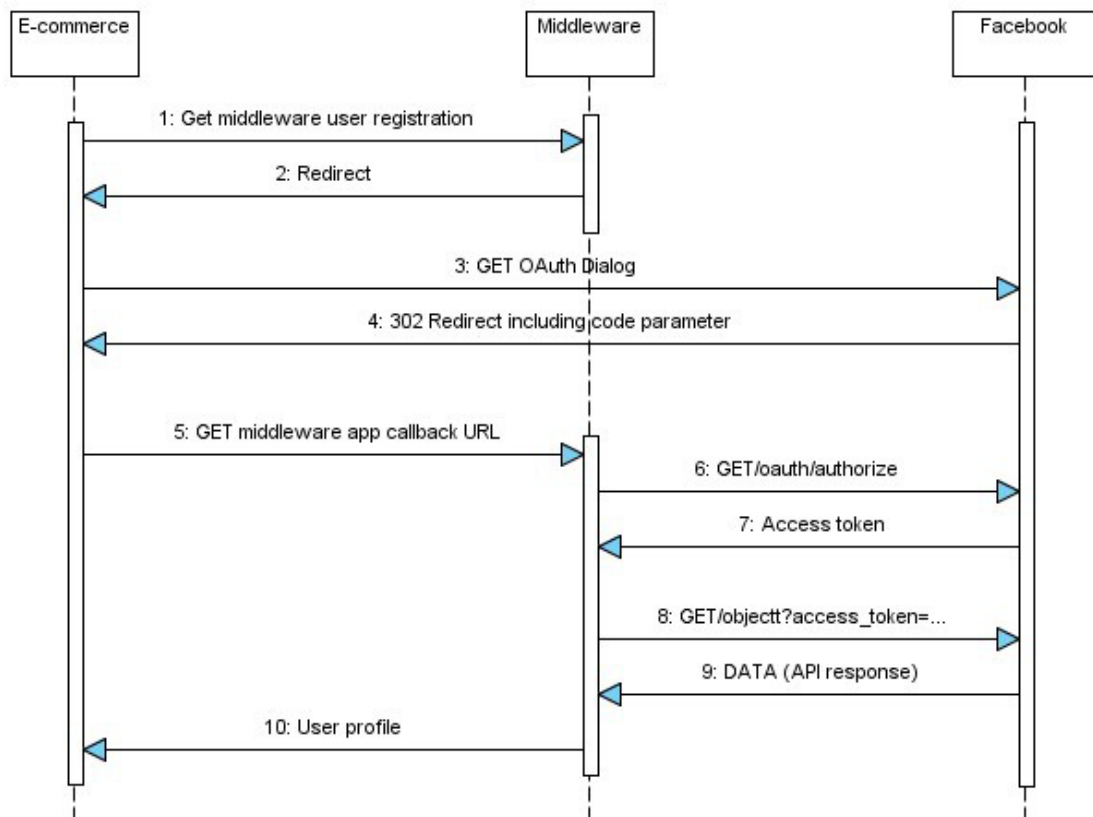


Fig. 2 Communication between tiers in e-commerce platform using Web services

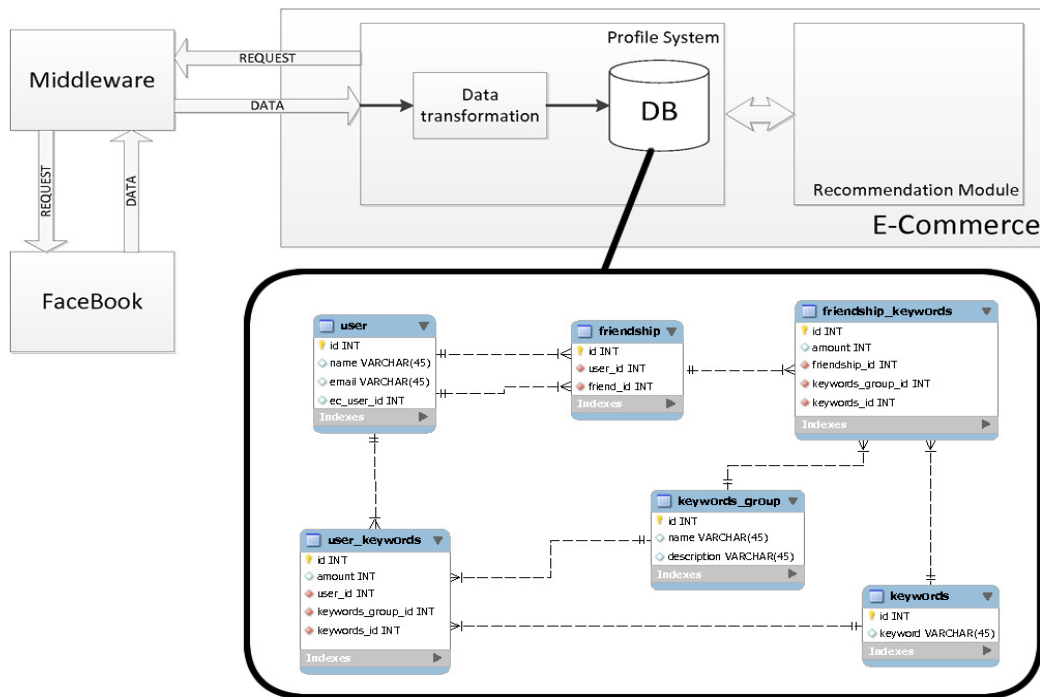


Fig. 3 Database model of profile system in e-commerce platform

system module is connected with the rest of e-commerce application by the *ec\_user\_id* key in *User* table.

In the profile system single client is identified by an email address that is used as a login to Facebook account. This gives us the possibility to automatically recognize, whether the user friends are already registered in e-commerce system. With this information final recommendation process can be improved.

#### B. Data source

Keywords obtained from Facebook can be categorized by their source of appearance:

- from user posts,
- from links published by the user,
- from comments added by the user,
- from user friends posts that the user likes (marked with 'I like it'),
- from links published by user friends that the user likes (marked with 'I like it'),
- from comments added by user friends that the user likes (marked with 'I like it'),
- from user messages (sent and received).

Each group presented above has its own weight representing significance of occurrence of words from particular group.

#### C. Web service application

The activity diagram of the middleware is shown on Fig. 4.

At first, the EC platform executes middleware application by sending a request for user data. Middleware makes a series of various Facebook API calls, dependent on how many data source sorts mentioned in part B are available for a given user. After receiving each response, data are

parsed from Facebook JSON format and processed. In this process, the combined user profile is being created in a middleware data layer. This temporary scheme reflects the structure required to integrate acquired data with EC user data. When the process is complete, the prepared user profile can be exported from temporary database structure and resend to EC platform in JSON format. This last call could also remotely invoke procedure that will update EC user profile subsystem with the latest obtained data.

#### D. Technical implementation notes

As it was mentioned earlier, Facebook gives developers access to the RESTful API, therefore the simplest way to obtain data is to invoke the service by HTTP request. Accordingly, any Web development programming language can be used to implement the middleware (e.g. PHP, Ruby, Java, Python). The authors suggest using PHP due to its stable position as a Web programming language on the market and a full support of Web services by language native libraries. Furthermore, as the most of websites are built with PHP, using this language to create own modified implementations of proposed middleware application could help to facilitate development process (e.g. some parts or single functions of data processing algorithms required by the middleware could already exist in EC platform). PHP is a cross-platform language, which can access various databases through consistent interface of PHP Data Objects (PDO) extension [16]. PDO database-specific drivers include i.a. opensource products like MySQL and PostgreSQL as well as commercial databases MS SQL Server and Oracle. However, such abstract interfaces exist in all modern Web programming languages and the use of Web services represent another abstraction layer. Therefore, the proposed architecture of middleware sys-



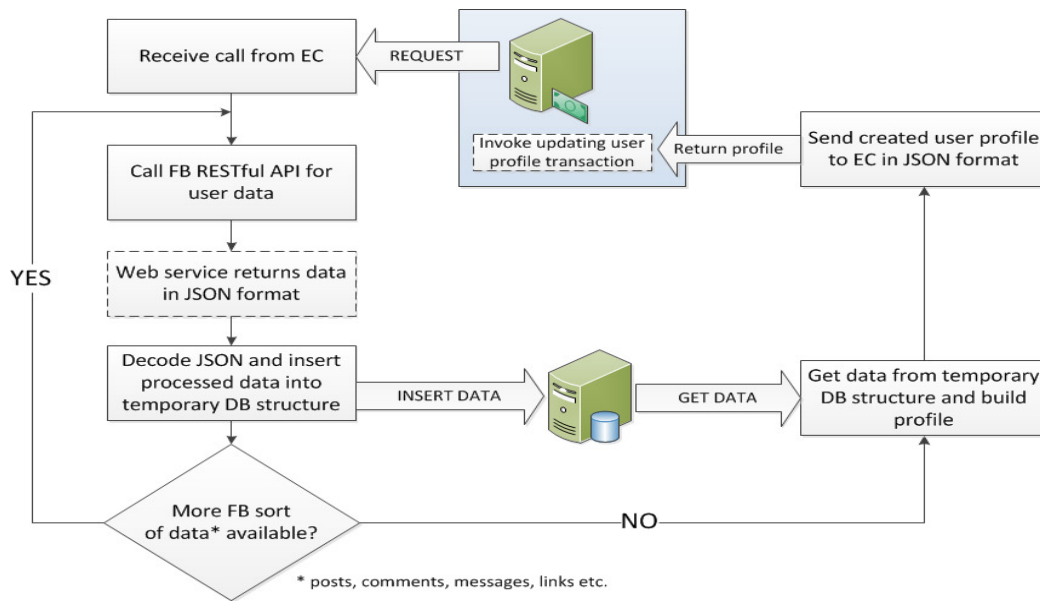


Fig. 4 Web service middleware application activity diagram

tem is flexible and does not require any fixed technical assumption concerning specific programming language and database system used in implementation.

#### V. RECOMMENDATION MODULE IN E-COMMERCE SYSTEM

Proposed system provides list of keywords which add new contexts to data sets. User profiles extended by the data obtained from Facebook are better inputs for demographic filtering.

Information concerning what user likes (objects marked by 'I like it' on Facebook) can be valuable for content-based approach recommendations.

Due to acquiring information regarding user friends on Facebook relations between users in e-commerce system can be discovered. Basing on those relations, it can be assumed that these users have similar interests. Moreover, this similarity is narrowed down to particular fields of interests based on keywords list in the context of particular friendship. As a consequence, collaborative filtering can be performed on enriched set of data.

#### VI. FURTHER WORKS

Further works should focus on defining weights for particular groups of keywords. Weights will represent significance of these groups and will have influence on values of keywords in those groups in recommendation process.

#### REFERENCES

- [1] Pasquale De Meo, Giovanni Quattrone, Domenico Ursino, "A query expansion and user profile enrichment approach to improve the performance of recommender systems operating on a folksonomy", *User Model. User-Adapt. Interact.* 20(1): 41-86 (2010).
- [2] Marco Degemmis, Pasquale Lops, Giovanni Semeraro, "A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation", *User Model. User-Adapt. Interact.* 17(3): 217-255 (2007).
- [3] Aleksandra Klasnja Milicevic, Alexandros Nanopoulos, Mirjana Ivanovic, "Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions", *Artif. Intell. Rev.* 33(3): 187-209 (2010).
- [4] Ziming Zeng, "An Intelligent E-commerce Recommender System Based on Web Mining", *International Journal of Business and Management*, Vol. 4, No. 7, July 2009.
- [5] Yoon Ho Choa, Jae Kyeong Kimb, Soung Hie Kima, "A personalized recommender system based on web usage mining and decision tree induction", *Expert Systems with Applications* 23 (2002) 329-342.
- [6] Janusz Sobiecki, Damian Fijałkowski, "Student Automatic Courses Schedule". Published in *New challenges for intelligent information and database systems*, ed. Ngoc Thanh Nguyen, Bogdan Trawiński and Jason J. Jung (eds.). Berlin, Heidelberg, Springer, cop. 2011. s. 219-226.
- [7] Rosario Girardi, Leandro Balby Marinho, "A domain model of Web recommender systems based on usage mining and collaborative filtering", *Requirements Engineering Journal* 1:23-40, 2006.
- [8] Frank Edward Walter, Stefano Battiston, Frank Schweitzer, "A model of a trust-based recommendation system on a social network", *Autonomous Agents and Multi-Agent Systems* 16(1): 57-74 (2008).
- [9] P. Bonhard, M. A. Sasse, "'Knowing me, knowing you' — using profiles and social networking to improve recommender systems", *BT Technology Journal*, Vol. 25 - No. 3, July 2006.
- [10] Shlomo Berkovsky, Tsvi Kuflik, Francesco Ricci, "Mediation of user models for enhanced personalization in recommender systems", *User Modeling and User-Adapted Interaction*, 18(3): 245-286, 2008.
- [11] F. Carmagnola, F. Cena, L. Console, P. Grillo, M. Perrero, R. Simeoni, F. Vernerio, "Supporting content discovery and organization in networks of contents and users", *Multimedia Systems*, Vol. 17, No. 3, December 2010.
- [12] J. Ben Schafer, Joseph Konstan, John Riedl, "Recommender Systems in E-Commerce", *Published in the Proceedings of the 1st ACM conference on Electronic commerce*, New York, 1999.
- [13] M. Pazzani, "A Framework for Collaborative, Content-Based and Demographic Filtering", *Artificial Intelligence Review.* 13(5-6) 393-408, 1999.
- [14] Facebook Graph API, <http://developers.facebook.com/docs/reference/api/>
- [15] Open Graph Protocol, <http://ogp.me>
- [16] PHP PDO, <http://www.php.net/manual/en/intro.pdo.php>



# Geospatial presentation of purchase transactions data

Maciej Grzenda, Krzysztof Kaczmarek, Mateusz Kobos, and Marcin Luckner  
Faculty of Mathematics and Information Science  
Warsaw University of Technology  
Warsaw, Poland

Email: {m.grzenda, k.kaczmarek, m.kobos, m.luckner}@mini.pw.edu.pl

**Abstract**—This paper presents a simple automatic system for small and middle Internet companies selling goods. The system combines temporal sales data with its geographical location and presents the resulting information on a map. Such an approach to data presentation should facilitate understanding of sales structure. This insight might be helpful in generating ideas on improving sales strategy; consequently improving revenues of the company. The system is flexible and generic – it can be adjusted to process and present the data within different levels of administrative division areas, using different hierarchies of sold goods. While describing the system, we also present its prototype that visualizes the data in an interactive way on a three-dimensional map.

## I. INTRODUCTION

COMPANIES selling goods have many possible ways to devise strategies of improving their business model and adjusting it to changing business conditions. One of them is to use business intelligence tools to automatically gather, process, analyze and visualize data that is important for the company in hope of obtaining useful insights that can be used to improve company's functioning. One of the most promising and simple approaches to this problem is to combine company's private data with publicly available data in order to obtain a useful synthesis of these two. An independent problem is how to handle and integrate different dimensions of company's data. One of the dimensions is the temporal one: the business conditions change over time and the company's decision-makers have to be able to follow changing trends in order to e.g. predict future behavior of the market. Another important dimension is the spatial one: different administrative regions have different business environments, and different business strategies might be more or less suitable for different sales areas (e.g. some regions might need more billboard advertisements while others might need more on-line advertisements).

In this paper, we describe an idea for an automatic system that combines private and publicly-available data of spatial and temporal type and visualizes it on a map. The main goal of the system is to present sales data of a company in an useful and interactive way. The system is simple but generic – it can be adjusted to process and present data within different levels of administrative division areas, using different hierarchies of goods sold by the company. Apart from describing the general idea, we also present a prototype of such a system that uses

data from one of the Polish companies. The company is one of the largest Internet sellers of tires in Poland.

An overall process of data acquisition and transformation in the system is presented in Fig. 1. Our system automatically combines purchase transaction records data with information about spatial placement of administrative division areas of region of interest to locate an approximate place where each purchase was delivered. To be more precise, we use the information about delivery town and zip code of a purchase to determine which administrative division area the buyer is situated in. Each area has GPS coordinates assigned to, so the data related to this area can be easily placed on a map. The data is saved in a form of a relational database. Next, a geographic data visualization application is used to present the data in an interactive and user-friendly way. Since there are many mature applications which can be used as a visualization engine, we decided not to implement our own in the prototype. Public and free tools, although not perfect, are mature enough to be used in a professional solution. Their displaying capabilities are not limited to any particular area and can usually show different geographical regions all over the Earth. One of the most popular tools of this type, i.e. spatial data viewer equipped with ability to load user data, is Google Earth [1]. It proved to meet our requirements and we used it in our prototype (the prototype allows also sharing the visualization on-line via Google Maps).

### A. Related Research

Problem of storing and presenting spatiotemporal data is generally addressed by dedicated systems: SOLAP (Spatial On-Line Analytical Processing) being *a visual platform built especially to support rapid and easy spatiotemporal analysis and exploration of data following a multidimensional approach comprised of aggregation levels available in cartographic displays as well as in tabular and diagram displays* [2]. As all OLAP-based solutions, they require wide knowledge of data processing and data mining, in this case often combined with expertise in cartography. Another drawbacks of these systems are high licence fee and maintenance costs and therefore low return on investment values which are not acceptable for small companies.

Our lightweight data processing components try to answer spatiotemporal data analysis demands in much simple and cheaper way.

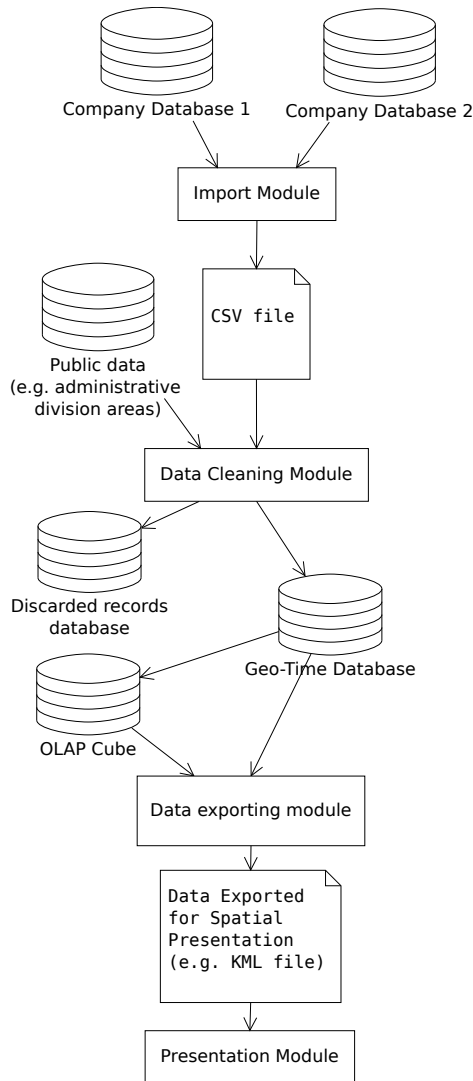


Fig. 1. Data processing phases

An important research was done to allow for fast and precise spatiotemporal aggregation calculation. This task is not easy due to imprecise querying and spatial selection criteria. Especially R-Tree structure [3] with many improvements is used to store spatial information [4]. Adding the time dimension was analyzed in many works i.e.: [5], [6], [7], [8]. Another path in research is devoted to streaming data systems and calculation of incremental spatial aggregates [9].

In our research, we focus mostly on Internet transaction data cleaning and coupling it with spatial information leaving an effective data storage and querying methods as open topics. Usage of R-Trees is still possible and could improve performance for large datasets. However, currently we do not consider this to be an important problem since our data comes from small companies and does not exceed volumes that can be effectively processed by a simple relational database management system. We tend towards simplicity for users processing Internet transactions.

Available tools for modeling and visualization of spatial data can be categorized as a stand-alone and web-based [10]. Typical stand-alone commercial products are ArcGIS and MapInfo. Both are expensive and dedicated for advanced users. As an alternative, PyNGL and PyNio applications, developed using Python programming language, are available. These applications generate 2D visualizations in several formats.

Among web-based visualization applications there are also solutions, which are based on commercial products (Bentley Map, ESRI) but many of these systems work only with their own datasets [11]. The prototype proposed in this paper is based on free software and allows presentation of user's data against a background of a third-source data.

## II. DATA EXTRACTION, TRANSFORMATION, AND LOADING

In this section we describe an algorithm which is used to process input data and prepare presentation layer in our system.

To create the final presentation, we gather data from different sources:

- 1) company's private database of transactions,
- 2) administrative areas with zip codes,
- 3) statistical data for administrative regions.

The most important data comes from a transactional database of a company. Obviously, this information contains quantities, products, categories, clients, prices, values, etc. An initial data transformation module processes the data and prepares it to be imported into our tool. Possibly, the most simple way to import this data is to use a \*.csv file where each row of the file describes a single purchase transaction. Each transaction in such a file is described by: time of the purchase, delivery zip code, delivery town, price of the purchase, quantity of the ordered product, and localization of a product in a hierarchy of types of products (see Fig. 2, *Purchase* table). In case of the data used by our prototype, we have two levels of the product. The product is the tire in this case. The top-level type is a brand of the tire while second-level type is the name of the tire, unique within the bounds of a single brand.

Each purchase can be approximately located on a map, and as such it is presented with respect to different levels of administrative division. Our system allows defining custom hierarchy of levels suitable for given application domain (see Fig. 2, *Administrative division hierarchy* group of tables). For example, in case of a company selling products in the USA, the hierarchy might look as follows: "state" → "county" → "city, town, or village". In case of our prototype, the data comes from a company selling products exclusively on the territory of Poland. Thus, while visualizing the spatial information, we use the information about Polish administrative territorial division. Polish territory consists of 16 voivodeships or provinces. Each voivodeship, called "województwo" in Polish, consists of a number of second level of local government administration areas, each one called "powiat". There is a total number of 379 powiats in Poland. For each considered administrative area, we

have obtained map coordinates of its center. A center for an administration region is calculated automatically as a centroid of administration area border taken from public government database [12]. For each powiat, we have also gathered a list of zip codes belonging to the powiat and town names connected with each zip code (see Fig. 2, *ZipCodeTown* table).

Because some town names can be written in a number of different but equivalent ways, we also use a simple text file in a \*.csv format to store information about alternative spelling of names of some of the towns. In case of our prototype, two sample entries in this text file are: 12-220, Ruciane Nida, Ruciane-Nida and 80-299, Gdansk-Osowa, Gdansk where the first element is the zip code, the second one is the alternative spelling, and the last one is the canonical name (see Fig. 2, *TownNameAlias* table).

Our main goal in processing the above-mentioned data is to assign suitable administrative division areas of each level to each purchase transaction (i.e. delivery destination). To achieve this goal, we clean the data (described in Section II-A), then we transform and combine it (described in Section II-B), and finally we load into a final database used by an application that visualizes the data (see Fig. 3). In the description of the data processing, we concentrate mainly on the spatial dimension, but the temporal information is still present in the data, although its processing is limited mainly to generating the final summary statistics from selected time interval.

The last source of data is the statistical data of administrative division regions important for particular business. This could include for example population, climate, or number of high schools. If we possess an information connected to given administration area, it can be imported and presented together with statistical transaction information. It can also be used to normalize presented data, like for example displaying number of sold bottles of water per person.

#### A. Data Cleaning

Due to characteristics of the input data i.e.: large influence of the human factor, possible mistakes, uncertainties, and ambiguity, the imported data has to be cleaned. The general approach is to accept correct records, repair the records that we know how to repair, and discard all others. The discarded data is saved in an auxiliary database along with information why each record was discarded. By inspecting this auxiliary database, we can check if the cleaning process improperly throws out useful records, and if it is the case, we can try to improve the cleaning algorithm.

Among all of the fields in the input records, the zip code and the town name have to be given a special care since normally they are entered by hand by each buyer using a web order form, and as a result there might be many possible versions of the same information entered. In case of our prototype, we deal with Polish zip code. It has a format of XX-XXX, where X is a single digit. While cleaning the zip code value, we: 1) remove all the spaces; 2) replace \*, \_, =, / symbols with hyphen; 3) remove textual zip code suffix (if any) consisting

of e.g. town name; 4) replace letters “o” and “O” with zero; 5) add hyphen in appropriate place; 6) add leading zero and a hyphen in a four-digit zip code without hyphen.

Next, while cleaning the town name value, we: 1) remove excessive spaces; 2) convert the name to a title format (a capital letter at the beginning of each word); 3) convert the name to the canonical name if it is in the table of the names with alternative spelling. In case of our prototype, the name of the analyzed town is sometimes “test” which is not a real name, but just a marker of a record created for test purposes. In such situations, the analyzed record is discarded.

Additionally, if type hierarchy of a purchased product is not fully specified in the record, the record is discarded. In case of our prototype, we discard the record if either brand or type name of a tire is absent.

#### B. Combining Geospatial and Time Information

After doing the basic cleaning of the data, we try to assign administrative division area of the lowest level to each purchase record. It is worth noting that the higher-level administrative areas do not have to be assigned explicitly since each lower-level area is assigned to a single higher-level area. In case of our prototype, this task is done in two steps. In the first step, we look for a powiat identified uniquely by the given zip code only. If it fails, we proceed to the second step and look for the powiat identified uniquely by the purchase’s (zip code, town name) pair. A more precise description of this process is presented as a pseudocode below:

```

p ← {get powiat names associated with given zip code}
if |p| = 1 then
  {assign powiat to the given record}
else if |p| = 0 then {Given zip code was not found in the
  database}
  {discard record}
else {There was more than one powiat found for given zip
  code}
  {We were unable to uniquely identify the powiat using
  zip code only, so we will try to do it using also the town
  name}
p ← {get powiat names associated with given zip code
and town name}
if |p| = 1 then
  {assign powiat to the given record}
else if |p| = 0 then {powiat for given (zip code, town
name) pair was not found}
  {discard record}
else {There was more than one powiat found for given
(zip code, town name) pair}
  {discard record}
end if
end if

```

As can be seen, the data is discarded in various points of the cleaning and transformation process. Since we are storing discarded data in an auxiliary database, we can easily generate some high-level statistics showing how much data was discarded and what was the reason of the rejection.

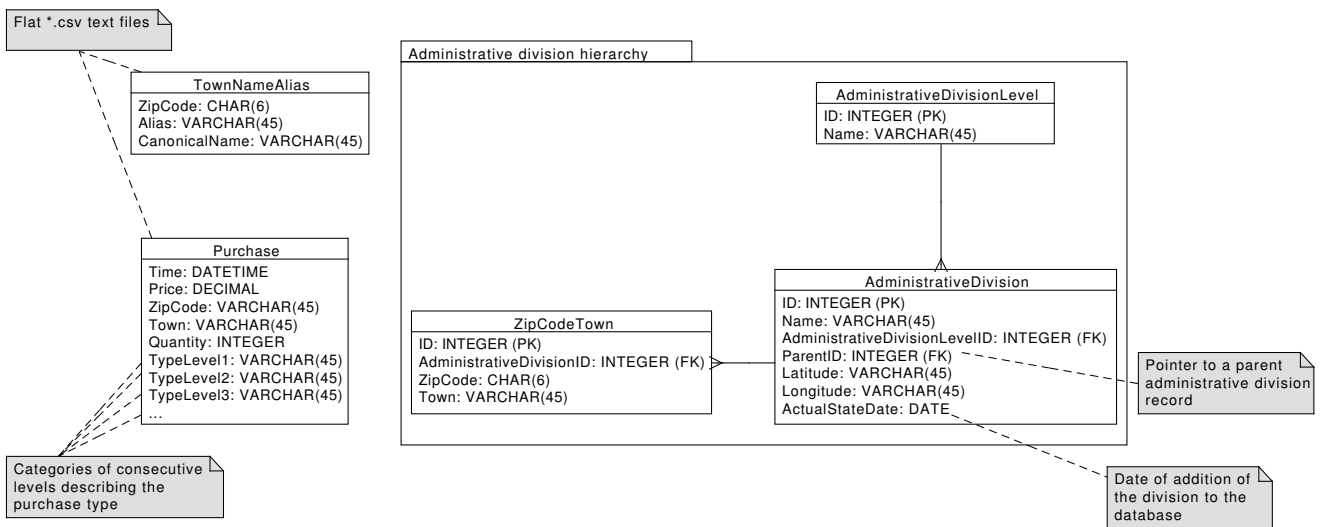


Fig. 2. A schema of the input data that we use to create the final database. Each box represents a logical database table. The following structures are presented: *Purchase* – a table of purchase transactions, *TownNameAlias* – a table of alternative spelling of names of selected towns, *Administrative division* – group of tables describing the administrative division of the area.

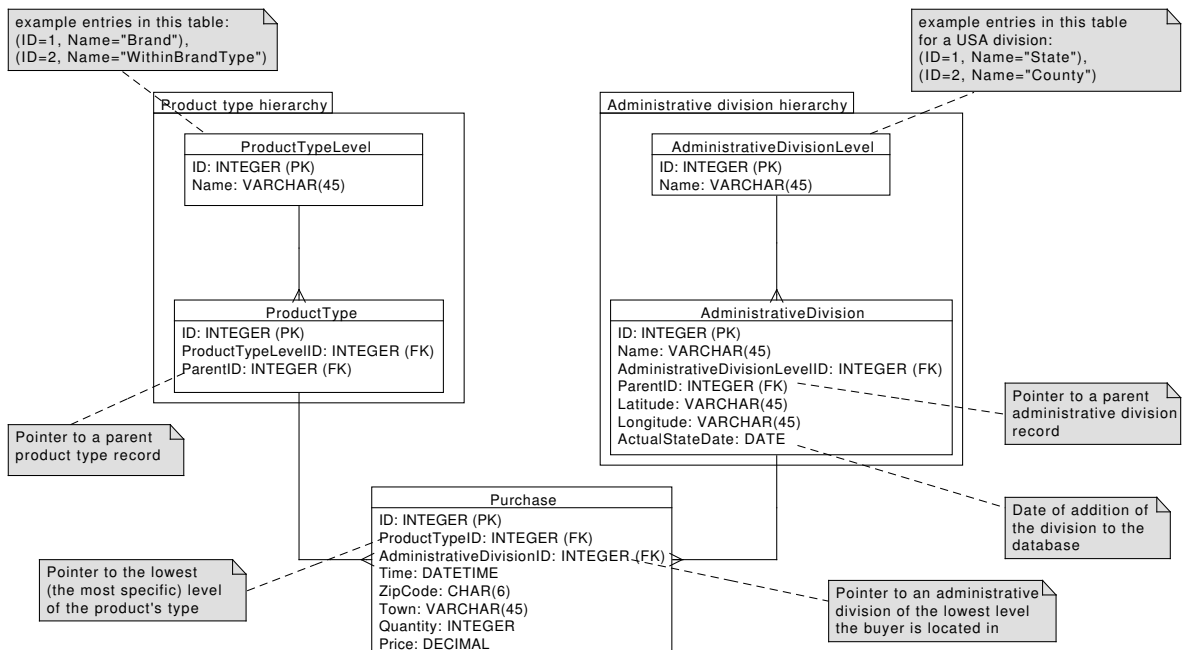


Fig. 3. A schema of the final database used by the application.

Overall, just a small percentage of the data is discarded in the cleaning process.

C. Statistical Processing and Data Output

After the transactions data is cleaned and stored in a database, we can start extracting statistical information. In many cases, for most of the small companies, this step will just calculate simple aggregates like sum of transactions or average

purchase value. A set of simple queries may be used to process this information and send it to the presentation layer. However, in various situations a more complicated statistics like growth of value per product category may be needed. In these cases an additional OLAP system may be used to store temporal aggregates.

A map feeder module uses all available information to create the presentation layer. It combines area, transactions,

and statistics to automatically produce results. Due to possibly large volume of data to be processed it should work offline and in a batch mode. Output data, depending on a time window involved and length of analysed period, may have from tens to hundreds of megabytes.

The output data is prepared in a format acceptable by the presentation module (see the next section for details).

### III. PRESENTATION LAYER OF WORKING PROTOTYPE

The final database (see Fig. 3) comprising of integrated data from different sources is used as a basis for producing input data for the visualization module. Although any tool can be used to display the data, it should have at least basic capabilities required for user-friendly operations:

- zooming in and out with administration areas appearing automatically,
- panning around the map,
- displaying of user data values,
- time axis and ability to move in time and display data for given time window,
- ability to divide user data into layers or provide data grouping.

The Google Earth (GE) application has all the above-mentioned characteristics, that is why we use it as a visualization engine in our prototype. The data accepted by the GE is described in an XML file in a format called OpenGIS KML Encoding Standard (abbreviated simply as “KML”). This format is specifically designed to describe a way of visualizing geographic data. We use its basic capabilities to visualize sales data bars as three-dimensional polygons on a three-dimensional map of Poland.

After loading the \*.kml file generated from our database into the GE, the user sees sales performance bars placed on each administrative area. There are four bars per area, each one corresponds to sales performance in one of four consecutive months (see Fig. 4). The user can utilize many options implemented in GE to navigate and manipulate data:

- change viewed time frame,
- run month-by-month animation showing changes in sales performance (see Fig. 5),
- change point of view,
- select subset of the data to visualize,
- get detailed information about a selected sale (number of transactions, total value, etc.).

One of the most important limitations of GE as a visualization tool in our system is that data subsets may only be defined as disjoint groups in XML format. Therefore data must be repeated in many so-called folder structures in order to achieve visualization of the same property in different layers or areas. This could result in a huge KML files if one would like to see different products divided into different areas. Also, adding time dimension multiplies the file size by the number of time steps. We observed in our prototype a file size growth of two orders of magnitude when using 24 time steps (each corresponding to a single month) instead of using a single aggregated step.

However, the system should also work with larger datasets. KML-based models can manage datasets with millions of records [13]. This is especially true when a network links technique is used [14].

### IV. CONCLUSIONS AND FUTURE WORKS

We presented a lightweight automatic system for combining, processing and presenting sales-related data. The presented prototype of the system relies on batch processing for data analysis and on Google Earth application as a viewing module. Our solution, although very simple, could be used by most of Internet sellers providing them a simple and convenient way to observe spatial and temporal relationships in sales data. Future work on the system involves a more thorough incorporation of the statistical data of the administrative regions into the system. Another idea is to provide visualization of results of some basic data mining processing of the analyzed data (e.g. showing clusters of regions that are similar in some specified way).

### ACKNOWLEDGMENT

The authors would like to thank one of the largest Internet sellers of tires in Poland: ORZEŁ S.A., Ćmiłów ul. Willowa 2-4, 20-388 Lublin, Poland. The company made available approximately two years of Internet purchase transactions data to us.

### REFERENCES

- [1] G. Corporation, “Google earth,” [www.google.com/earth](http://www.google.com/earth).
- [2] Y. Bédard, S. Rivest, and M. José Proulx, “Spatial on-line analytical processing (solap): Concepts, architectures, and solutions from a geomatics engineering perspective,” in *Data Warehouses and OLAP: Concepts, Architecture, and*. Press, 2006, p. 298319.
- [3] A. Guttman, “R-trees: A dynamic index structure for spatial searching,” in *International Conference on Management of Data*. ACM, 1984, pp. 47–57.
- [4] Y. Theodoridis and T. Sellis, “A model for the prediction of r-tree performance,” 1996, pp. 161–171.
- [5] Y. Tao, J. Sun, and D. Papadias, “Analysis of predictive spatio-temporal queries,” *TODS*, vol. 28, pp. 295–336, 2003.
- [6] M. Hadjieleftheriou, G. Kollios, V. J. Tsotras, and D. Gunopulos, “Efficient indexing of spatiotemporal objects,” 2002, pp. 251–268.
- [7] Y. Theodoridis, M. V. T. Sellis, M. Vazirgiannis, and T. Sellis, “Spatio-temporal indexing for large multimedia applications,” 1996, pp. 441–448.
- [8] Y. Tao, G. Kollios, J. Considine, F. Li, and D. Papadias, “Spatio-temporal aggregation using sketches,” in *ICDE*, 2004, pp. 214–226.
- [9] J. Zhang, “Spatio-temporal aggregation over streaming geospatial data,” in *Proceedings of the 10th International Conference on Extending Database Technology Ph.D. Workshop*, 2006.
- [10] D. Kannangara, N. Fernando, and D. Dias, “A web based methodology for visualizing time-varying spatial information,” in *Industrial and Information Systems (ICIIS), 2009 International Conference on*, dec. 2009, pp. 233–238.
- [11] J. K.P. and W. N.T.S., “Product development for presentation of temporal gis results for non gis specialists, engineer,” *Journal of the Institution of Engineers*, vol. 51, no. 5, pp. 44–50, 2008.
- [12] Surveyor General of Poland, “geoportal.gov.pl,” [geoportal.gov.pl](http://geoportal.gov.pl).
- [13] J. Wood, J. Dykes, A. Slingsby, and K. Clarke, “Interactive visual exploration of a large spatio-temporal dataset: Reflections on a geo-visualization mashup,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 6, pp. 1176–1183, nov.-dec. 2007.
- [14] U. Dadi, C. Liu, and R. Vatsavai, “Query and visualization of extremely large network datasets over the web using quadtree based kml regional network links,” in *Geoinformatics, 2009 17th International Conference on*, aug. 2009, pp. 1–4.



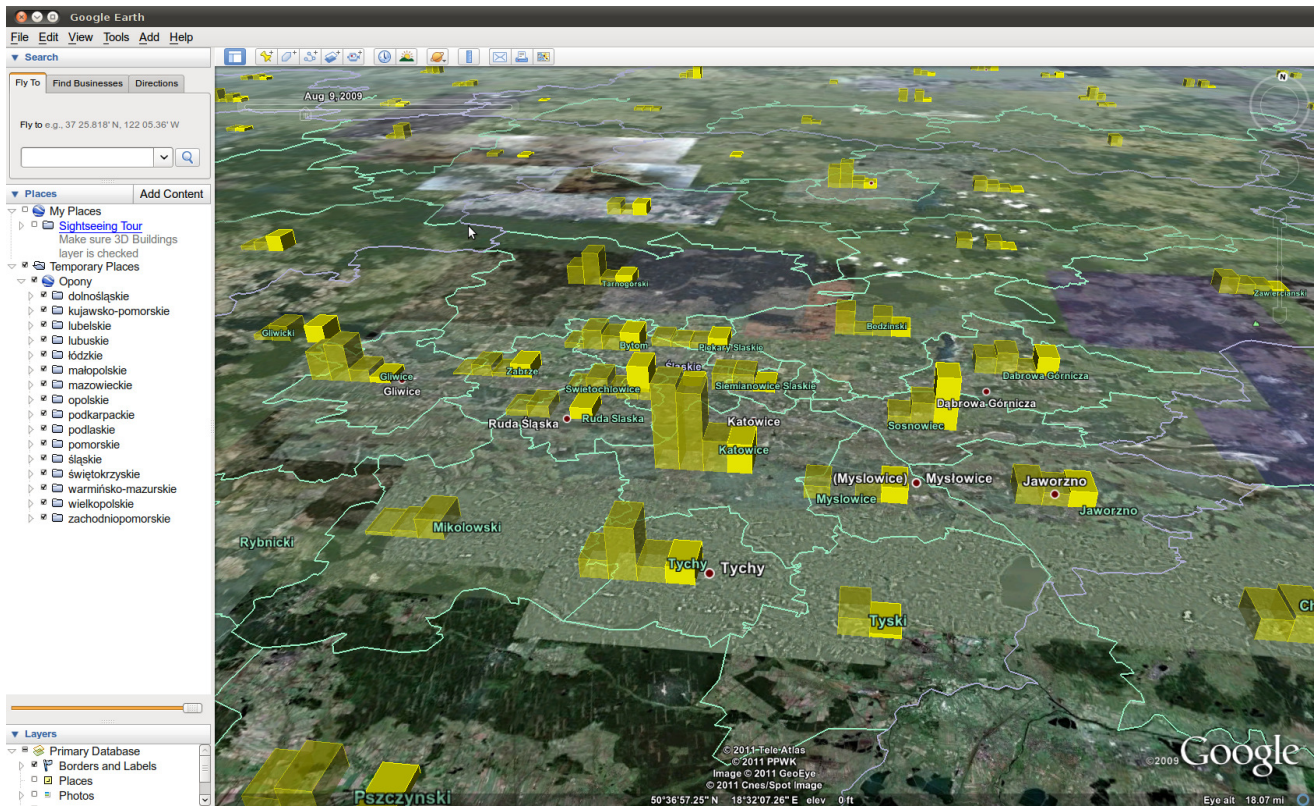


Fig. 4. A sample screenshot of the presentation layer. We can see a Google Earth's satellite image of a part of Poland with borders of powiats marked. Three-dimensional bars on the territory of each powiat correspond to sales performance in four consecutive months with the brightest bar on the right side corresponding to the most recent month.

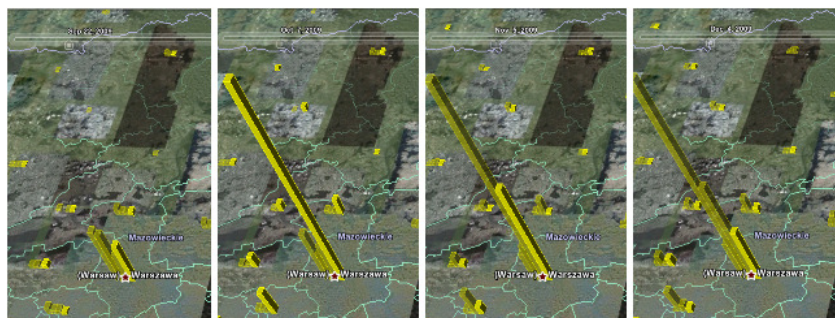


Fig. 5. Subsequent frames of animation showing how the sales performance is changing during four consecutive months. The brightest bar on the right side corresponds to the most recent month.

# Explaining MCDM acceptance: a conceptual model of influencing factors

Martina Maida, Konradin Maier, Nikolaus Obwegeser and Volker Stix  
Vienna University of Business and Economics, Institute for Information Business,  
Augasse 2-6, 1090 Wien, Austria

**Abstract**—The number of newly developed Multi-Criteria Decision Making (MCDM) methods grew considerably in the last decades. Although their theoretical foundations are solid, there is still a lack of acceptance and application in the practical field. The objective of this research is the development of a conceptual model of factors that influence MCDM acceptance that serves as a starting point for further research. For this purpose, a broad diversified literature survey was conducted in the discipline of technology adoption and related topics (like human computer interaction) with special focus on MCDM acceptance. The constructs collected within the literature survey were classified based on a qualitative approach which yielded a conceptual model structuring the identified factors according to individual, social, technology-related, task-related and facilitating aspects.

**Index Terms**—Technology Acceptance, Multi Criteria Decision Making, Decision Support.

## I. INTRODUCTION

RESEARCH in the field of technology acceptance has been subject to numerous developments in the last decades. Additionally, research in this specific area is of quite broad nature, such that it builds on various contributing domains like innovation-research, human-computer interaction (HCI) and many others.

Moreover, the advent of decision support technologies in the early 70s has been the start of what should become an active research field in both information systems (IS) and operations research (OR). While theoretical contributions show significant advancements in this area, the adoption rate of sound decision support methodologies in the practical field remains on a rather poor level. Thus, the acceptance of decision support systems (DSS) evolved as a special case of technology acceptance research.

The underlying paper aims at analyzing and integrating the main research streams in the fields affected. Therefore, a comprehensive literature survey was conducted to identify the inner structure of this field. This laid the groundwork for the design of our conceptual model. We started the survey by examining the most prominent models in the discipline of technology acceptance. These basic publications then served as a starting point for a snowball-technique based literature search. We reviewed all major journals for research of technology acceptance-related topics for the time span of 1980 until today. Furthermore, special attention was given to those research efforts that were concerned with multi-criteria decision-making

(MCDM) problems. Although the term MCDM emphasizes the complexity of the decision problems targeted, both terms DSS and MCDM are often used synonymously (as in the underlying paper). This initial phase resulted in a compendium of over 100 constructs that were found in literature to have an influence on technology acceptance, respectively MCDM acceptance.

In a second phase we performed a qualitative analysis on the (at this point) inhomogeneous collection of influencing constructs. There were also many variables with different levels of semantic granularity. Thus, the main objectives of this analysis were (i) to clearly define the semantics and denotation of each construct as intended by the original author, (ii) to identify any redundancies and (iii) to mark different levels of detail within the constructs. Furthermore, we established a mapping over the course of the analysis that builds up to a network-like structure and allows us to depict related constructs and parent-child relationships.

In a third phase, the consolidated and non-redundant list served as a basis for a process of inductive category formation [1]. We therefore discussed various schemata and concepts that could fit the underlying data along with a review of classifications proposed by other researchers or theories. The category formation process led to valuable insights on the details of the constructs and its interdependencies and eventually resulted in the categorization scheme and conceptual model that will be described in Section III.

The remainder of this paper is structured as follows. First, we present a review of the major streams that constitute the basis for our research, as described in Section II and with special focus on technology acceptance and MCDM acceptance. Section III presents and discusses the conceptual structure and the developed model. A detailed description of each group is given along with exemplified member constructs. Section IV provides a conclusion and points out promising research areas for ongoing investigations.

## II. LITERATURE SURVEY

This section describes two of the major streams of research that were examined within the literature review. As stated before, special focus was on technology acceptance models, related domains and on topics concerning MCDM acceptance.

### A. Technology Acceptance Models

Research in the field of technology acceptance has been an active field of research for the last decades. It is a telling observation that the original reason for academics to perform research in this area was mainly of practical nature: What are the driving factors for failure or success of technology? This was soon adapted to a behavioral, human-centered view on the problem, changing the main research question to: How do individuals perceive software, their surroundings and what beliefs ultimately lead to usage of a technology? Consequently, much research effort was put into psychological analysis and theory-building of cognitive processes that resulted in numerous models and theories in the respective field. While these advancements are undoubtedly valuable for the forthcoming of the scientific field, some researchers call for more diversification in this research area. When Orlikowski and Iacono titled their heavily discussed research paper “Desperately seeking the ‘IT’ in IT research: A call to theorizing the IT Artifact” [2], they intended to break the ice for what is often prematurely dismissed as system-building task: research on the actual IT artifact. Although the area of technology acceptance can be considered a rather broad research field with numerous drivers for successful acceptance, the usage of a certain technology is at the very core of it. Prominent behavioral models try to explain the lack of acceptance from a human centered perspective. While these models do not differentiate much from a technological point of view, research in human-computer interaction (HCI) focuses on the investigation of specific characteristics. Additionally, investigations on the influence of individual traits, the social environment and task specifics have been successfully added to the field.

1) *Behavioral models*: One of the most prominent and disputed contribution to the area of technology acceptance has been made by F.D. Davis with the proposal of the technology acceptance model (TAM) [3]. TAM is a psychological model based on the theory of reasoned action (TRA), developed by Ajzen and Fishbein [4]. It tries to illustrate the abstract relationships that lead to failure or success of a technology. The original version of TAM is limited to only a few very high-level constructs, such as perceived usefulness or perceived ease of use, and has hence been subject to criticism and further development. Follow-up models such as TAM2 and TAM3 basically augment the initial high-level model with the integration of numerous fine-grained influence factors [5], [6]. TAM and its successors have been used intensively in empirical research and therefore constitute one major part in the field of technology acceptance. Alongside, DeLone and McLean proposed the information systems success model, integrating six major categories of measures that affect IS success [7]. In contrast to TAM, the IS success model is not only focused on acceptance of a technology but rather on the individual and organizational impacts. After numerous contributions following the initial proposal a revised model of the IS success model was proposed ten years later that replaced

the orientation on impacts with net benefits and allows for feedback loops [8].

In addition to these specialized works many other contributions from psychology and cognitive sciences have found their way into technology acceptance research (e.g. Bandura’s social cognitive theory (SCT) or the motivational model (MM) proposed by Davis et al. [9], [10]).

2) *Technological research*: Due to the fact that the technological artifact is at the center of technology acceptance research, many contributions from the field of human-computer interaction (HCI) are valuable when adopted and integrated into acceptance research. HCI research can be considered the intersection between behavioral sciences and computer science, therefore offering insights into the design and perception of IS-artifact characteristics. Especially when considering visual representations for IS, the cognitive fit theory (CFT) proposed by Vessey allows for a deeper understanding of the possible disadvantages that come with the utilization of such [11], [12]. Moreover, the computers are social actors (CaSA) approach shows how different levels of perceived social presence can influence acceptance and usability of a technology [13].

3) *Other contributions to the field*: As follows from the above, technology acceptance research is embedded in a rather broad social environment and subject to numerous influencing factors. Many other research streams aside from the aforementioned behavioral and technological research areas are providing promising contributions to this field. A prominent example is the model of task-technology fit (TTF), that focuses explicitly on the degree of compatibility between task and technology [14]. While most research attempts incrementally add to the forthcoming of the field, others try to abstract existing knowledge to form a more holistic approach. This strategy has been pursued by the authors of the unified theory of acceptance and use of technology (UTAUT), who tried to integrate the findings of eight existent models/theories (including TRA, TAM, MM, etc.) to establish a single but comprehensive approach [15].

### B. MCDM Acceptance

Within the broad field of technology adoption the usage behavior of decision support systems (DSS) has emerged as an important subfield of research. We argue that several reasons account for this development. First, the problem of supporting decision makers in making good decisions has always attracted many researchers. On the other hand, the acceptance of decision support methods and systems (further referred to as DSS acceptance) within the practical field is rather low. Therefore, it does not come as a surprise that the gap between theoretical advancements in decision support and poor adoption of DSS in the practical field has become its own research area. A second reason for this development is that there are several major differences between most conventional information systems and decision support technologies. Conventional IS technologies (e.g. mail clients, word processing, etc.) are rather simple in terms of control. This



means, that the user usually has the ability to understand and to predict the system's reaction to inputs. Therefore, the user perceives nearly absolute control over the outputs generated by the system. In contrast, most DSS are based on complex mathematical models to process information which reduces the understandability and predictability of the system's reactions. Thus, the user perceives substantially less control of the decision support system's behaviour and outputs than he perceives using a conventional IS technology.

As DSS acceptance is considered a special case of technology acceptance, research in DSS usage behavior evolves around similar main constructs as research in technology adoption (e.g. intention to use decision aid [16], decision quality [17], perceived usefulness [18]). However, research in DSS acceptance differs distinctly from other areas of technology adoption due to the explicit separation of the acceptance of the system from the acceptance of the underlying theory implemented in the system. The rationale for this is that the user has to accept both the MCDM method (theory) and the technology (tool) implementing this process [18].

A prominent model in the context of evaluating the acceptance of decision making methods is the effort-accuracy model of cognition developed by Payne et al. [19]. This model suggests that decision makers are naturally capable of several decision making strategies and select one of these strategies based on trade-off considerations between the effort to implement a strategy and its accuracy. This model has been extensively used in the context of decision support acceptance, for example by Benbasat and fellows (e.g. [20], [21], [16]) but also by others (e.g. [18]). Based on this model, it was shown that a certain decision making strategy is more likely to be used if a DSS reduces the cognitive effort to employ this strategy relative to other strategies [20].

On the system-side of DSS acceptance, much research focuses on the identification and evaluation of design features that influence the acceptance of DSS technologies. This includes, for instance, the design of the user interface (e.g. [22], [23]) and other topics related to human-computer interaction like the wording and structuring of the dialogue with the user [24]. An important concept within DSS acceptance literature is the decisional guidance framework developed by Silver [25]. Decisional guidance is defined as the way how a DSS guides and directs its users as they execute their decision processes. It has been the groundwork for much empirical research (e.g. [18], [16]) and has been incorporated in Benbasat's concept of explanation facilities, which provide the user with how and why explanations as well as with process guidance [26], [16].

### III. A CONCEPTUAL PERSPECTIVE ON MCDM ACCEPTANCE

Research in technology acceptance is closely related to and often based on psychological concepts that target human cognition and perception. Due to the broad area of possible influences on the usage of technology, most researchers try to narrow down the scope of their research by limiting empirical

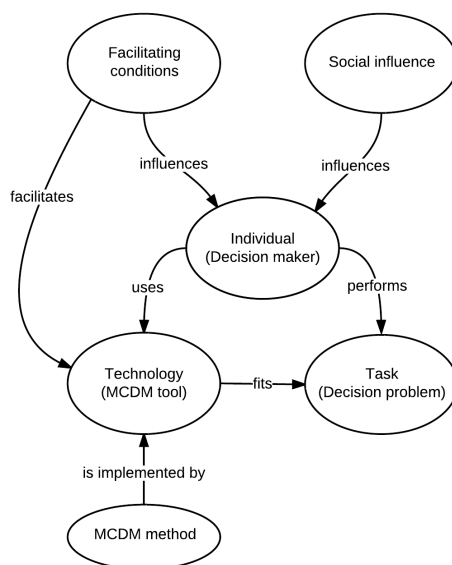


Fig. 1. Conceptual model of factors involved in MCDM tool usage behavior

investigations to certain areas of technology acceptance (e.g. visualization capabilities, individual differences). As a result of this practice a great number of models and theories have spawned that explain small parts of this research area. This process led to a rather unstructured research field where it seems hard to identify clear streams and future research possibilities. Therefore, as stated in Section I, one main goal of this research attempt is to synthesize and structure the list of possible influence factors and to conceptualize a model. Using an inductive categorization formation approach, we established the following major groups of influencing factors: individual, task, technology, method, social and facilitating conditions. These groups are intended to serve as a conceptual categorization for low-level constructs. High-level constructs (e.g. intention to use or perceived ease of use) tend to be an aggregation or a result of the combination of multiple low-level constructs and are therefore not clearly assignable to a single group. Due to the focus on MCDM systems the technical terms of this research area are used when applicable (individual - decision maker, technology - MCDM tool, task - decision problem).

The model presented in Fig. 1 provides a static perspective on the system at hand. It is not intended to explain or hypothesize on causalities or dependencies. It depicts a conceptualized overview of the field of MCDM acceptance and its key influences. Each ellipse represents one group that has been identified as described above. Edges between the groups model their associations and are labeled to describe the respective semantics of their relationship. The edge "uses" represents the most important relationship concerning acceptance, that is, the actual usage of the MCDM tool by the decision maker. In fact, this edge represents the core of MCDM acceptance research. Of course, this research includes not only the acceptance of tools but also the acceptance of MCDM methods. The

method itself, however, is used (and perceived) only via the MCDM tool which implements the method. The same line of arguments also applies to the relationship between the MCDM method and the decision problem (task) at hand. While the MCDM method supports solving the decision problem for sure, this relationship is conveyed by the tool implementing the method and assisting the solving of the task. Thus, there is only an arrow, from MCDM tool to decision problem, but no arrow from MCDM method to decision problem. The edge “performs” reflects the original problem situation or motivation for the usage of most IS, that is an individual has to carry out a specific task. While the relation between the decision maker and the MCDM tool as well as the relation between the decision maker and the decision problem are shown as activities by the decision maker (“uses” and “performs”), the groups facilitating conditions and social factors influence the decision maker (edges pointed towards DM). The edge labeled “facilitates” accounts for the need of certain enabling resources for some MCDM tools.

In the following we will give a short definition of our understanding of each group along with a presentation of key concepts and exemplary constructs.

#### A. Individual

The group of individual characteristics covers relevant aspects of the individual (decision maker) that influence the willingness to use a technology. This covers a quite wide range of factors like personality traits, demographic criteria, abilities, knowledge and affects.

For example, individual characteristics like *computer self-efficacy* (beliefs of being able to perform a specific task by using an IT system) or *computer playfulness* (describing the degree of cognitive spontaneity in microcomputer interaction) have been found to show a significant effect on perceived ease of use and therefore on technology adoption behavior (TAM3, [6]). Furthermore, constructs like *age*, *gender* or *experience* (moderating the individual’s usage behavior) establish this group (UTAUT, [15]).

From the five factor model’s point of view (FFM [27]) there are five individual traits that represent a personality in a highly aggregated manner: *openness*, *conscientiousness*, *agreeableness*, *extraversion* and *neuroticism*. Combined with general models of technology acceptance, the “big five” have been used to show that the personality traits of an individual have a significant influence on the willingness to use a certain technology [28].

Since research on the individual is part of various academic disciplines, many other models and researchers contributed characteristics to this group. For example, *attitude* (TRA, [4]), *affect* ([29]) or *propensity to trust* (Integrative Model of Organizational Trust, [30]) are constructs which are summarized within this group.

#### B. Task

The group of task-related constructs covers relevant aspects of the task (decision problem) at hand, which effect the

user’s evaluations of technologies intended to support him in performing the task.

For example, *task difficulty* (non-analyzable search behavior) and *task variety* (number of exceptions) distinguish routine tasks from non-routine task. A prominent model, which is based on this characterization of tasks, is the task-technology-fit model (TTF, [14]). Based on former research, this model also characterizes tasks by their *task interdependence* (with other organizational units). The TTF states that the more users are engaged in non-routine and interdependent tasks the more they demand from the technology, which in turn leads to lower evaluations of the respective technology. The model further states that this lower evaluations will effect the perception of usefulness and thus the utilization of the technology.

A behavioral model focusing on decision making, which is based on task characteristics, is the effort-accuracy model of cognition [19]. Within this model, decision tasks are characterized by their complexity which increases with constructs like *number of alternatives* or *number of dimensions*. The effort-accuracy model of cognition states that the complexity of the decision problem has a significant influence on the decision strategy used by decision makers [31]. Thus, we argue that a DSS which does not provide decision strategies (MCDM tool) appropriate to the complexity of the decision problem at hand, is not likely to be used.

Besides these basic models, task-related characteristics are subject to active research. For instance, the *risk* inherent to a task can affect the willingness to delegate the task to others, which also might be true for technologies [30], [16]. Another example is the degree to which a DM is *accountable* for the decision, which also influences the behavior of the decision maker [19]. We subsume such and similar abstract properties of tasks under this category of task-related characteristics, and argue that these characteristics have a major influence on the perception of the system’s usefulness.

#### C. Technology

The group of technology-related characteristics covers relevant aspects of the IT-artifact influencing the individual’s willingness to use the respective IT-artifact (MCDM tool).

For example, *visualization* capabilities can be regarded as one key-characteristic of a technological system and is therefore subject to active research. Following a long discussion on whether to prefer graphical vs. tabular representations, Vessey proposed the theory of cognitive fit (CFT, [11]) to integrate the different perspectives on which type of visualization fits to different types of data and task (spatial vs. symbolic). It states that a picture is not always worth a thousand words but in fact hinders cognition when used for the wrong purpose. Based on CFT, Speier found that visual representations not only have to fit the underlying type but also the complexity of the task [22].

*Social presence*, to name another construct, states that humans frequently apply social norms and rules towards computers. Nass, Steuer and Tauber presented this new paradigm called computers are social actors (CaSA, [13]) and triggered a

series of research attempts to investigate on how to increase or decrease the perceived social presence of computers in various fields of application (e.g. e-learning software, [32]).

Among a number of other constructs, we found that *job relevance* (degree of fit between technology and task [5]), *explanation facilities* (integration of how, why and process explanations into the software [33]) or *process guidance* (active guidance through the complete decision process [34]) belong to this group as well. Following the understanding that these characteristics of an IT artifact carry the potential to influence the degree of acceptance substantially, we subsume these factors within the group of technology-related influences.

#### D. Method

The group of methodical influences covers relevant aspects of the MCDM method influencing the individual's willingness to use the MCDM tool at hand. This group is a special case of technology related factors which can be distinctly attributed to the MCDM method underlying the respective technology.

For example, constructs like the *decision strategy* and perceived *decision strategy restrictiveness* and their respective influence were investigated by Wang and Benbasat on the basis of the effort-accuracy framework of cognition [16]. Their results showed that the more a user perceives a decision aid as being restrictive regarding the freedom to apply their preferred decision process the lesser is the user's intention to use the DSS.

Kotteman and Davis, on the other hand, found evidence in the literature that the degree of *decisional conflicts*, which increases with the salience of trade-offs, has a direct influence on the failure or success of decision support systems. They conclude that prominent constructs like perceived usefulness are not suitable indicators for actual performance of a DSS [35].

The constructs belonging to this group have an important influence on the acceptance of a DSS, since the user has to accept both, the decision strategy and its implementation (see Section II-B). Following this line of reasoning, we separate factors which can be attributed to the method from those that are attributed to the tool. We bear in mind, however, that this segregation is mainly conceptual since both groups are highly interconnected in empirical settings.

#### E. Social influence

The group of social influence covers relevant aspects of the social system influencing the decision maker's willingness to use the technology at hand.

For example, *subjective norm* (the degree to which an individual perceives social pressure to perform or not perform a behavior) is a major influence of the social system on the individual's behavior. Beside individual factors, *subjective norm* has been used within TRA and its successor, the theory of planned behavior, to explain intention to perform a behavior [36], [4].

*Image* (the degree to which an individual believes that using the technology will enhance one's social status) is another construct which falls into the category of social influence. Among other constructs, image and subjective norm have been integrated into TAM2 to explain perceived usefulness [5]. TAM2 accounts for the relatedness of image and subjective norm by pointing out that image is partly determined by subjective norm.

By incorporating the group of social factors, we acknowledge that individuals are always part of a social system which significantly influences their behavior and thus their technology usage.

#### F. Facilitating conditions

The group of facilitating conditions represent the organizational and technical support that is available to a decision maker or tool for the usage of a technology.

For example, *perceptions of external control* as proposed in TAM3 is used in a similar way, expressing the degree to which an individual believes that organizational and technical resources exist to support the use of the system [6]. Taylor and Todd propose further constructs in the decomposed TBP, like *resource facilitating conditions* (regarding beliefs about the availability of general resources such as time and money) and *technology facilitating conditions* (regarding technology compatibility) [37]. The construct of *end user support*, introduced in the work of Igarria and Iivari [38], also suggests that organizational support for using a system can enhance acceptance.

Although the existence of facilitating conditions is not necessarily a prerequisite for general MCDM acceptance, these factors can directly influence the individual's perception of the technology. Hence, consistent with our findings, *facilitating conditions* have already been presented as a highly-aggregated factor in UTAUT [15].

Table I summarizes the assignment of constructs from the respective models/theories to the categorization proposed. It can be observed that while some models pursue a comprehensive approach and hence integrate constructs from many categories, others specialize on certain areas.

## IV. CONCLUSION AND FURTHER RESEARCH

The objective of this study was to conceptualize a structural model that reflects the various research areas and perspectives on decision support acceptance. The model-building followed an extensive literature survey in the area of technology acceptance with special emphasis on MCDM acceptance. We could identify six major groups of influencing criteria that were put into context using a graphical representation. This conceptualization is consistent with former research results. For example, the UTAUT model also incorporates social influence, facilitating conditions and individual characteristics (the latter split into multiple detailed characteristics) as major determinants of technology adoption behavior. Furthermore,

TABLE I  
SAMPLE MAPPING OF MODELS/THEORIES TO FACTOR GROUPS

	individ.*	social	facil. cond.*	technology	method	task
UTAUT	X	X	X	X		
TAM3	X	X	X	X		
TPB	X	X	X			
MPCU	X	X	X	X		
FFM	X					
EAMC					X	X
CFT				X		
CaSA				X		
TTF	X			X		X

\*individ. = individual, facil.cond. = facilitating conditions

UTAUT=Unified Theory of Acceptance and Use of Technology, TAM3=Technology Acceptance Model 3,

TPB=Theory of Planned Behavior, MPCU=Theory of PC Utilization, FFM=Five Factor Model,

EAMC=Effort-Accuracy Model of Cognition, CFT=Cognitive Fit Theory, CaSA=Computer are Social Actors,

TTF=Task-Technology Fit

the TTF model is based on similar groups of characteristics (task, technology and individual) to explain IS utilization. That there is some agreement on major factors driving technology acceptance in general and MCDM acceptance in particular is a promising result towards a more unified view of and research in technology acceptance.

However, our findings also show that most research focuses on the individual's perception and related behavioral consequences. Rather little effort has been put into the analysis of the actual IT-artifact and how its characteristics influence its perception of the user. This also holds true for other drivers of technology acceptance. For example, questions like how to design user support services to increase the users' perception of facilitating conditions do not receive much attention although they have the potential to substantially increase user acceptance of technologies in the practical field. Thus, the analysis of how the design of concrete artifacts influences user evaluations seems to be a promising area for further research.

#### ACKNOWLEDGEMENT

This research has been funded by the Austrian Science Fund (FWF): project number TRP 111-G11.

#### REFERENCES

- [1] P. Mayring, "Qualitative content analysis," *Forum: Qualitative social research*, vol. 1, no. 2, 2000. [Online]. Available: <http://nbn-resolving.de/urn:nbn:de:0114-fqs0002204>
- [2] W. Orlikowski and C. Iacono, "Desperately Seeking the "IT" in IT Research: A Call to Theorizing the IT Artifact," *Information Systems: The State of the Field*, pp. 19–42, 2006.
- [3] F. Davis, R. Bagozzi, and P. Warshaw, "User acceptance of computer technology: a comparison of two theoretical models," *Management science*, vol. 35, no. 8, pp. 982–1003, 1989.
- [4] M. Fishbein and I. Ajzen, *Belief, attitude, intention, and behavior: an introduction to theory and research*. Addison Wesley Publishing Company, 1975.
- [5] V. Venkatesh and F. Davis, "A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies," *Management Science*, vol. 46, no. 2, pp. 186–204, 2000.
- [6] V. Venkatesh and H. Bala, "Technology acceptance model 3 and a research agenda on interventions," *Decision Sciences*, vol. 39, no. 2, pp. 273–315, 2008.
- [7] W. DeLone and E. McLean, "Information systems success: the quest for the dependent variable," *Information systems research*, vol. 3, no. 1, pp. 60–95, 1992.
- [8] W. DeLone and E. McLean, "The DeLone and McLean model of information systems success: A ten-year update," *Journal of management information systems*, vol. 19, no. 4, pp. 9–30, 2003.
- [9] A. Bandura, *Social Foundations of Thought and Action: A Social Cognitive Theory*. Prentice Hall, 1985.
- [10] F. Davis, R. Bagozzi, and P. Warshaw, "Extrinsic and intrinsic motivation to use computers in the workplace," *Journal of Applied Social Psychology*, vol. 22, no. 14, pp. 1111–1132, 1992.
- [11] I. Vessey, "Cognitive Fit: A Theory-Based Analysis of the Graphs Versus Tables Literature\*," *Decision Sciences*, vol. 22, no. 2, pp. 219–240, 1991.
- [12] I. Vessey and D. Galletta, "Cognitive fit: An empirical study of information acquisition," *Information Systems Research*, vol. 2, no. 1, pp. 63–84, 1991.
- [13] C. Nass, J. Steuer, and E. Tauber, "Computers are social actors," in *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence*. ACM, 1994, pp. 72–78.
- [14] D. Goodhue, "Understanding user evaluations of information systems," *Management science*, pp. 1827–1844, 1995.
- [15] V. Venkatesh, M. Morris, G. Davis, and F. Davis, "User acceptance of information technology: Toward a unified view," *MIS quarterly*, pp. 425–478, 2003.
- [16] W. Wang and I. Benbasat, "Interactive Decision Aids for Consumer Decision Making in E-Commerce: The Influence of Perceived Strategy Restrictiveness," *MIS Quarterly*, vol. 33, no. 2, pp. 293–320, 2009.
- [17] M. Limayem and G. DeSanctis, "Providing decisional guidance for multicriteria decision making in groups," *Information Systems Research*, vol. 11, no. 4, pp. 386–401, 2000.
- [18] T. Chenoweth, K. Dowling, and R. St Louis, "Convincing DSS users that complex models are worth the effort," *Decision Support Systems*, vol. 37, no. 1, pp. 71–82, 2004.
- [19] J. Payne, J. Bettman, and E. Johnson, *The adaptive decision maker*. Cambridge Univ Pr, 1993.
- [20] I. Benbasat and P. Todd, "The effects of decision support and task contingencies on model formulation: A cognitive perspective," *Decision Support Systems*, vol. 17, no. 4, pp. 241–252, 1996.
- [21] P. Todd and I. Benbasat, "Evaluating the impact of DSS, cognitive effort, and incentives on strategy selection," *Information Systems Research*, vol. 10, no. 4, pp. 356–374, 1999.
- [22] C. Speier, "The influence of information presentation formats on complex task decision-making performance," *International Journal of Human-Computer Studies*, vol. 64, no. 11, pp. 1115–1131, 2006.
- [23] E. Bernroider, N. Obwegeser, and V. Stix, "Introducing complex decision models to the decision maker with computer software - the profile distance method," *Journal of Systemics, Cybernetics and Informatics*, vol. 8, no. 3, pp. 24–28, 2010.
- [24] M. Gonul, D. Onkal, and M. Lawrence, "The effects of structural characteristics of explanations on use of a DSS," *Decision Support Systems*, vol. 42, no. 3, pp. 1481–1493, 2006.
- [25] M. Silver, "Decisional guidance for computer-based decision support," *MIS Quarterly*, pp. 105–122, 1991.
- [26] S. Gregor and I. Benbasat, "Explanations from intelligent systems: Theoretical foundations and implications for practice," *MIS quarterly*, pp. 497–530, 1999.
- [27] P. Costa and R. McCrae, "The NEO Personality Inventory manual," *Psychological Assessment Resources*, Odessa, 1985.
- [28] A. Sharma and A. Citrus, "Incorporating Personality into UTAUT: Individual Differences and User Acceptance of IT," in *Proceedings of the Americas Conference on Information Systems*, 2004, pp. 3348–3353.
- [29] H. Triandis, "Values, attitudes, and interpersonal behavior," in *Nebraska Symposium on Motivation*, vol. 27, 1980, pp. 195–259.
- [30] R. Mayer, J. Davis, and F. Schoorman, "An integrative model of organizational trust," *The Academy of Management Review*, vol. 20, no. 3, pp. 709–734, 1995.
- [31] J. Payne, "Task complexity and contingent processing in decision making: An information search and protocol analysis\* 1," *Organizational behavior and human performance*, vol. 16, no. 2, pp. 366–387, 1976.

- [32] F. Tung and Y. Deng, "Designing Social Presence in e-Learning Environments: Testing the Effect of Interactivity on Children." *Interactive learning environments*, vol. 14, no. 3, pp. 251–264, 2006.
- [33] J. Dhaliwal and I. Benbasat, "The use and effects of knowledge-based system explanations: theoretical foundations and a framework for empirical evaluation," *Information Systems Research*, vol. 7, no. 3, pp. 342–362, 1996.
- [34] Y. Siskos and A. Spyridakos, "Intelligent multicriteria decision support: Overview and perspectives," *European Journal of Operational Research*, vol. 113, no. 2, pp. 236–246, 1999.
- [35] J. Kottemann and F. Davis, "Decisional Conflict and User Acceptance of Multicriteria Decision-Making Aids\*," *Decision Sciences*, vol. 22, no. 4, pp. 918–926, 1991.
- [36] I. Ajzen, "The theory of planned behavior," *Organizational behavior and human decision processes*, vol. 50, no. 2, pp. 179–211, 1991.
- [37] S. Taylor and P. Todd, "Understanding Information Technology Usage: A Test of Competing Models," *Information Systems Research*, vol. 6, no. 2, pp. 144–176, 1995.
- [38] M. Igbaria and J. Iivari, "The effects of self-efficacy on computer usage," *Omega*, vol. 23, no. 6, pp. 587–605, 1995.



# A Context-Aware Mobile Accessible Electric Vehicle Management System

Nils Masuch, Marco Lützenberger, Sebastian Ahrndt, Axel Heßler, and Sahin Albayrak  
 DAI-Labor, Technische Universität Berlin  
 Faculty of Electrical Engineering and Computer Science  
 Ernst-Reuter-Platz 7, 10587 Berlin, Germany  
 Telephone: +49 (0)30 - 314 74000, Fax: +49 (0)30 - 314 74003  
 Email: {firstname.lastname}@dai-labor.de

**Abstract**—In the coming years, the German traffic situation will undergo a challenging addition. Major car manufacturers have scheduled the year 2013 as cutoff for electric mobility. Yet, current studies indicate that range limitations and insufficient charging infrastructure endanger the acceptance for electric vehicles (EV). This is regrettable, and not only for the producer, but also for less obvious parties such as local energy providers which consider electric vehicles as remedy to one of their most severe problems of managing the grid load balance. In this paper we introduce a mobile accessible EV management system which accounts for the mobility of the user and also integrates web-based (commercial) services of third parties. Our objective is to counter the limitations of electric mobility and also to facilitate all of its (business) perspectives. We want to render electric mobility a success and support its trendsetting character.

**Index Terms**—Mobile environments, Mobile commerce, Distributed system, Web-based services, Electric vehicles, Charging station

## I. THE AGE OF ELECTRIC MOBILITY

IN THE coming years the traffic situation on German roads will undergo a ground-breaking, futuristic, and yet ambiguous change. Major car manufacturers have scheduled the year 2013 as cutoff for electric mobility. This ambitious aim is facilitated by the German government, which proposes the magic number of one million electric vehicles on German roads by the year 2020 [1]. However, to the day, the optimism regarding the acceptance of electric mobility is most often cushioned by results of market analysis. One of these studies [2] has been performed by *Ernst & Young* most recently. Figure 1 illustrates the answers to one of the study’s key question: “Which factors would make you most hesitant to choose a Plug-in Hybrid Electric Vehicle (PHEV) or EV as your next vehicle?”.

The presented numbers imply that, e-mobility is up against a set of severe problems. Beside the pricing issues, potential buyers are mainly scared by the limited range of electric vehicles and also by limited access to charging capabilities. Both problems necessitate considerations and planning of each intended ride. To support a driver in this task, we have developed an automated management system which operates on the daily schedule of the user.

Based on the location, the timing and the priority of the scheduled appointments, the system computes a fitting

		Access to charging stations	Price	Battery driving range	Reliability/service ability	Performance and handling
China		69%	57%	73%	64%	57%
Japan		60%	73%	43%	36%	35%
US		75%	74%	75%	57%	49%
Europe	France	74%	63%	81%	26%	46%
	Germany	74%	66%	75%	46%	52%
	Italy	64%	62%	62%	42%	54%
	UK	71%	60%	71%	47%	50%
<b>Average</b>		69%	67%	66%	49%	48%

■ Highest response rate for each factor    ■ Lowest response rate for each factor

Fig. 1. Answers to the question: “Which factors would make you most hesitant to choose a PHEV or EV as your next vehicle?”, raised within an *Ernst & Young* study [2].

“driving-strategy”, proposes alternative charging intervals, and is also able to recommend optimised appointment sequences by rearranging less prior, non time-critical tasks. The computation also accounts for infrastructural conditions and proposes charging intervals only in close distance to charging capabilities. We applied a web-based approach for the management of the user’s appointments and allow access not only from desktop computers but also from mobile devices in order to facilitate the system’s flexibility and to ensure its application to real world situations.

However, the domain of electric vehicles is complex and not only affects the driver, but also opens new business perspectives and opportunities. Energy providers for example have high hopes in electric mobility and consider EVs as remedy for one of their most critical problems: Grid load balance. By using electric vehicles as “rolling batteries”, energy providers currently develop area-wide mechanisms to charge electric vehicles when there is little grid load and much energy (and preferably a lot of volatile energy) available, and avoid charging periods when there is large demand from the energy grid. Yet, having in mind the restrictions of EVs, it is obvious that an according mechanism has to account for the

driver and his intended rides. For the reason of data integrity, it is also clear, that scheduled appointments undergo the privacy of the driver and cannot directly be forwarded to energy providers for optimisation purposes. We have designed our management system to act as loosely coupled middleware, able to comprehend many input channels, such as schedules provided by users, priority signals provided by an energy provider, infrastructure availabilities provided by energy infrastructure providers and many more.

We provide mobile access and allow users to manipulate their schedule at any time, each addition comprising re-optimisation. Previously computed charging intervals are again measured against the provided priority signal and possibly shifted to more effective time slots. Of course the shifting of already advised charging intervals can only be interesting for the energy provider to regulate his grid load. For the driver there seems to be no apparent benefit, yet, having dynamic energy tariffs in mind, the appeal becomes more obvious. With special conditions for the beforehand booking of particular charging intervals, customers may profit monetarily, while energy providers gain advantage in grid load regulation.

To sum up, electric mobility faces a lot of challenges, but also opens a lot of business perspectives. With our scheduling system we aim to do both, counter limitations and facilitate further possibilities. In this paper we describe the principle of our scheduling system and show how we designed the application to access web-based energy provider services (see Section II). To clarify our approach and to evaluate or work we use an exemplary scenario in which we also motivate the necessity for mobile access (see Section III). Subsequently we will classify and distinct our approach from related works (see Section IV). Finally, we wrap up with a conclusion (see Section V).

## II. SUPPORTING THE DRIVER

The challenge of managing the usage of electric vehicles and their related limitations and potential benefits is a complex one that is dependent to different actors with sometimes conflictive interests. Further, due to the nature of each scheduling problem, the system has to be highly dynamic in order to adapt its decisions to changes in the environment. In the following we will first describe the structure of our management system approach followed by a definition of the distributed services that are relevant within our domain. Further the charging management approach will be described in detail and finally we point out the mobile aspects of our system and its interaction with third party services.

### A. The management system approach

In order to support the driver comprehensively regarding the management of charging issues while utilising an electric vehicle, the management system requires access to various resources providing information about charging options, preferences and necessities. More specifically these can be prognosticated driving patterns, energy progression curves, energy price curves, wind energy curves and infrastructural

availabilities. Such a distinct set of data types is typically not provided by a single party or company and therefore the different information parts have to be requested from multiple sources. In some cases – for example the energy price curve – it might be even reasonable to request the data from different providers in order to build up a larger amount of planning options.

The major challenge for the management system lies in the creation of a charging schedule for the driver's EV out of the provided information. Thereby the result has to satisfy the driver's standards. But how is the driver's standards be defined?

The biggest issue in that context is certainly the mobility warranty in terms of the driver's driving patterns. This means the driver wants to travel with his EV without any sorrow about the state of charge. Out of this reason the management system undertakes the task of selecting a charging interval within the day guaranteeing that the vehicle will have enough energy until then. At the same time, however, the system has to validate that at the place, where the driver aims to be during a potential charging interval, there can be found a respective charging station nearby, which has to be available furthermore. This is why the management system not only searches for charging intervals, but at the same time for concrete charging stations. If the properties of the station are appropriate our approach triggers the reservation of the desired time slots. Another aspect are the energy costs that arise for charging transaction. It can be assumed that in the near future the energy provider will propose variable power prices in order to regulate the demand according to their infrastructural needs. Doing so, the purchase of expensive regulating energy can be avoided and regenerative energy, with the disadvantage of unsteady production, can be adapted monetarily to the demand and is therefore more attractive even at unusual energy consumption times. In that context it makes sense to the driver of an EV to fall back on low priced time slots in times he is flexible. Therefore the charging management has to select a solution that fulfills either the mobility warranty as well as the monetary aspect with flexible weightings on both parameters depending on the driver's profile.

However, in order to be able to trigger the charging management process the management system must find a way to evaluate the energy progression curve of the EV's battery for a certain prospective time interval. This is in turn only possible, if some kind of driving pattern can be assumed. In our approach we decided to deduce the expected journeys out of the driver's personal calendar. Doing so, the appointments and its location information are being extracted in order to know where to drive to. In order to compute the journeys in an automated way we defined a tag pattern within the appointments location field. If the driver writes down only one location the system assumes that this is the target place and the journey will start from a predefined standard location. But if two locations are being named within the field, the system computes a route from the first to the second named location. Based on that, the system is able to build up a prospective en-



ergy progression curve and check whether additional charging events are necessary or not.

### B. It's all about services

Due to the high level of distribution, which evolves from the different kinds of services being invoked, our management system is based upon a modular, service-oriented approach, which can be extended anytime with additional functionalities.

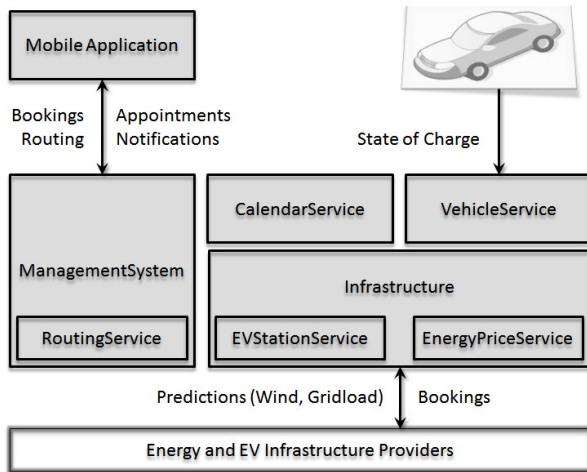


Fig. 2. The electric vehicle management system and related services

Figure 2 illustrates its structure and the external services which are being accessed during runtime. In the following the services and its functionalities are being described shortly.

- **Calendar Service:** The calendar service accesses a user-defined calendar resource and requests all prospective appointments from it. Furthermore the calendar service offers the possibility to easily add additional appointments to the source calendar.
- **Routing Service:** Proprietary service that provides the computation of routes regarding different preferences, such as length and speed. The result is the fundament for the computation of the energy progression curve.
- **Vehicle Service:** Wraps two separate services, namely the energy consumption service and the vehicle state service. The energy consumption service relies on the energy profile of the vehicle's battery and computes the energy consumption of routes in dependency to the expected state-of-charge. The vehicle state service is able to provide dynamic information like the actual SoC and position to the requester.
- **Charging Station Service:** Provides information about existing charging stations of a specific supplier near a requested location. Furthermore, offers information about the technical properties of the charging station and enables to book the charging station access for a certain time interval. According to this a charging station availability check is also possible.

- **Energy Price Service:** Provides a variable, time-dependent energy price curve of the energy provider, which is being continuously.

### C. Charging interval selection

The charging management process is triggered by the management system after certain events. These can be the appearance of new appointments (initial planning) or the dynamic and significant change of the energy provider's price function (potential rescheduling). In these cases the charging management behaves as follows.

The starting point for the algorithm is the prospective energy progression function of the EV, deduced by the driver's appointments. By means of that, it is checked whether the EV's SoC is expected to fall below a predefined, fixed threshold somewhere in the future. If so, the system extracts all time slots, where the EV is being parked before the threshold violation will occur and where a potential charging event will lift the curve above the threshold. Each of the possible charging options is now analysed regarding the location of the driver at that time and whether there is a charging station located nearby. At the same time, the charging station related energy price curve is being requested and checked for the prices with the desired interval. Out of these information, a set of potential charging options is generated, sorted by the cheapest charging station - energy price combination. Finally the system selects the cheapest charging interval that is available for the current price conditions. Since the price curve might change for the selected time interval dynamically, we defined in our approach that the energy provider sells the energy at maximum for the current price and might be even cheaper if the curve is being corrected downwards. Afterwards the driver is being informed about the updated charging planning via the mobile accessible application and after a confirmation the necessary bookings will be performed.

### D. Mobile user control

In our approach the user has two possibilities to control and interact with the management system. On the one hand, he is able to do that indirectly by inserting appointments in his personal calendar, which leads to a deduction of driving necessities and therefore to a triggering of the charging management process. On the other hand he can interact dynamically via a mobile EV management application. Doing so, the mobile application enables the user at every time to request the current planning state, to configure it and to initiate and control replanning actions due to dynamic events.

So the mobile application provides the configuration of user preferences, which enables the driver to orient the charging management more on the monetary aspects or the absolute mobility warranty. Further, a list of energy providers and their respective charging station infrastructure the system is planning on can be selected as well as a definition of a standard location the route planning is computing the journey from by default. Each modification within the user preferences leads to a rescheduling process.

In addition the user can look at his appointments in a calendar view. Within this perspective not only the synchronised appointments from the user's personal calendar are displayed but also the evaluated journeys and charging events. If the driver wants to start a journey he can select the appropriate one from the calendar, whereupon a route overview is being opened. Furthermore, it is possible to add new appointments directly within the mobile application calendar, which are finally synchronised to the user's background calendar.

The driver is also able to check the current state of his electric vehicle, if an activated vehicle state service is installed within the car. This offers the possibility being always informed about the current state of charge, even when not being at the vehicle's location. Therefore driver keeps a total overview about the processes that are happening within the car, for example when the car is charging and the user thinks about leaving the charging station a bit earlier than initially planned.

A very important task for the mobile application is the completion of charging booking decisions by interacting with the user. After the system has evaluated the optimal charging slot based upon the user preferences the driver gets an inquiry for confirmation. In that situation the driver can overview the costs for that event and compare them to alternate options. If he confirms a booking for the selected charging station time interval combination is being booked. Alternatively the driver can select other charging options if he is not satisfied with the system's decision. Consequently the driver is always able to fully control the planning of charging events and can also define charging events without any request to the charging management module.

Another aspect, which is covered by the mobile application, are the interaction mechanisms between user and system when unexpected events occur. These can be initiated either by the system or the user. So, for example, the driver is always able to cancel a booking if he recognises that he will not be at the place for the defined charging interval. In other cases the system initially informs the driver about some relevant state changes (unexpected, increased energy consumption, cheaper price curve, malfunctioning charging station) and proposes alternatives to him. For example, let us regard the price curve which changes during time. If some charging slots have been already booked and the system recognises that the price curve is now cheaper during another, later charging slot option than actual selected one, the user gets informed about that dynamically and can decide whether to change his booking or not.

Therefore the mobile management application can be described as a comprehensive interface to the user, which enables, besides configuration of the management system, the access to information about the energy providers infrastructure and the performance of usage rights transactions.

### III. EVALUATION

In order to evaluate our approach of a mobility management system with a mobile application for user system interaction

regarding the user's benefits, we describe a typical scenario in the following, where our application is intended to be used. In particular, we compare the differences to a static charging management system which does not provide *Mobile Commerce* access to energy providers and finally elaborate the advantages of our mobile solution.

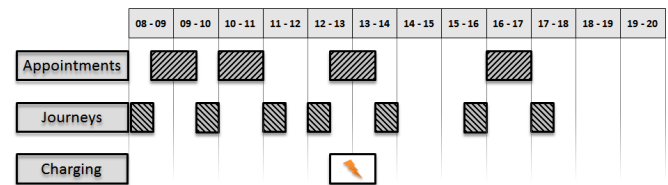


Fig. 3. The driver's appointment schedule and the initial journey and charging event results.

Figure 3 illustrates the appointment schedule of an electric vehicle holder. Taking the appointments into account the management system has deduced all relevant journeys and computed its expected energy consumptions (with regard to the specific vehicle characteristics). Out of it a time dependent energy progression curve for the specific day and without any charging events in it yet has been build up.

Based on these information the charging management algorithm analyses when a charging event has to be scheduled at latest in order to adhere the mobility warranty and not to undercut a predefined lower energy threshold. Doing so, different time intervals come into play for charging. In our case the undercut of the energy threshold without charging occurs during the journey from 13:30 to 14:00 h. In this respect all parking events before that journey are left for consideration as long as there exist charging stations nearby the respective appointment locations. Now the energy price curve is relevant for further decisions. Fig. 4 shows the curve for our scenario indirectly. In the foreground there are two curves, one of them representing the overall load within the energy network, the other the amount of wind energy. In our case both curves are the fundament for the energy price curve, which is represented by the different types of shade in the background. Darker intervals indicate expensive charging slots while bright ones are cheap.

The charging management now computes the quality of the remaining, potential charging events in consideration of the mobility warranty and the energy price curve. In our scenario the algorithm evaluates a charging event during the meeting from 12:30 to 13:30 h as optimal. When looking at the energy price indicator in the figure, it is obvious that these interval is brighter than all the other potential intervals before. Another aspect the algorithm looked at when selecting a charging event is the distance from the charging station to the appointment location. If it exceeds a certain length the solution option is discarded, which was not the case in our scenario. If nothing changes on the user's appointment schedule until the start of the first journey the charging management is finished after the confirmation of the user, which triggers the system to

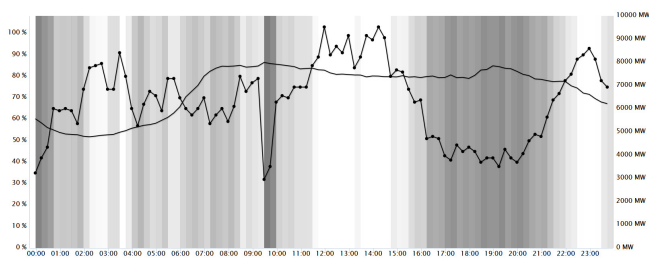


Fig. 4. Absolute amount of regenerative energy (line with dots; in MW) within the grid and its load. The background colors indicate the deduced energy price signal (Bright intervals represent cheap prices, dark intervals expensive ones)

communicate with the energy provider and to book the selected time interval at the actual conditions.

Until now, the management system has solely performed an initial charging event planning. In reality, however, cases will occur when the initial planning is not sufficient anymore and the user has to be involved in an interaction process. This will be shown in the following.

In our scenario the user starts his day and drives with his electric vehicle to his first meeting without any charging event being scheduled for that time. At 09:30 h he drives to the next appointment. Shortly before arriving, the appointment at 16:00 h is canceled by the organiser. The management system triggers the charging management process, which computes again the energy progression curve for the day and recognises, that, of course, no changes in the charging planning has to be done due to the omission of the journeys for the canceled meeting. But at the same time the management system checks for a energy price curve and notices a significant difference to the old price curve. Meanwhile, especially the prices at the current scenario time are much cheaper than our booked charging interval at 12:30 h. Therefore the system checks whether it is possible due to the omission of the late scheduled journeys to charge already earlier without triggering an energy threshold violation. Since this is the case in our scenario the charging management system searches for a charging station of the same energy provider nearby the immediately impended appointment. The system contacts the driver via his mobile application and offers him to shift his charging event to now and to directly drive to the evaluated charging station (see Figure 5). As a motivation the application displays the monetary benefit the user has when charging right now in comparison to the initial booked time slot. The driver confirms and drives immediately to the proposed charging station. When arriving he notices that the charging station parking lot has been illegally occupied by another vehicle and he is therefore not able to charge there. Because of that the user opens his mobile application and requests the cancellation of his booking and the search for an alternative charging station nearby. After processing the cancellation, the system checks for different charging stations and finds one from a different energy provider, which is available and has just a slightly

more expensive energy rate than the occupied one. The mobile application informs the driver and shows the route to the aimed spot, whereby the user confirms the booking and drives to the charging station. After the arrival, the user authenticates itself with his personal RFID chip and the charging station opens the loading hatch. After connection is set up and the charging has started the user walks the few hundred metres to the appointment location.

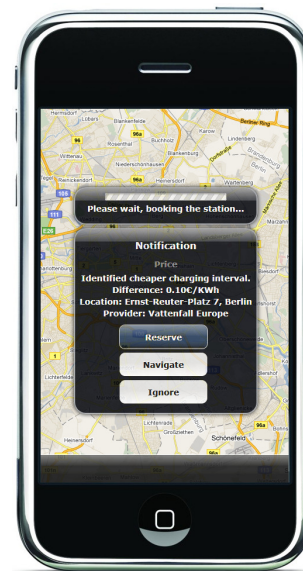


Fig. 5. Screenshot of our mobile application. Offering the driver to shift a charging event to now due to cheaper price.

The above described scenario shows in several respects how mobile interaction can serve the user when utilising electric vehicles in daily life. Especially in situations, when events occur very shortly like the cancellation of an appointment or the change of the energy price curve, the charging management can propose the result of his replanning process dynamically to the user, which can lead to a monetary benefit or simply keep the mobility warranty up. Using a static system, where the driver is just able to check the plans for example at the computer at his working place, such use cases can not be handled. This shows that a static system collapses every time when changes occur shortly, since they cannot be communicated between user and system. Within the scenario the driver is in the unpleasant situation that the aimed charging station, though booked, is physically not reachable. With the help of the mobile application the user was able to directly request a solution from the management system. In a static system the user would have had to search for an alternative charging station by himself which leads to significant additional expenses. In conclusion, the usage of dynamic, mobile management applications with an integration of *Mobile Commerce* services in the context of electric mobility leads to an significant additional benefit for the driver by supporting him comprehensively at every time and every place.

#### IV. RELATED WORK

It is difficult to provide a clear structure to this section since our work touches many different topics and domains of research. While the distributed nature of our system is to be assigned to the field of agent-oriented software development, the introduced scheduling and optimisation algorithms are unmistakably related to the comprehensive realm of operations research.

However, what our application clearly offers is mobile access to services which simplify, facilitate and support the management of electric vehicles. These services are web-based and in the case services provided by energy providers, feature a commercial character, since “rights to use services” are transferred.

Literature refers to this principle as *Mobile Commerce* for which *Tiwari* and *Buse* provide the following definition:

“*Mobile Commerce is any transaction, involving the transfer of ownership or rights to use goods and services, which is initiated and/or completed by using mobile access to computer-mediated networks with the help of an electronic device.*” [3]

Compared to the more common and superordinate domain of *Electronic Commerce* (or *E-Commerce*), *Mobile Commerce* emphasises on services which can be accessed at any time and from anywhere. As a matter of fact, *Mobile Commerce* accounts for an entire set of promising capabilities, such as location- and context-awareness.

In the following we present driver assistance systems which are similar to the one we have developed. In this survey, we try to answer the question in how far principles of *Mobile Commerce* have been applied for this type of application.

The *Blink Mobile Application* [4] is a mobile app from ECOItality<sup>1</sup> introduced at the Electronic Drive and Transportation (EDTA) conference<sup>2</sup>. It allows to access the Blink Network<sup>3</sup> anywhere and will be soon available for free via all major mobile device application stores. The Blink Network is an EV charging infrastructure including EV charging stations, software and online services. The mobile application allows to find available public and commercial charging stations. This can be done location-based via the mobile device’s wireless or GPS coordinates or manually. Furthermore it allows to reserve charging stations, to receive charging status updates, to view additional informations about the charging station and to route to the chosen station. Nevertheless, the Blink Mobile Application do not cover all functions of the Blink Network. So it is not possible to use the scheduling function of the network from mobile access. However, many mobile devices can still access this feature using the website of the Blink Network. There they user is able to schedule a charge or plan reservations based on travel routes.

The *ChargePoint App* [5],[6] developed by Coulomb Technologies<sup>4</sup> allows to locate Charge Point Networked charging stations. It’s available for Apple iPhone ad BlackBerry. The app allows to find charging stations near to a specified address (US, Europe, Australia), to receive status informations about the stations, to trigger the charging process and to receive real-time notifications of the running process. As the supported devices have the possibility to route, the app can show directions to charging stations, too. Also it can calculate the cost of a charge.

The *PlugShare App* [7],[8] helps EV drivers to find charging stations and homes or lots that will allow them to recharge for free. It is not developed by a EV infrastructure provider like the both introduced before and is driven by the idea to build up a community of people how share there resources (under the term *plug-share*). This is much similar to the couchsurfing<sup>5</sup> project and is driven by the idea that “...*the infrastructure to charge is everywhere.*”[7].

Similar to this approaches there exists several others that allow to find charging stations by web or mobile access. The *Alternative Fueling Station Locator* [9] use Google Maps technology to show locations of EV charging and other alternative fueling stations. The *DriveAlternatives App* [10] does pretty much the same and offers additional features like favorite stations, photos, comments and email alerts for station changes. *CarStations* [11] offers a user driven international directory and mapping service for EV charging stations. Unlike the preliminary introduced approaches, these ones offer no functions to initiate or complete transaction in the purpose of *Mobile Commerce*.

To sum up, none of the examined approach provides dynamic planning which is triggered by environmental or user dependent changes. Nevertheless, the *Blink Mobile Application* combines scheduling features which *Mobile Commerce* aspects. In detail this means that the user is able to plan and reserve charging stations. In contrast to our approach this must be done manually and there is no automated planning behaviour based on variable price signal curves, which supports the user. As with the *ChargePoint App* the customer is able to initiate a charging process, monitor the costs of the running process and stop them if needed. Scheduling is widely missing here. Both applications have been developed by EV infrastructure providers and their extend of *Mobile Commerce* is limited to the manufacturer’s products. The developers of *PlugShare App* present a different approach, which complies with the “rights to use services” specification. Locations for charging electric vehicles are proposed and can be booked free of charge via SMS, email or a call. The cost are actually borne by the provider of the charging station. The approach is based on the idea to provide a community-based backup network in case a driver measures his consumption wrong. No scheduling mechanism is implemented here.

<sup>1</sup>ECOItality, Inc. – <http://www.ecotality.com/>

<sup>2</sup>EDTA – <http://www.electricdrive.org/>

<sup>3</sup>The Blink Network – <http://www.blinknetwork.com/>

<sup>4</sup>Coulomb Technologies, Inc – <http://www.coulombtech.com/>

<sup>5</sup>CouchSurfing – <http://www.couchsurfing.org/>

## V. CONCLUSION

In this paper we introduced a management system for electric vehicles. We motivated the necessity for such an application with the restrictions and limitations of the first EV generation. We also provided figures on the common acceptance towards electric mobility in order to emphasise the severity of its shortcomings and to support our motivation. Subsequently we introduced our management system and its basic functionality of optimising the use of an electric vehicle by accounting for the driver's appointments and additional data from a web-based energy provider service. After outlining the basic idea and functionality we described our implementation in detail and described a simple, yet expressive example in order to evaluate our work and to clarify its principle. Finally we compared our work to related approaches.

### A. Implications

As mentioned above, we used a rather simple scenario to evaluate our work. The idea was to clarify the functionality of our approach and to emphasise our main objective of providing mobility for the driver. However, by integrating the web-service of the energy provider for the optimisation routine we showed that e-mobility provides business perspectives beyond those of the car manufacturer and their customers. In our example both, the driver and the energy provider were able to profit from our management system. While the driver gained profit in a better pricing, the energy provider managed to shift one particular charging interval to a — from his perspective — far better period of time. Admittedly, the benefit of shifting a single charging interval is infinitesimal small, yet, on a large scale the principle is considered a remedy to one of their most severe problems of managing the grid load balance. Currently, there many research projects trying to provide solutions for this exact challenge. In fact, this work was actually done as a part of this initiative [12].

### B. Contribution

The contribution of our approach is twofold. To start with, we have managed to support the driver of an electric vehicle in his daily routine. It was our objective to counter the limitations of electric mobility and based on our evaluation we can say, that we have managed to utilise the capabilities of electric vehicles in a far more effective way. Of course, the reliability of fuel driven cars is a dream of the future, but improvement here rather lies with the car manufacturers and battery producers.

Beside a more effective utilisation, we have shown, that our approach is able to support business perspectives around e-mobility. It is our belief, that the target group of electric mobility exceeds that of common "vehicle customers" and opens perspectives beyond that. We also think, that these services are not explicitly geared towards e-mobility, but also affect regular traffic. For our work we used a prototypical implementation from a national energy provider as an example, but many ground-breaking services are available today already. As an example, consider the many available parkingspot services

(i.e. *Parkingspots*<sup>6</sup>), which are able to retrieve the best fitting parking lot (tariff and location) for a given online query. Another example is the *Waze*<sup>7</sup> service, which provides routing functionality, based on real-time traffic flow. It is our belief that service-based guidance systems as those, mentioned above will establish in the near future. In this work we presented a concrete way to merge the interests of a person with the recommendation of a guidance system.

### C. Future Work

In short, we plan to extend our work to comprehend more third party services.

In some scenarios we had to deal with the problematic that the user's schedule exceeded the range capabilities of the electric vehicle. As an alternative we intend to propose multi-modal strategies to the user — involving public transport. It is our idea, to access online time-tables of a local public transport operator and use the data for our computation and to provide the user with detailed information on the planned journey (line, departure times, estimated arrival times, walking distances, parking lot, etc.).

As a second extension, we plan to integrate a car sharing capability for our system. We want to give users the chance to offer their scheduled rides to others, which are able to check in for desired rides by some mobile interface.

## VI. ACKNOWLEDGMENTS

This work is partially funded by the *Federal Ministry for the Environment, Nature Conservation and Nuclear Safety* under the funding reference number 16EM0004.

## REFERENCES

- [1] The Federal Ministry for the Environment, Nature Conservation and Nuclear Safety, "German federal government's national electromobility development plan," The Federal Ministry for the Environment, Nature Conservation and Nuclear Safety, August 2009.
- [2] J.-F. Tremblay, "Gauging interest for plug-in hybrid and electric vehicles in ley markets," in *Proceedings of the 2<sup>nd</sup> German Electric Vehicle Congress*, June 2010.
- [3] R. Tiwari and S. Buse, *The Mobile Commerce Prospects: A Strategic Analysis of Opportunities in the Banking Sector*. Hamburg, Germany: Hamburg University Press, 2007.
- [4] ECotality, Inc., "ECotality announces blink mobile application for smartphones and mobile devices," ECotality, Inc., April 2011.
- [5] Coulomb Technologies, Inc., "Coulomb technologies announces iphone app for chargepoint networked charging stations," Coulomb Technologies, Inc., February 2010.
- [6] Coulomb Technologies, Inc., "Coulomb technologies chargepoint network releases new mobile app for locating available charging stations," Coulomb Technologies, Inc., March 2011.
- [7] T. Woody, "For electric car owners, a way to share juice," *The New York Times*, New York, NY, USA, March 2011.
- [8] L. Whitney, "Plugshare app finds electric-car charging stations," *Xatori*, Inc., USA, March 2011.
- [9] U.S. Department of Energy's National Renewable Energy Laboratory, "Mobile alternative fueling station locator," U.S. Department of Energy, April 2009.
- [10] DriveAlternatives.com, "Drivealternatives – your green driving solution," [www.drivealternatives.com](http://www.drivealternatives.com) (19.06.2011), October 2009.
- [11] CarStations.com, "Carstations - find your charge," [www.carstations.com](http://www.carstations.com), (19.06.2011).

<sup>6</sup><http://www.parkingspots.com>

<sup>7</sup><http://world.waze.com/>

[12] The Federal Ministry for the Environment, Nature Conservation and Nuclear Safety, "Erneuerbar mobil — marktfähige lösungen für eine

klimafreundliche elektromobilität," The Federal Ministry for the Environment, Nature Conservation and Nuclear Safety, April 2011.



# NotX

## Service Oriented Multi-platform Notification System

Filip Nguyen, Jaroslav Škrabálek

Faculty of Informatics

Masaryk University

Brno, Czech Republic

Email: [nguyen.filip@mail.muni.cz](mailto:nguyen.filip@mail.muni.cz), [skrabalek@fi.muni.cz](mailto:skrabalek@fi.muni.cz)

**Abstract**—This report describes NotX—service oriented system, that applies ideas of CEP and SOA to build highly reusable, flexible, both platform and protocol independent solution. Service oriented system NotX is capable of notifying users of external information system via various engines; currently: SMS engine, voice synthesizer (call engine) and mail engine. Adaptable design decision makes it possible to easily extend NotX with interesting capabilities. The engines are added as plug-ins written in Java. There are plans to further extend NotX with following engines: Facebook engine, Twitter engine, content management system engine. Also the design of NotX allows to notify users in their own language with full localization support which is necessary to bring value in today's market. Most importantly, the core design of NotX allows to run under heavy load comprising thousands of requests for notification per second via various protocols (currently Thrift, Web Services, Java Client). Thus NotX is designed to be used by state of the art Enterprise Applications that require by default certain properties of their external systems as scalability, reliability and fail-over.

**Index Terms**—information system, soa, notx, cep, sms, voice, phone, mail, notification, enterprise, java, active mq, jms, jee, j2ee

### I. INTRODUCTION

NOTIFICATIONS have been studied as valuable tool in context of *ubiquitous computing* [3] and little more simplistic version of them (email notifications) are present in almost every *information system* as a standard approach to notify (and prompt) user in the case of password change, registration approval or account state change. But the real power of notifications come when there is more sophisticated business logic associated with generation of these events such as in [4]. Other useful applications of such a notification service are areas where traditional paper based communication/notification means are used [5]. Consider simple example—in information system dedicated to organize academic conferences important criterion for usability would be for user to receive notifications about paper submission and paper approval or rejection. They would also expect to be notified about other more real-time events like reschedule of certain presentation. This kind of business logic is usually system-specific but means of delivering these notifications are usually the same: email or SMS (short message service for cellular network). There is one additional channel that we find very useful (also indicated in [1]) and that is voice channel, namely text to speech synthesis delivered into cellular network. Because of repetitive use of

this notification infrastructure (e.g. [6]) it would be beneficial to create service that would provide all these notification means.

In this report we describe service that complies to above description—NotX. In the first part of the paper we describe business requirements that are relevant for such service. Then the actual architecture and technological details of NotX are presented. The last part of paper is dedicated to discussing the development process used to drive NotX development and possible directions of further work on NotX.

### II. BUSINESS REQUIREMENTS

NotX's first deployment hence first real use case is to serve as a notification service to Takeplace [11] information system to send various notifications including:

- emails with password change/registration
- rescheduling of presentation (this is typically delivered by SMS or voice)

Notification is sent dynamically via appropriate engine according to user settings and global NotX settings. Voice and SMS engines are a very fast way to notify the user but uses of these engines are charged so their use is limited and must be controlled.

Voice notifications can be delivered into cellular network or to SIP (Session Initiation Protocol - signaling protocol for internet phone communication). Motivation for SIP can be found in [7].

Because it is anticipated that the use of notifications will be massive and certain groups of users will be repetitively notified (for example attendees of certain conference) we demand *tagging* of users. Information system developer (IS developer) should be able to send notifications to either specific user or to specific tag.

Required operations to be performed via NotX are:

- tag (userid, tag)
- unTag (userid, tag)
- sendNotification (dest, msgType, templateName, placeholderVals)

The *tag* parameter in *tag* and *unTag* is text with '.' characters permitted, e.g. *ConfernceA.attendee* or *ConferenceA.speaker*. The first part of *tag* parameter up to '.' is called *domain*. The tag does not have to include the domain, the domain is

used only for billing and statistical purposes. The *userid* is unique identifier of user to be tagged. The *sendNotification* operation is used to send notification itself. The parameter *dest* is used to specify to which entity the message should be sent. It can be either *userid* prefixed with ':' or it can be tag. When tag is used in this parameter the message will be sent to all tagged users. Parameter *msgType* is used to add more semantic to message, e.g.: *important*. Administrator of NotX can use this semantic parameter to configure NotX to send all *important* messages via predefined engine, e.g for TTS (text to speech synthesis). Next parameter *templateName* is used to specify which message should be sent, for example template for registration approval *registration\_approval* is template of message that is sent when registration process is successful. Lastly *placeholderVals* is associative array that is used to inject values into template.

Takeplace itself is a distributed web application, and has many different developers that are experienced in various technologies (ranging from PHP to Servlets) hence every one of these developers is used to a different way of accessing services. NotX should take this into account and make it as easy as possible to access NotX service.

Next important requirement is concerned with internationalization. Because academical conferences are usually attended by participants from various countries it is convenient for them to receive notifications in their own language. This is important equally for voice, sms and email notifications.

Because NotX uses charged services like cellular network the NotX has to keep track of sent notifications with information about domain to which they were sent.

Regarding nonfunctional requirements, the most important is to handle peaks of notifications with persistent fail-over. Usually if there is one big notification for all participants of major conference there can be thousands of various messages sent via email, SMS or voice synthesizer (TTS). It's not necessary to deliver all notifications at once but system should not render unresponsive or shouldn't crash and all messages should be delivered.

### III. ARCHITECTURE

NotX is developed to be a scalable platform and protocol independent Service. Currently the NotX is deployed to serve as a service for Takeplace so thousands of messages can arrive per second at peak hours.

Necessary attribute of reusable software service is its platform independence. That's why NotX and its components are built using Java programming language. NotX itself is web application that is built using build tool Maven [2].

The main output of the build process are two WAR (Web Application Archive) files: *Notx.war* and *Communication.war* which is web application that exposes protocols used as an interface into NotX. These main components are depicted in Figure 1. These WAR archives correspond to main components of NotX architecture - the core logic itself (NotX) and communication module (the *Communication.war*).

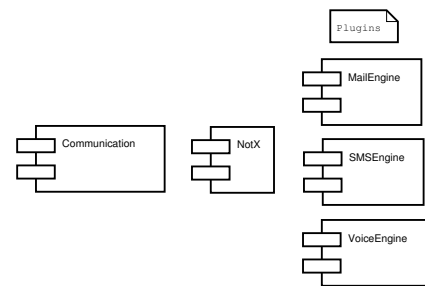


Fig. 1. Components of NotX

JMS (JSR 914) is specification for messaging API between loosely coupled components of information system. We are using Apache Active MQ implementation which supports persistent fail-over. Considering fail-over there are several fails that can happen during NotX's lifecycle:

- 1) Problem with external engine provider (SMS or voice)
- 2) Problem with connectivity to communication module with NotX
- 3) Bug in NotX logic
- 4) Any fail of hardware while processing notification

To address all of these problems our architecture is queue centric as seen in overall design in figure 2. After receiving request for notification the communication module immediately sends the request into persistent queue.

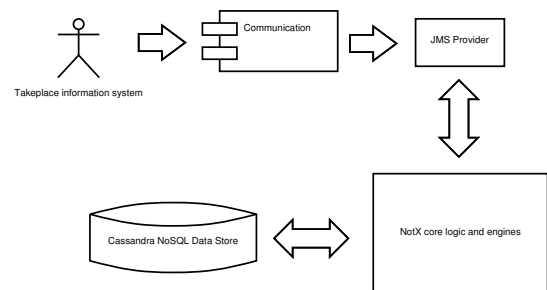


Fig. 2. Overall design

The most important operation of NotX is *sendNotification*. We will describe core logic behind this operation in more detail. As noted in business requirements this operation takes four parameters: *dest*, *msgType*, *templateName*, *placeholderVals*. Important logic takes place when *sendNotification* is called and destination is set to some specific tag, for example *ConferenceA.attendees*. Following steps take place after request has arrived to communication module:

- 1) Communication module recognizes the request as notification request and puts new notification request message *A* into message queue
- 2) NotX logic starts processing *A* by looking up *N* users which are to be notified by notification in *A*. Then NotX generates *N* messages  $\{A_1, \dots, A_N\}$  and puts them back into message queue.



- 3) Note that up to this point there was no interaction with any engine. Now NotX will be continuously receiving messages  $A_x \in \{A_1, \dots, A_N\}$  from message queue and each such message is processed as follows:
  - a) Find language  $L$  of user for whom  $A_x$  is dedicated.
  - b) Look up template for  $A_x$  according to  $L$
  - c) Now NotX injects *placeholderVals* into template and uses selected engine to notify the user. If this whole process is successful then notification statistic is saved into data store.

If there would be any kind of problem with data store or connectivity the messages are kept in message queue for administrator to manually decide how to deal with them.

Step number two is very important. The generation of  $A_1, \dots, A_N$  messages helps to more evenly distribute load on the system and also helps traceability of the system. For example when notification request is to be processed for *ConferenceA.attendees* that can mean notification of 1000 users. When even 1 notification fails it is beneficial to know which one failed and why. After bug fixing it's important to be able to swiftly retry sending exactly the same notification as failed previously.

#### A. Protocol access

Requirement for protocol access may occur in many contexts. NotX provides following interfaces (as depicted in Figure 3):

- JSON (JavaScript Object Notation) interface via HTTP POST for simple notification sending
- Thrift interface for higher level languages (framework for cross-language service development)
- native Java libraries
- web services

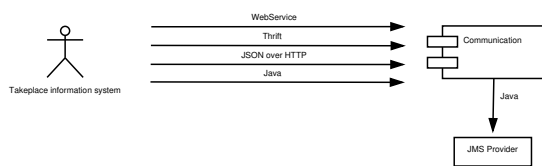


Fig. 3. Protocols

Adding new communication protocol is fairly easy. It means modifying Communication module, which resides in directory *src/notx-communication*. When adding new protocol, it is necessary to implement all NotX methods. Each method implementation usually just creates standard JMS message and puts it into MQ. Then only modification of *Communication-Main* class will make sure that after launch of communication module the interface into NotX will be functional.

#### B. Data storage

NotX uses data store for:

- information about users - contacts and tags
- statistics

- fail-over

Each user has his contacts stored in data store. This way NotX is able to send notifications by any engine for this particular user.

Statistics are saved mainly to charge users of paid services and performance tinkering.

Data storage is also used as a fail-over mechanism. Whenever a notification message is not sent successfully it is saved into data store and can be viewed via web interface with exception that caused the failure. It's possible to send specific failed notifications back to the message queue to retry the sending.

The data store itself is implemented using Cassandra NoSQL database. Decision to choose this storage type was led by need for multi-platform and highly scalable data store.

#### IV. DEVELOPMENT PROCESS

The development process of NotX was driven by SCRUM methodology. This agile process introduced by Schwaber and Shuterland [8] suits development of the NotX best because requirements were from start more about searching of possibilities instead of launching repeatable processes.

SCRUM itself is being used with two week sprints (sprint is one iteration in SCRUM). Each sprint starts with sprint planning, where spring goal is presented (major functionality, or tangible goal that is to be produced by this sprint) and product backlog items for this sprint are presented (product backlog item is high level business requirement). SCRUM itself doesn't give many hints how to specify backlog items, but there are publications addressing this issue e. g. [9] which introduce *user stories* into SCRUM.

Then development proceeds and at the end of the sprint *sprint review* takes place where output of an iteration is presented. In NotX settings the sprint review and sprint planning took place same day, usually on Wednesday.

Product backlog as well as sprint backlog are kept in Open Office spreadsheet. This low tech approach always yields less administration and more focus on actual work. From backlog it can be derived how much work was spent on specific product backlog item each day and how well estimated the task was.

To our knowledge there are no major modifications of SCRUM methodology for web development (also in [10] no consistent difference was reported in decision making for web projects). There are however some subtle differences when developing system such as NotX in general (not just with agile practices):

- External tools spike first
- Sprint review should contain technological details
- Sprints to refactor code has to be more explicitly specified
- More focus on automatized integration testing

It's essential that each external tool like TTS system or SMS gateway that are used to carry out notifications are spiked first before adding any product backlog items that are dependent on this TTS. We recommend to have a sprint in which external tools are examined. Such a sprint helps in planning next

sprints because developers of NotX can help product owner to prioritize and estimate product backlog items that will include external tool usage.

Sprint review should include technological details because product owner represents technologically experienced users (developers of information system).

Sprints to refactor code has to be specified very explicitly with carefully formulated sprint goal. Sprint goals of these sprints shouldn't be vague or not measurable like: create more readable code. But there should be measurable goals e. g.:

- write automated test that will fire up in-memory database and perform CRUD (Create Read Update Delete) operations
- rewrite logic of configuration loading and present this new design using class diagram and sequence diagram at sprint review

Focus on integration testing comes from the fact that NotX itself uses several external systems and lot of logic is simple orchestration between JMS provider and external engines. This makes unit testing less effective.

All points above can be addressed with SCRUM by managing content of product backlog and sprint reviews.

#### V. FURTHER WORK

In future, we plan to extend NotX to be publicly available service to be used by any IS developer. Technically it is possible right now because NotX supports lot of protocols for communication. There are, however, some missing functionalities like IS developer registration or billing reports.

Last important way to add more functionality to NotX is extending its communication module. There are many possibilities:

- Facebook engine—engine that notifies directly into accounts wall or private message
- Twitter engine—sends the notification to twitter
- FTP/SCP engine—puts the notification on FTP server or via SCP on some server
- IRC engine
- Skype engine—calling by skype. We didn't tested feasibility of this option yet.

The design of NotX, especially fact that it stores user information is preparation of NotX to become fully publicly available service that won't disclose users contacts. By managing contacts NotX is also single point where user can change his contacts and single point in which user can cut off potential notifications from various sources.

#### VI. CONCLUSION

In this paper we reported state of NotX - Service with capability of sending notifications via various engines. NotX gives value added to information system developers by taking burden of setting up infrastructure to send SMS, voice and email notifications. Additionally NotX helps with contacts management as it stores the contact information about users and doesn't reveal those contacts to IS developer.

NotX reduces time to integrate interesting functionality for any new information system with low development time and bring out of the box governance capabilities like fail-over, statistics and large scale notification sending in various languages. While doing all this NotX doesn't disclose any information about the user except his identifier that will be used to send notification.

#### ACKNOWLEDGMENT

The authors would like to thank Pavol Grešša for refining architecture of NotX and also to Lukáš Rychnovský for ideas from CEP and experience with building large scale distributed application that he shared.

#### REFERENCES

- [1] Kyuchang Kang, Jeunwoo Lee and Hoon Choi, "Instant Notification Service for Ubiquitous Personal Care in Healthcare Application" in *International Conference on Convergence Information Technology 2007* pp. 1500-1503
- [2] Apache Maven Project  
<http://maven.apache.org/>
- [3] Schmandt, C. and Marmasse, N. and Marti, S. and Sawhney, N. and Wheeler, S. "Everywhere Messaging" in *IBM Syst. J.*, vol. 39, issue 3-4, July 2000, p. 660-670
- [4] J. Jeng and Y. Drissi, "PENS: A Predictive Event Notification System for e-Commerce Environment" in *The Twenty-Fourth Annual International Computer Software and Applications Conference*, October 2000.
- [5] Chi Po Cheong; Chatwin, C.; Young, R.; "An SOA-based diseases notification system" in *Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on*, vol., no., pp.1-4, 8-10 Dec. 2009 doi: 10.1109/ICICS.2009.5397519
- [6] Mohamed, Nader Al-Jaroodi, Jameela Jawhar, Imad A generic notification system for Internet information in *Information Reuse and Integration, 2008. IRI 2008. IEEE International*
- [7] A. Sadat, G. Sorwar, M. U. Chowdhury, "Session Initiation Protocol (SIP) based Event Notification System Architecture for Telemedicine Applications" in *1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse (ICISCOMSAR'06)*, pp. 214-218, July 2006.
- [8] Ken Schwaber, Mike Beedle *Agile Software Development with Scrum* Prentice Hall, 2001
- [9] Mike Cohn *User Stories Applied For Agile Software Development* Addison-Wesley, 2010 ISBN:0-321-20568-5
- [10] Carmen Zannier and Frank Maurer Foundations of Agile Decision Making from Agile Mentors and Developers in *Extreme Programming and Agile Processes in Software Engineering*, June 2006, LNCS 4044, p. 11-20
- [11] <http://www.takeplace.eu/en>

## Commonality in Various Design Science Methodologies

Łukasz Ostrowski  
Dublin City University,  
Glasnevin, Dublin 9, Ireland  
Email:  
lostrowski@computing.dcu.ie

Markus Helfert  
Dublin City University,  
Glasnevin, Dublin 9, Ireland  
Email:  
markus.helfert@computing.dcu.ie

**Based on reviewing foremost literature, the paper discusses various design science research methodologies along with their case studies. It concentrates on activities (tools, methods, actions) that are used while constructing an artefact. We have identified common activities occurring across ‘design’ steps, which were not indicated in their methodologies. Combining them and drawing on that finding, we propose a concept of reference model, which gives more insights and additionally dissipates design science high level of abstraction.**

### I. INTRODUCTION

**O**VER the last years design science (DS) research has received increased attention in computing and information systems (IS) research [1,2]. It has become an accepted approach for research in the IS discipline, with dramatic growth in recent, related literature [3,4,5,6].

Research, as a process, is “the application of scientific method to the complex task of discovering answers (solutions) to questions (problems)” [7]. We can differentiate between the study of natural systems, such as physics, biology, economics and sociology [8], and the creation of artificial ones, such as medicine and engineering [8,9]. The core mission of the former is to develop valid knowledge to understand the natural or social world, or to describe, explain and possibly predict. The centre of the latter is to develop knowledge that can be used by professionals in the field in question to design solutions to their field problems. Understanding the nature and causes of problems can be a great help in designing solutions, and is the focus of design science [10]. However, design science does not limit itself to the understanding, but also aims to develop knowledge on the advantages and disadvantages of alternative solutions [10]. Though literature reflects healthy discussion around the balance of rigor and relevance [11] in DS research, agreement on the DS fundamentals aspects such as definition, methods, outputs has yet to be achieved [12]. The area is still being shaped [2,13].

Views and recommendations on the methodology of DS research vary among papers, e.g. [14,15,16,7,17,18,19,20,21,22]. One set of guidelines by Hevner [11] has been widely cited, there being concern with

their high-level and lack of specificity [23]; however, some papers reveal few instances of their actual applications [24].

Thus, though generally highly regarded and widely cited, DS methodological guidance from the precursors Hevner [11] and Walls [25] is seldom ‘applied’, suggesting that existing guidelines and methods are insufficiently clear, or inadequately operationalised - still too high a level of abstraction [18]. Alturki [23], inspired by Winter [12] stating that there was a “lack of a commonly accepted reference process model for DS research”, structured DS Roadmap to guide researchers across the DS lifecycle. In our opinion, this is the most comprehensive collection of design science paradigms to date. However, we argue that some proposed DS methodologies concentrate on developing artefacts for specific aspects of IS [26,14,5] and therefore we still take others into account.

In this paper we discuss common activities that occur across various DS methodologies in a step in which an actual artefact is being created/produced/developed. Some authors refer to the step as build [17], design & development [18], design solution [27], or develop (construction) [23] For the purpose of this paper we refer to it as the construction step.. The paper is organized as follows. First, following Offerman’s [16] claim that not much guidelines is provided in IS literature on construction step, we will present the lack of details by discussing selected DS methodologies. Next, we identify activities of that step in case studies that were conducted in order to evaluate those methodologies. By activities, we mean tools, methods, and/or actions taken by researchers to gain sufficient knowledge in order to create/produce/develop an artefact. It’s worth noticing, that these activities, even actually used, were not mentioned in those methodologies, but Offerman’s [27]. Then, we distinguish these activities that are common across DS methodologies and propose a concept of a reference model for the construction step. It could be seen as a guideline for this step regardless of selected methodology. Finally, we will discuss further research on the reference model and its application to DS.

### II. VARIOUS DESIGN SCIENCE METHODOLOGIES

Methodology is the philosophy of the research process which “includes the assumptions and values that serve as a

rationale for research and the standards or criteria the researcher uses for interpreting data and reaching conclusion” [28]

A number of researchers, both in and outside of the Information Systems (IS) discipline, have sought to provide some guidance to define design science research [11]. Their work in engineering [29,30,31,32], computer science [33,34], and IS [35,36,33,17,20,25,22] have aimed to collect and distribute the appropriate reference literature [19]; characterize its purposes, differentiate it from theory building and testing research and from other research

paradigms.

They enhanced its essential elements; and claim its legitimacy. Some researchers in IS and other disciplines have contributed ideas for process elements [29,34,30,25,20,33]. These papers include some component in the initial stages of research to define a research problem. Figure 1 illustrates the most influential papers helping shape design science tenet over two decades, in our opinion. Nunamaker et al. [7] and Walls et al. [25] emphasized theoretical bases, whereas engineering researchers [29,30] focused more on applied problems. Takeda et al. [34]

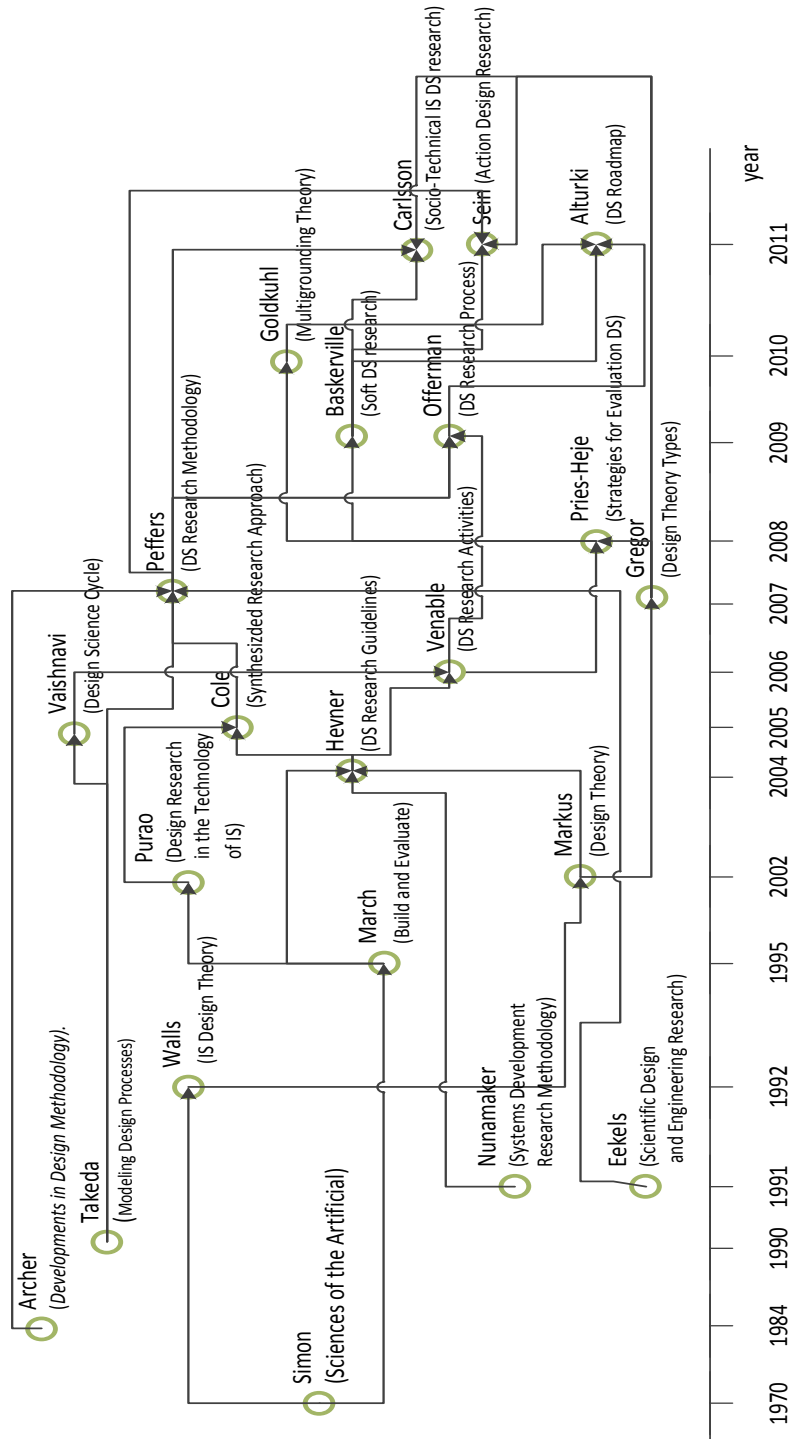


Figure 1 Design Science paradigms over years

suggested the need for problem enumeration. Hevner et al. [11] asserted that design science research should address important and relevant problems. Based on those representative papers which stated or suggested process elements, the components of the design science research methodology (DSRM) were synthesized by Peffers [18].

Even though there were different methodologies, we observed a common agreement on their outcomes. Researchers define the DS outcome as an artefact, in form of a construct, model, method, and an instantiation [17,11]. Some researchers understand artefacts as “things”, i.e. entities that have some separate existence [31]. Constructs are defined as “concepts” and “conceptualizations” [17] and “vocabulary and symbols” [11]. These constructs are abstracted concepts aimed for theorizing and trans-situational use. “Conceptualizations are extremely important in both natural and design science. They define the terms used when describing and thinking about tasks” [17]. Models are not conceived as abstract entities in the same way as constructs. “Models use constructs to represent a real world situation – the design problem and its solution space...” [11] “Models aid problem and solution understanding and frequently represent the connection between problem and solution components enabling exploration of the effects of design decisions and changes in the real world.” [11]. A method is defined as “a set of steps (an algorithm or guideline) to perform a task” [17]. An instantiation is a prototype or a specific working system or some kind of tool [31]. Most researchers agree on those form of artefacts (e.g. [19,3,23]); however, the methodology to achieve them varies [18,26,14].

### III. COMMONALITY IN ARTEFACT CONSTRUCTION

Having thoroughly read articles from Figure 1 we observed that researchers (e.g. [26,19,21]) clearly pointed out to the *construction* step as the one where the artefact is formed; however, without giving much details on how to approach it. To gain additional details, we decided to connect those steps with case studies of their methodologies. We excluded papers that did not present design science methodologies or put forward case studies that did not provide enough insight to withdraw seeking activities. Then, we created a table that provided only names of the construction steps in proposed design science methodologies and descriptions of undertaken activities in relevant case studies. Commonalities in different steps were out of scope in our search. Upon constructing the table, we analysed those activities in regards to the source from which information about artefact is gathered. We observed that two main streams could be distinguished. Researchers either reached to relevant literature or collaboration with relevant practitioners in order to construct artefacts.

Upon constructing the table we observed that the main activities in construction steps concerned literature review and collaboration with practitioners. By literature review we understand activities that lead to review the critical points of current knowledge and or methodological approaches on a

particular topic (e.g. the seeking solution). It may be seen as preparation, gathering knowledge, or building foundation on which the artifact is being constructed. Collaboration with practitioners reveals that the act of designing does not occur in isolation. It is a living process engaging practitioners from the field. The bilateral construction of an artefact falls within the scope of engaged scholarship presented by Van de Ven. [37]. The level of engagement may depend on the nature of seeking artefact. Nonetheless the conducted activities, such as focus group discussions, semi-structured interviews, and workshops will still be concerned as the main facilitators in the act of design. Our search indicated, that in 78% of all case studies, the researchers gathered relevant information, for constructing artefacts, from literature and contacting practitioners from the field. The rest 22% focuses mainly on relevant literature. Those facts and the commonality that we discovered from various design science methodologies led us to suggest a concept of the reference model that could facilitate the construction step in DS research.

Figure 2 illustrates a place of our concept model in DS. The reference model will provide description of activities and the proper order that should be undertaken in construction step regardless of used DS research methodology. It will play role of facilitator guideline rather than a solution adviser.

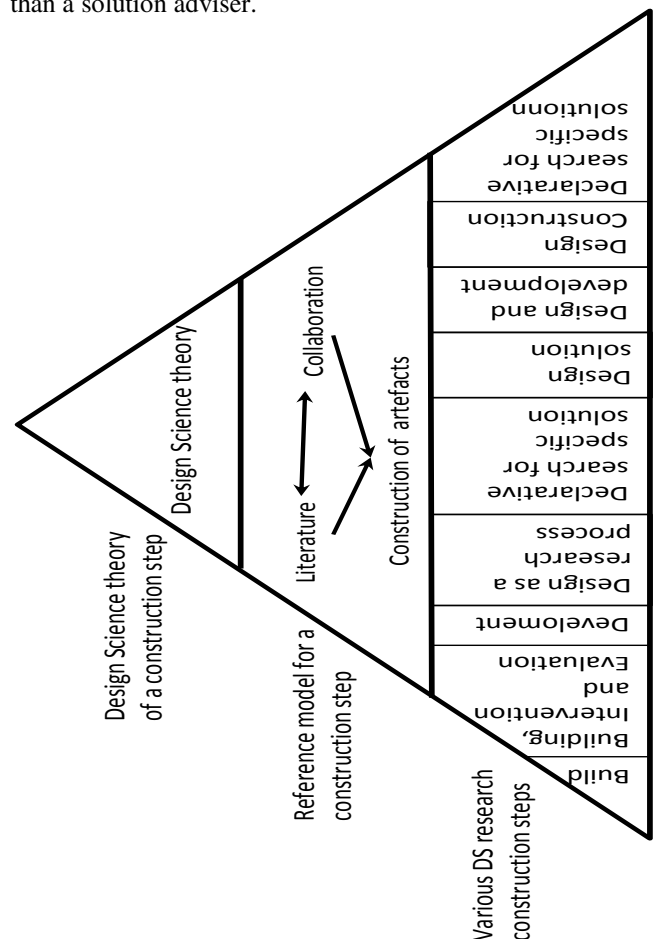


Figure 2 Place of Reference Model in DS

#### IV. CONCLUSION

In summary, we observed that literature and collaboration with practitioners play an important role in constructing/producing/developing an artefact. However, since our concept was made on case studies from various design science research methodologies, additional research on these activities is advised. The reference model can be further constructed upon investigating how these activities impact on DS research methodologies and particular artefact. In other words, to get reference model we need filter out all factors which occurrences are caused by specific solutions rather than constitute a solid path.

In this paper we discussed commonality that occurs in various design science methodologies in respect to a construction step. This is the step when information on and construction of solution is produced. We stated that DS methodology should not only indicate directions of research, but provides detailed steps on how to conduct this research. We identified common activities of construction step in case studies, which were conducted and presented throughout variety of DS papers. We proposed that these activities could be used as a reference model that would facilitate construction step regardless of selected DS research methodology. Before designing such a model, we suggested additional research, which would focus on the potential impact of those activities on particular methodologies. This is required in order to state a proper order and detail those activities further. Next the reference model will be validated via case studies in our future work.

#### REFERENCES

- [1] J. Iivari, "A Paradigmatic Analysis of Information systems as a Design Science," *Scandinavian Journal of Information Systems*, vol. 19, no. 2, pp. 39-64, 2007.
- [2] B. Kuechler and V. Vaishnavi, "On Theory Development in Design Science Research: Anatomy of a Research Project," *European Journal of Information Systems*, vol. 17, no. 5, pp. 489-504, 2008.
- [3] G. Goldkuhl and M Lind, "A Multi-Grounded Design Research Process," in *Global perspectives on design science research DESRIST 2010*, vol. 6105, Berlin, 2010, pp. 45-60.
- [4] Richard Baskerville, "A Response to the Design-Oriented Information," *European Journal of Information Systems*, pp. 11-15, 2011.
- [5] S.A. Carlsson, S. Henningsson, Hrstinski S., and Keller C., "Socio-technical IS design science research: developing design theory for IS integration management," *Information Systems and E-Business Management*, vol. 9, no. 1, pp. 109-131, 2011.
- [6] Hubert Osterle et al., "Memorandum on design-oriented information," *European Journal of Information Systems*, pp. 7-20, 2011.
- [7] J.F. Nunamaker, M. Chen, Purdin, and T.D.M., "Systems Development in Information Systems Research," *Journal of Management IS*, vol. 7, no. 3, pp. 89-106, 1991.
- [8] J. E. Van Aken, "Management Research Based on the Paradigm of the Design Sciences: the Quest for Tested and Grounded Technological Rules," *Journal of Management Studies*, vol. 41, no. 2, pp. 219-246, 2004.
- [9] H. A Simon, *The Sciences of the Artificial*, 3rd ed. Cambridge: MIT Press, 1996.
- [10] J.E Van Aken, "Management Research as a Design Science: Articulating the Research Products of Mode 2 Knowledge Production in Management," *British Journal of Management*, vol. 16, no. 1, pp. 19-36, 2005.
- [11] A.R. Hevner, S.T. March, J. Park, and S Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, pp. 75-106, 2004.
- [12] R. Winter, "Design Science Research in Europe," *European Journal of Information Systems*, vol. 17, no. 5, pp. 470-475, 2008.
- [13] J. Iivari and J. Venable, "Action research and design science research—seemingly similar but decisively dissimilar," in *17th European Conference on Information Systems*, 2009.
- [14] R. Baskerville, J. Pries-Heje, and J Venable, "Soft Design Science Methodology," in *DESRIST 2009*, Malvern, 2009.
- [15] P Järvinen, "Action Research is Similar to Design Science," *Quality & Quantity*, vol. 41, no. 1, pp. 37-54, 2007.
- [16] A. Hevner, "A Three Cycle View of Design Science Research," *Scandinavian Journal of Information Systems*, vol. 19, no. 2, pp. 87-92, 2007.
- [17] S. March and G Smith, "Design and Natural Science Research on Information Technology," *Decision Support Systems*, vol. 15, no. 4, pp. 251-266, 1995.
- [18] K. Peffers, T. Tuunanen, and M. Rothenberger, "A Design Science Research Methodology," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45-77, 2007.
- [19] V. Vaishnavi and B. Kuechler, "Design Research in Information Systems," *Association for Information Systems*, 2005.
- [20] M. Rossi and M.K. Sein, "Design Research Workshop: A Proactive Research Approach.," in *26th Information Systems Research Seminar in Scandinavia*, Haikko, 2003, pp. 9-12.
- [21] R. Baskerville, J. Pries-Heje, and J. Venable, *Soft Design Science Research: Extending the boundaries of Evaluation in Design Science Research*. Pasadena: Claremont Graduate University, 2007.
- [22] J. Venable, "A Framework for Design Science Research Activities," in *The 2006 Information Resource Management Association Conference*, Washington DC, 2006.
- [23] A. Alturki, Gable G.G., and Bandara W., "A Design Science Research Roadmap," in *DESRIST 2011*, vol. LNCS 6629, Heidelberg, 2011, pp. 107-123.
- [24] M. Indulska and J.C. Recker, "Design Science in IS Research: a Literature Analysis.," in *4th Biennial ANU Workshop on Information systems Foundations*, Canberra, 2008.
- [25] J. Walls, G. Widmeyer, and O. El Sawy, "Building an Information System Design Theory for Vigilant EIS," *Information Systems Research*, vol. 3, no. 1, pp. 36-59, 1992.
- [26] M.K. Sein, O. Henfridsson, S. Pura, M. Rossi, and K. Lindgren, "Action Design Research," *MIS Quarterly*, vol. 35, no. 1, pp. 37-56, 2011.
- [27] P. Offermann, O. Levina, Schönherr M., and U Bub, "Outline of a Design Science Research Process," in *Design Science Research in Information Systems and Technology*, Malvern, 2009.
- [28] K.D. Bailey, *Methods of Social Research*.: The Free Press, 1982.
- [29] L.B Archer, "Systematic Method for Designers," in *Developments in Design Methodology*, London, 1984, pp. 57-82.
- [30] J. Eekels and N.F.M. Roozenburg, "A Methodological Comparison of the Structures of Scientific Research and Engineering Design-Their Similarities and Differences," *Design Studies*, vol. 12, no. 4, pp. 197-203, 1991.
- [31] G. Goldkuhl, "Design Theories in Information Systems – A Need for Multi-Grounding," *Journal of Information Technology and Application*, vol. 6, no. 2, pp. 59-72, 2004.
- [32] Y Reich, "The Study of Design Methodology," *Journal of Mechanical Design*, vol. 117, no. 2, pp. 211-214, 1994.
- [33] M. Preston and N. Mehandjiev, "A Framework for Classifying Intelligent Design Theories," in *The 2004 ACM Workshop on Interdisciplinary Software Engineering Research*, New York, 2004, pp. 49-54.
- [34] H. Takeda, P. Veerkamp, T. Tomiyama, and H. Yoshikawam, "Modelling Design Processes," *AI Magazine*, vol. 11, no. 4, pp. 37-48, 1990.
- [35] L. Adams and J. Courtney, "Achieving Relevance in IS Research via the DAGS Framework," in *37th Annual Hawaii International Conference on System Sciences*, 2004.
- [36] R. Cole, S. Pura, M. Rossi, and M.K Sein, "Being proactive- Where Action Research Meets Design Research," in *26th International Conference on Information Systems*, Atlanta, 2005, pp. 325-336.
- [37] A. H Van de Ven, *Engaged Scholarship: A Guide for Organizational and Social Research*. Oxford: New York: Oxford University Press, 2007.

# A Hybrid Algorithm for Detecting Changes in Diagnostic Signals Received From Technical Devices

Tomasz Pełech-Pilichowski  
AGH University of Science and  
Technology, Faculty of  
Management, ul. Gramatyka 10,  
30-067 Kraków, Poland  
Email: tomek@agh.edu.pl

**Abstract**—In this paper, a hybrid two-level algorithm of the original changes in diagnostic signals received from multiple technical devices is presented. Research are aimed at identification of the changes, deviations or patters (events), through concurrent diagnostic signals processing, which occur in one selected signal (proposed algorithm is adjusted to omit concurrent and time-lagged changes). In the first stage, detection is based on non-stationarity detection with the short-term prediction comparison. In the second stage, dedicated distance-like measure is employed. Detection results obtained for sample random signals including simulated large deviations are presented.

## I. INTRODUCTION

TECHNICAL systems, for example Intelligent buildings ones, are often consist of technical networked devices supervised by a central unit or controller. Efficient system working depends on the reliable installed devices operation which may be successfully provided by the diagnostic signals concurrent monitoring. Such processing of available set of diagnostic signals, in particular in real time systems, may indicate the work status of devices or results in alarm notification resulting from defective device work, faults, power instability, loss of network security or generated signal reliability. On the other hand, selected signals may contain random, temporary changes, statistically insignificant, being response to the irrelevant devices interferences, networked and supervised by a central unit. Such changes can trigger false alarms or can directly influence the work of other connected devices (provide incorrect input signal) thus endanger unstable operation of the whole system.

Event detection from diagnostic signals is usually based on implementation of a detection algorithm capable of identify the well-defined, expected unusual behavior of the processed signal (short or long-term non-stationarities) [5]. Detection algorithms are often based on a statistical and frequency domain data characterization [2], [21], [8], adapted to sampling frequency and consistency of available dataset. In such cases, complex system monitoring is based on the individual signal processing which is not sufficient to identify coincidence of events in other analyzed time-series.

This work was supported by the European Regional Development Fund, Grant no. UDA-POIG.01.03.01-12-171/08/00

Considering diagnostic signals produced by multiple system devices, a reliable detection requires the use of dedicated algorithms based on concurrent or parallel processing of all available signals to capture non-random and relevant changes in selected (real-time monitored) one, strictly including time lags between occurring events as a possible effect of transmission delays, data queuing, assumed real-time regime disturbances.

The aim of this paper is to describe and present research progress on detection algorithms dedicated to capture original changes, deviations, or patterns, i.e. which are not an effect of the same external factors, thus occurring in the only one diagnostic signal – including time-delays between possible changes (avoiding changes presented in both signals – concurred or lagged). In the paper, a hybrid detection algorithm is presented, based on concurring processing of pairs of signals in a moving window of a fixed length where detection task splits into two levels: (1) a non-stationarity detection and (2) its confirmation with distance-like similarity method. Proposed approach is focused on reducing false alarms and early efficient detection of emergency situations in multiple diagnostic signal sets.

This idea is a development of earlier work aimed at the significant event detection from time series based on statistical signal analysis [5], distance-like methods [6], short-term prediction efficiency comparison [16],[18],[15], immune paradigm employed to event detection support [17], [20], [10], [22] and two-level algorithms dedicated to process signals in real time systems [19].

## II. EVENT DETECTION FROM SIGNALS RECEIVED FROM MULTIPLE DEVICES

Event detection from time series is based on the processing of subsequent samples to identify an unusual process behavior, i. e. non-stationarities caused by external non-random factors. To achieve reliable results, the signal analyzes should be performed in a moving window of a fixed length [3], [5], [4] depending on intended impact of historical data. Event detection may be viewed as the unsupervised classification task where one class is described (one-class classification) and a formal method dedicated to distinguish between normal and anomaly class [10], [19].



Assuming that signals are generated by technical devices (for example intelligent building ones), during “normal” system behavior stationary is identified (lack of long-term and short-term abrupt changes of the mean value or variance); an appearance of non-stationarity may be recognized as the alarm signal or a specific event, i.e. pattern or unexpected change. Such sequences of changes may be visible as the deviations from the short/long term mean value exceeding a fixed threshold as the standard deviation multiplicity of arbitrary selected value. Depending on event detection task, changes of the same sign (positive or negative ones), different sign (or its absolute values) or patterns – i.e. original configuration of deviations are tested (lack of changes between series of deviations of fixed length is often assumed).

To detect short and long-term changes in a selected diagnostic signal, one may use classical, robust procedures (for example, Page-Hinkley one [1]) which reasonable employing is often limited by the computation time or a need of long data sets processing which results in usually unacceptable time-delays [14]. Implementation of the detection algorithm strongly depends on the signal characterization (statistical, frequency), properties (dimensionality, completeness), attributes of possible events (amplitude, duration, periodicity, delay) or the assumed detection error [19].

Change detection from one signal is produced with implementation of the detection algorithm and usually it is sufficient for a short-term changes identification, especially when one diagnostic signal from one device is obtained. For multiple signals, such approach may result in a number of false alarms or undetected events as a result of random signal changes (for example, short-term power differences, time delays during wire/wireless data transmission) or inability to identify a complex alarm situation (i.e. to detect a real device damage, multiple signals are often needed – temperature, input/output, revolutions per minute of central processing unit fan etc.). Moreover, significant changes may be announced by the short-term deviations in another signal. Thus, processing of a diagnostic series sets seems to be a promising way to capture such changes, deviations or patterns in a target signal.

Event detection from a selected diagnostic signal from the large available signals set (including heterogeneous ones) can be based on the concurrent, parallel or distributed time-series processing, to identify both single changes and dependencies between events detected from individual signals (with suitable algorithms). Such complex processing may be a way to faster detection through the short-term “announcing” events recognition which can precede long-term changes of time series the statistical properties [14].

There is a number of methods dedicated to time series quantitative analyzes [19], like the statistical analysis of the frequency of events [12], trends, deviants and outliers, the patterns and characteristics similarity comparison [14]. Algorithms based on neural networks [7], genetic algorithms [13] and other data mining techniques including similarity measures and distance ones [23], [9] also may be employed [8], [12]. Although such results can provide the reliable detection; their efficiency depends on dimensionality of the set of signals, the estimation of statistical parameters (mean value,

standard deviations, trend parameters), dynamics [21] and for selected ones – the learning period [17], [19].

For specific, not well-defined changes or patterns, processing of the set of signals with such standard algorithms are often not sufficient because of existing different signal properties, attributes of events, time-delays between changes in different signals and dependencies between them. Moreover, in many cases the aim is to detect only changes in one selected signal, excluding the synchronous changes in other signals and time-lagged ones. Such situations are encountered – for example – when reported statuses from the monitored devices don’t match.

### III. A HYBRID EVENT DETECTION ALGORITHM

In this paper, a hybrid event detection algorithm is proposed. It is based on the processing of pair of signals in two levels (see listing 1): first – the preliminary non-stationarity detection in individual signals, and then – the second one – the confirmation with distance-like similarity method, computed for both signals. Such approach relies on an the assumption that all signals of available dataset (or some of them) can be paired. Thus, summarized detection results (at a time instant) will indicate the complex dependencies within data set. Moreover, such computation is suitable for the distributing and employing one of many computational intelligence paradigms (like multi-agent systems).

In the first level (for sample  $n$ ) the non-stationarity is identified with the short-term prediction errors comparison obtained from the one-step-ahead zero-order-prediction (ZOP)/zero-order-hold (ZOH) model [2], [15] and the adaptive Holt predictor [11] in a moving window of constant length, suitable for non-stationary data analysis (in particular, trended time series). This preliminary detection procedure is suitable for time series which consist of the non-random components [19].

When short-term non-stationarity is recognized for a fixed number of samples (arbitrary adjusted threshold value), the second level detection is triggered (see listing 1, lines 9-15). To confirm change initial signal, the distance-like method is proposed.

The distance-like detection method – denoted as measure  $Z$  (or  $dZ$ ) and first mentioned in [6] – is dedicated to two signals similarity monitoring. It is based on the synchronous processing of two signals (denoted as  $x$  and  $y$ ) of fixed length in a moving window. For both signals  $x$  and  $y$ , the mean values of sub-sequences of absolute deviation values are calculated (the positive ( $x_{pm}, y_{pm}$ ) and the negative ( $x_{nm}, y_{nm}$ ) signs). The deviation is recognized when the absolute value exceeds a fixed threshold  $\rho_{zd}$  at the time-instant. For deviations smaller than threshold, zero value is assumed. As a result, two pairs of the positive values ( $x_{pm}, y_{pm}$ ) and negative ( $x_{nm}, y_{nm}$ ) are obtained.

```

1 n = 1
2 REPEAT
3 // level 1
4 perform ZOH prediction for both
   signals
```



```

5 perform Holt prediction for both
  signals
6 produce and compare short-term
  prediction errors
7 IF(short-term non-stationarity
  recognized)
8 {
9   // level 2
10  calculate dZ
11  IF(dZ is greater than fixed
    threshold)
12  {
13    // change detected
14    alert/raport the user/system
15  }
16 }
17 UNTIL n = N

```

Listing 1. Pseudo-code of the proposed event-detection hybrid algorithm for two signals of the length  $N$ . A fixed threshold  $p_z$  value is assumed.

The temporary similarity  $dZ$  is calculated as follows:

$$dZ = \sqrt{(x_{pm} - y_{pm})^2 + (x_{nm} - y_{nm})^2} \quad (1)$$

Processed data received from technical devices may contain the phased (time-lagged) events. To avoid such time delays between events/changes, during computing a final value of the measure  $dZ$ , a tolerance (denoted as  $L_{tol}$ ) is assumed. Such tolerance may be viewed as a permissible time delay between occurring changes in both signals. For each subsequent sample  $n$ , to compute  $dZ_n$  a  $L_{tol}$  number of the measures  $dZ_t$  is calculated in a moving window ( $t = n - L_{tol} + 1 \dots n$ ) giving a set of the  $dZ$  values. Finally,  $dZ_n$  is chosen as the smallest  $dZ_t$  value between two sub-series:

$$dZ_n = \min_{t=n-L_{tol}+1 \dots n} dZ_t \quad (2)$$

“Distance measure” term is used (instead of “distance” or “metric”) [19] because all formal definition of metric conditions are not satisfied (i.a. symmetry condition).

#### IV. CALCULATION RESULTS

To verify the effectiveness of the proposed hybrid detection procedure, the algorithm was implemented and tested on the random-simulated data. Signals were generated as time series of the length 144 containing pseudo-random values drawn from the standard normal distribution  $N(0;1)$ . Such relatively short length of the signal allows for a reliable analysis of the algorithm efficiency using data visualization.

To obtain data sets similar to real diagnostic signals (i.a. avoid negative values), a constant  $c = 30$  was added to each sample. In the next step, diagnostic signals were modified by adding the simulated changes for random generated time instants (randomly selected samples were increased). The length of simulated change was fixed to 5 in an experimentally way.

In this paper, presented detection results are chosen from among many possible ones, and they are depended on arbitrary parameter adjusting. In particular, in the first stage (the short-term prediction comparison computed for differences

between adjacent elements of processed series), the moving window width was fixed to 22. The second detection stage is triggered after recognition of series of – at least – three consecutive samples for which the non-stationary was found. To process the data in the same order of magnitude, signals were unified with their dispersions (in the moving window).

In the presented case, it was assumed that the preliminary detection performing in the first stage causes no need to use of the threshold value of the method  $dZ$ . Therefore, event detection is based on choosing the final  $dZ$  value from the set of length  $L_{tol}$  which was fixed to 5.

In the paper, four sample diagnostic signals are examined (see fig. 1-4) representing different event sets illustrated in the appropriate parts of the figures 1-16.

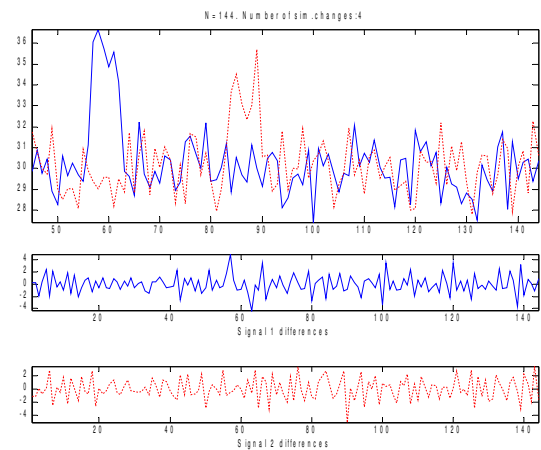


Fig. 1. Random-generated signals – part 1 (depicted with the solid and dotted lines), containing four simulated changes – the input (original) data (upper large sub-figures) and the differences between adjacent elements of the processed signals (2nd and 3rd sub-figure rows).

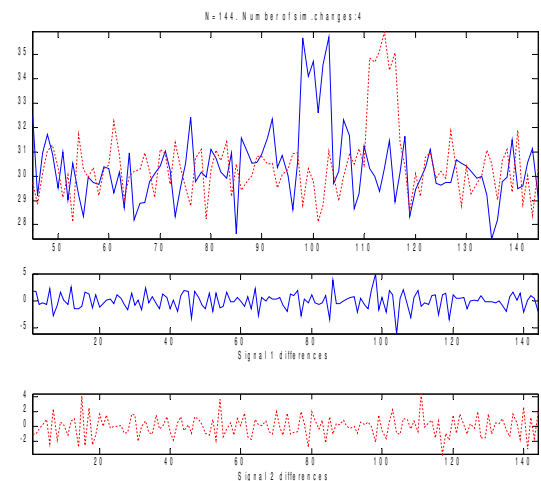


Fig. 2. Random-generated signals – part 2. Description of the symbols and lines – see Fig. 1.

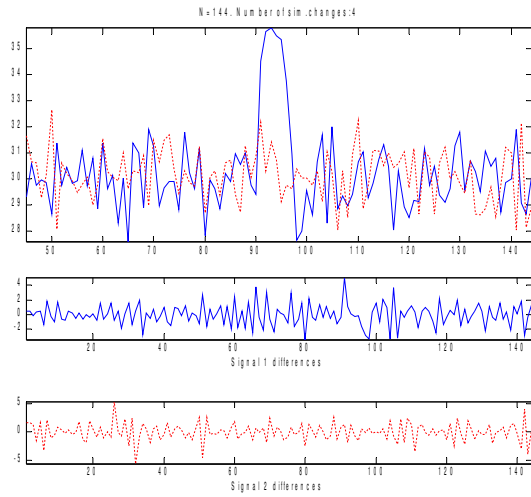


Fig. 3. Random-generated signals – part 3. Description of the symbols and lines – see Fig. 1.

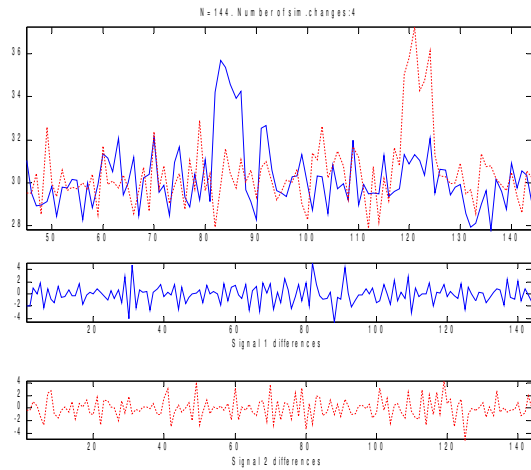


Fig. 4. Random-generated signals – part 4. Description of the symbols and lines – see Fig. 1.

It may be seen in fig. 5-8 that employing the detection performed in the first step results in the identification of the large deviations.

Fig. 9-12 illustrate the  $dZ$  value changes in the moving window. The basic assumption of the method is illustrated – especially – in fig. 10 (100-120th sample) where the  $dZ$  value is strongly depend on the samples included in the moving window. If changes in both signals are covered, the  $dZ$  value is near to zero; otherwise, abrupt  $dZ$  changes are visible.

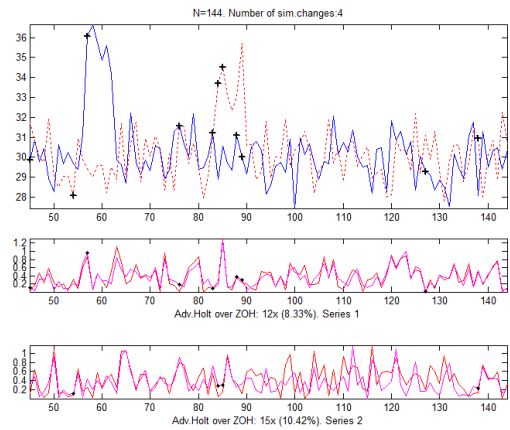


Fig. 5. Detection results obtained with the short-term prediction comparison (part 1) depicted in the row 2 and 3. Single non-stationarity detection (advantage Holt over ZOH) is denoted as black 'dots'.

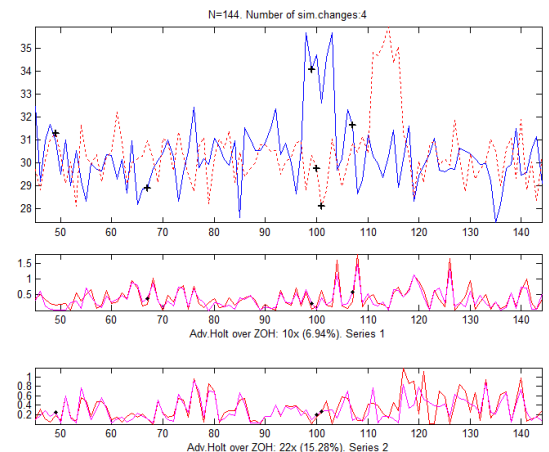


Fig. 6. Detection results obtained with the short-term prediction comparison (part 2). Description of the symbols and lines – see Fig. 5.

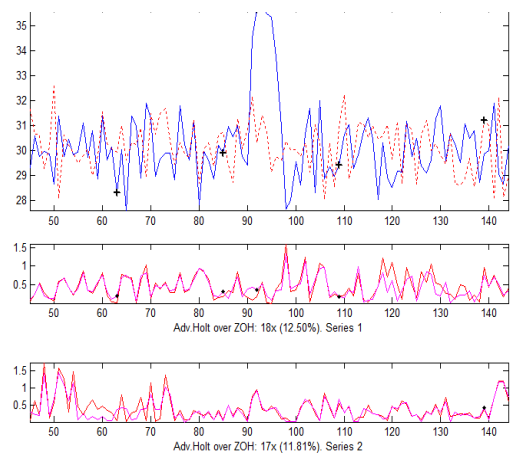


Fig. 7. Detection results obtained with the short-term prediction comparison (part 3). Description of the symbols and lines – see Fig. 5.

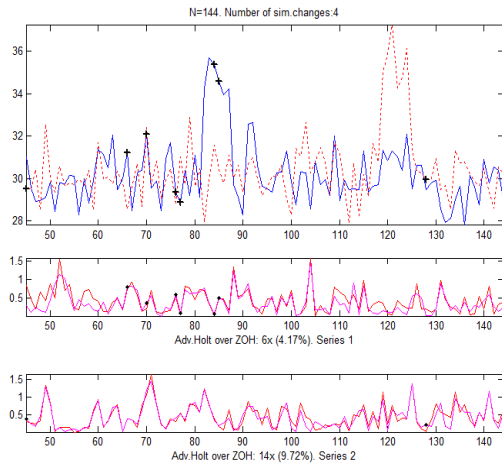


Fig. 8. Detection results obtained with the short-term prediction comparison (part 4). Description of the symbols and lines – see Fig. 5.

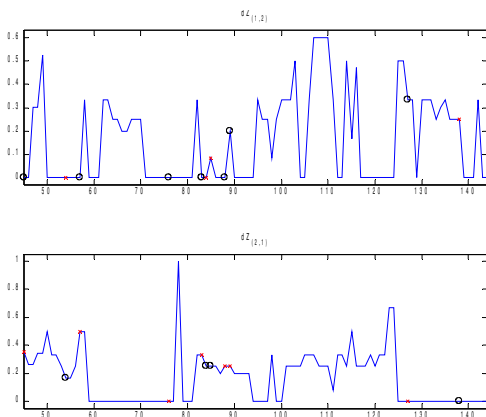


Fig. 9. Change detection performed with the proposed event-based similarity method dZ (part 1). Changed detected with short prediction comparison depicted as ‘circles’ and ‘asterisk’.

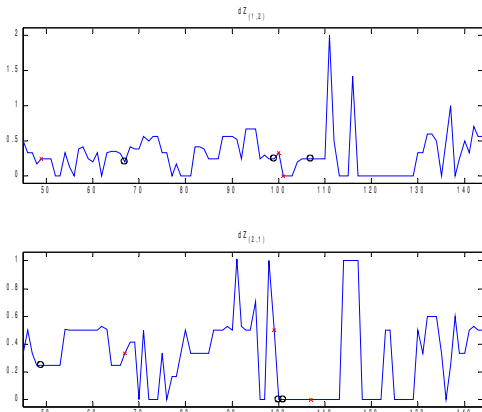


Fig. 10. Change detection performed with the proposed event-based similarity method dZ (part 2). Additional description – see Fig. 9

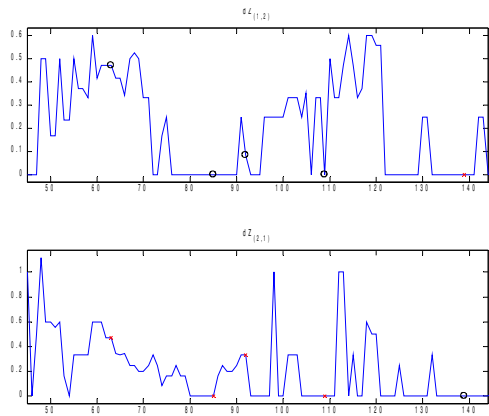


Fig. 11. Change detection performed with the proposed event-based similarity method dZ (part 3). Additional description – see Fig. 9

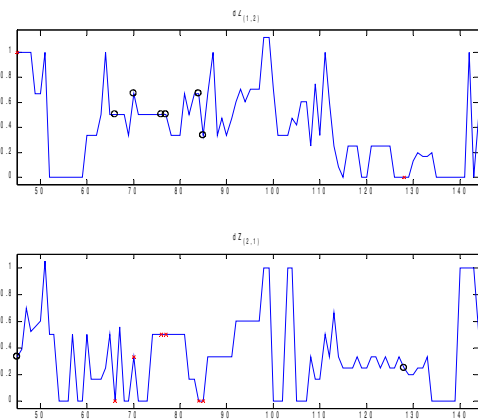


Fig. 12. Change detection performed with the proposed event-based similarity method dZ (part 4). Additional description – see Fig. 9

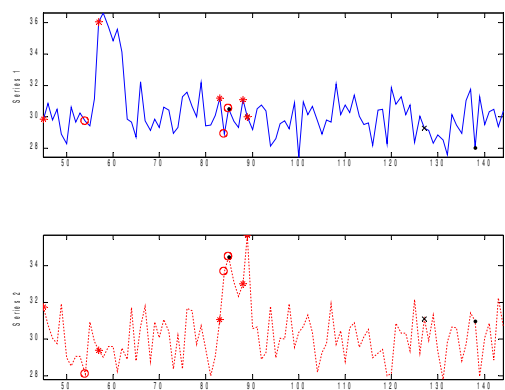


Fig. 13. Change detection from processed signals (part 1): 1 (depicted in the upper subfigure with solid line) and 2 (dotted line, the lower sub-figure). Description of used symbols: detected non-stationarities from signal no. 1 confirmed with  $d_{Z1}$  ( $H_1/d_{Z1}$ ) denoted as ‘cross’, confirmed with  $d_{Z2}$  ( $H_1/d_{Z2}$ ) depicted as ‘dot’; detected non-stationarities from signal no. 2 confirmed with  $d_{Z1}$  ( $H_2/d_{Z1}$ ) denoted as ‘asterisk’, confirmed with  $d_{Z2}$  ( $H_2/d_{Z2}$ ) – as ‘circle’.

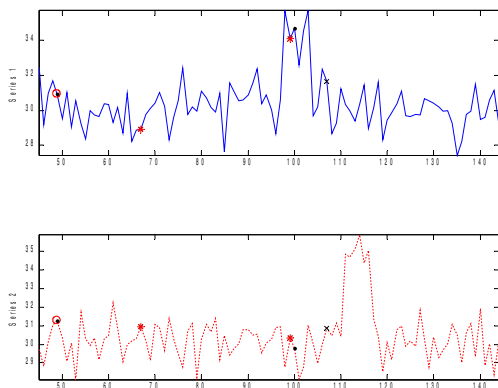


Fig. 14. Change detection from processed signals (part 2). Extended description – see Fig. 13.

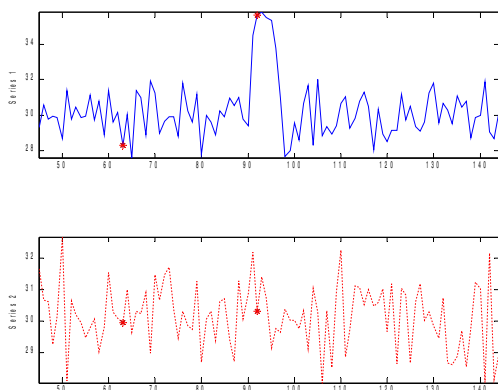


Fig. 15. Change detection from processed signals (part 3). Extended description – see Fig. 13.

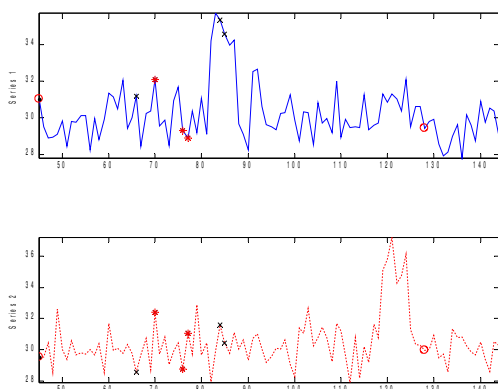


Fig. 16. Change detection from processed signals (part 4). Extended description – see Fig. 13.

As mentioned in the earlier part of this paper, research aimed at detection of the original changes in the one processed signal. Figures 13-16 show such event selection. Proposed hybrid algorithm is capable of detecting large, single

changes (see fig. 13 – for about 60th and 80th sample; in fig. 14 – about 100th sample, fig. 15 – about 95th sample). Notice, that few detection results may be viewed as false alarms (for example – fig. 16, about 100th sample), nevertheless, in such cases the interpretation will be connected with the analysis of small deviations that may be invisible in assumed figure resolutions.

## V. CONCLUSIONS

Event detection from signals received from computer systems can be focused on – depending on detection task – the independent signal processing (the identification of short and long-term changes in signals separately) or the available diagnostic signals set processing – which is more complex and allows to identify the changes in the analyzed (target) signal environment, including implicit events. The proposed detection idea is related to the second detection approach and it is a part of wide (realized and planned) research on event detection from time series through concurrent and parallel environment of analyzed signal monitoring. Summarized detection results will indicate the significant and insignificant changes and the dependencies within processed data set.

In this paper it was shown that proposed two-level algorithm is suitable to detect the changes in diagnostic signals which occur in the one signal only. The procedure efficiency was tested on the random generated signals containing simulated changes, however, for real signals, peer analysis of available signals set may be valuable.

The intention was to show the most relevant properties of the hybrid algorithm rather than data acquisition process mapping and further processing according to the real computing conditions and limitations.

Presented detection results are obtained using the algorithm whose parameters were adjusted in an experimentally way. Therefore, the parameters adapting is a promising way to effectiveness improvement. Further research will be focused on algorithm adaptation (corresponding to processed signals properties) and the stage 2 modifications towards the elimination of the false alarms (in particular, changes of the moving window width will effect in the detection resolution).

The proposed idea of multiple signal processing can be developed (especially taking into account the detection task complexity when processing large datasets) with the distributed processing of signals from networked devices as i.e. multi-agent systems or artificial immune systems (such paradigm appears to be helpful for detection viewed as an unsupervised classification, especially for signals contain many random and non-random components).

## REFERENCES

- [1] Benveniste A., Basseville M.: Detection of Abrupt Changes in Signals and Dynamical Systems. Lecture Notes in Control and Information Sciences, Vol. 77, Springer-Verlag, 1984
- [2] Box G. E. P., Jenkins G.M.: Time Series Analysis: Forecasting & Control, Prentice Hall, 1994
- [3] Brockwell P. J., Davis R. A.: Time Series: Theory and Methods. Springer Series in Statistics, 1991
- [4] Cole R., Shasha D., Zhao X.: Fast Window Correlations over Uncooperative Time Series. ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, 2005

- [5] Duda J. T., Pelech T.: Wykrywanie zdarzeń w szeregach finansowych z wykorzystaniem metod statystycznych. [In:] Inżynieria wiedzy i systemy ekspertowe, T.2, / red. Grzech A., Oficyna Wydawnicza Politechniki Wrocławskiej, 2006
- [6] Duda J. T., Pelech-Pilichowski T.: Miary podobieństwa szeregów czasowych w detekcji zdarzeń. [In:] Systemy wykrywające, analizujące i tolerujące usterki / red. Kowalczyk Z., PWN, 2009
- [7] Guh R., Zorriassatine F., Tannock J. D. T., O'Brien C.: On-line Control Chart Pattern Detection and Discrimination - a Neural Network Approach. *Artificial Intelligence in Engineering*, Vol. 13, Issue 4, Elsevier 1999
- [8] Guralnik, V., Srivastava, J.: Event detection from time series data. [In:] Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data mining, pp. 33-42, San Diego, California, USA 1999
- [9] Goldin D., Mardales R., Nagy G.: In Search of Meaning for Time Series Subsequence Clustering: Matching Algorithms Based on a New Distance Measure. *ACM International Conference on Information and Knowledge Management*, 2006
- [10] Hofmeyr S. A., Forrest S.: Immunity by Design: An Artificial Immune System. *Proceedings of the Genetic and Evolutionary Computation Conference*, San Francisco, 1999
- [11] Holt C. C.: Forecasting seasonals and trends by exponentially weighted moving averages. *Carnegie Institute of Technology*, Pittsburgh, Pennsylvania, 1957
- [12] Keogh E., Lonardi S., 'Yuanchi' Chiu B.: Finding Surprising Patterns in a Time Series Database in Linear Time and Space. [In:] *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, Alberta, Canada 2002, pp. 550-556
- [13] Kingdon J.: *Intelligent Systems and Financial Forecasting*, Springer, 1997
- [14] Mahoney M., Chan P.: Trajectory Boundary Modeling of Time Series for Anomaly Detection. *Workshop on Data Mining Methods for Anomaly Detection, Conference on Knowledge Discovery and Data Mining*, 2005
- [15] Pelech T.: Adaptive Holt's Forecasting Model Based on Immune Paradigm. *Problemy oswoenia poleznych iskopaemh. Zapiski Gornogo Instituta, Sankt Petersburg State Mining Institute*, 2006
- [16] Pelech T., Duda J. T.: Application of immune paradigm to monitoring of stock indices. *Problems of Mechanical Engineering and Robotics*, No.3, AGH-UST University Press, 2005
- [17] Pelech T., Duda J. T.: Event detection in financial time series by immune-based approach. *Intelligent Information Processing and Web Mining. Advances in Soft Computing*, Springer-Verlag, 2006
- [18] Pelech T., Duda J. T.: Immune Algorithm of Stock Rates Parallel Monitoring. *Information Systems and Computational Methods in Management*. Ed. J. T. Duda, AGH-UST University Press, 2005
- [19] Pelech-Pilichowski T., Duda J. T.: A two-level algorithm of time series change detection based on a unique changes similarity method. *IMCSIT Proceeding*, Wisła, PL, 18-20 October, 2010
- [20] Pelech-Pilichowski T., Duda J. T.: Wykorzystanie podejścia immunologicznego do prognozowania szeregów czasowych. *Automatyka: półrocznik Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie*, 2009
- [21] Wetherill G. B., Brown D.W.: *Statistical Process Control. Theory and Practice*. Chapman and Hall, 1991
- [22] Wierzbion S. T.: *Sztuczne systemy immunologiczne. Teoria i zastosowania*. Wyd. Exit, 2001
- [23] Yang K., Shahabi C.: A PCA-Based Similarity Measure for Multivariate Time Series. *ACM International Workshop on Multimedia Databases*, 2004



# Adapting Scrum for Third Party Services and Network Organizations

Lukasz D. Sienkiewicz  
Wroclaw University of Economics  
Komandorska Street 118/120,  
53-345 Wroclaw, Poland  
Email: sienkiewicz.lukasz@gmail.pl

Leszek A. Maciaszek  
Wroclaw University of Economics  
Komandorska Street 118/120,  
53-345 Wroclaw, Poland  
Email: leszek.maciaszek@ue.wroc.pl

**Abstract**—Large number of scientific publications and press releases demonstrate that organizations are adopting the Scrum software development method with success in almost all areas. Nevertheless, traditional Scrum method is not sufficient for managing work in Network Organizations where Third Party Service providers may know nothing about the Scrum. This paper describes the findings of a field study that explores the Scrum in Network Organizations. We extended Scrum core roles and proposed changes in Scrum artifacts that help in adapting the Scrum method to work in Network Organization where changes and high competition are the cornerstone of the entire process.

## I. INTRODUCTION

SELECTING appropriate approach to systems development is crucial for success of the project [1]. Fortiori this is especially important when firms are working as a Network Organization where the environment is turbulent and uncertain. Moreover, the firms that make up that kind of cooperation are likely to be using disparate methods for their internal development projects.

In this paper we propose an adaptation of the Scrum method that suits best for managing development of software applications in Agile environment in a Network Organization using third party internal and external services:

- We refer to the Network Organization and determine how the Third Party Services and the Scrum are interrelated.
- We take a *holonic view* [2][3] on the process and method of software development.
- We propose a Scrum-based software development model that specifically includes some novel and excludes some conventional artifacts and rules.
- We propose a set of metrics (i.e. Key Performance Indicators) that help to control and coordinate proposed model.

The proposed model offers an innovative approach for systems development in Network Organizations. The model has evolved from the Scrum method and has been checked in practice.

## II. NETWORK ORGANIZATION AND THIRD PARTY SERVICES IN TERMS OF SOFTWARE APPLICATION DEVELOPMENT

### A. Network Organization

“A pattern of social relations over a set of persons, positions, groups, or organizations” [4]

This definition proposed by Lee Douglas Sailer [4] and further elaborated by Marshall van Alstyne [5] is very useful because it emphasizes structure and different levels of analysis. The Network Organization is a collection of autonomous firms or units that behave as a single large entity (i.e. structure), using specific mechanisms to control and coordinate the entire project. The entities that make up that kind of organization are usually legally independent entities (separate companies). However, this is not the rule because some of them may be divisions within the company (sub-organization) that sell to outside customers, or they may be wholly owned subsidiaries providing the third party services to the entire network.

Some advantages of software development in network environment are customization, task basis, and the Structural Embeddedness [6]. These advantages are favouring individual firms and their members. Network Organizations occur in the situation when technology and markets are changing very fast, so consequently, all participants have to coordinate and control the units in some other way, for instance by mechanism design, trust, and Macro-Culture [5].

### B. Third Party Services

Network Organizations fall halfway between vertical integration and market disaggregation. They facilitate building packets of services, according to the nature of provided services and relations between them as shown in “Fig. 1”.

For the purpose of discussion, we have distinguished two types of third party services provided for software development in Network Organizations:

- Internal Services: time-consuming activities, which are important but additional to the entire project. Usually those are provided by subunits of a large company (e.g. internal UX expertise, internal testing, ICT support, etc.)
- External Services: all those issues that must not be covered by Internal Services and should be handled by other firms (e.g. authorized computer service, external testing service, translations, etc.)



We observe that companies that see their units as separate cost or profit centres (providers of internal services), may encourage the units to sell their services outside the company. The reason is that if the units have to operate within market, they will improve performance, better manage the prices, and of course earn money for the entire organization. The cooperation between services providers, usually establishes a long-term relationship between suppliers (i.e. providers of external services), who may then participate in planning sessions and influence the workload and schedule.

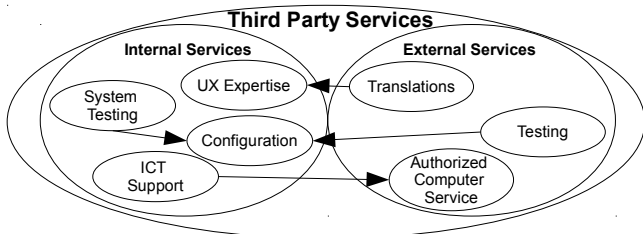


Fig 1. Relationship between Third Party Services in Network Organization

Both internal and external services may involve other third party services (internal or external) so the amount of dependencies, risk and uncertainties may be fast increasing. This in turn may adversely affect software development process by deterioration of quality, changes in the scope, delays in the schedule, and all this may contribute to the project failure.

In the next sections, we will explain the holistic nature of the Scrum. Based on an example of Network Organization and services, we will present the dependencies between key entities in the Scrum process and third party service providers.

### III. AGILITY IN NETWORK ORGANIZATION

#### A. Agile, Scrum and Engineering Practices

To highlight differences in the impact of Agile, scrum and Engineering Practices in Network Organizations we take advantage of the “*Three Level Framework*” designed by Geary A. Rummmler and Alan Brache [7]. We propose the framework that takes the Scrum viewpoint and distinguishes three types of layers:

- Organization Level: all activities that are additional for Scrum (e.g. Human Resources, financial, capability, management, etc.), and identify the organization point of view (i.e. market, competitive advantage, priorities, products and services).
- Process Level: series of steps, rules and artifacts, which are used by the Scrum team to produce the product or service. The goals of this level are developed from customer requirements (i.e. Sprint Planning) and benchmarking information (i.e. during Sprint Review/Retrospective Meeting).

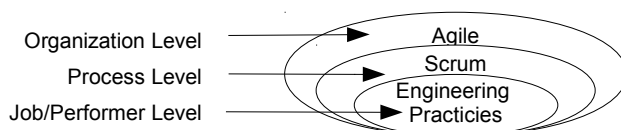


Fig 2: Scrum and Engineering Practices in the context of Agile implementation in Network Organization

- Job/Performer Level: all undertakings and instrumentation essential to achieving the goals of the process (e.g. code review, pair programming, continuous integration, etc.).

For the research presented here, we have assumed that Network Organization is following Agile principles and implements Scrum as a method for managing the software development team (Job/Performer Level in “Fig. 2”). Therefore, all Engineering Practices are considered separately, as not covered by Scrum but used as external or internal services (as discussed next).

#### B. Scrum Roles – Core and Additional

We would like to clarify that describing all Scrum principles and artifacts is not the subject of this paper, therefore we will not focus on default core Scrum roles (i.e. Product Owner (SM), Scrum Master (SM) and Scrum Team (ST)), but only on additional roles that are essential for further consideration.

In addition to core roles, we consider the groups of people known as: managers [8] and stakeholders (i.e. “... *any group or individual, who can affect or is affected by the achievement of the organisation's objectives.*” [9]).

For the sake of proper adaptation of Scrum to work with Third Party Services in Network Organization, we propose another core role, excluded from stakeholders group:

- Third Party Service Provider (S): organization or individual who provide your organization with specialized third party services (e.g. lawyers, accountants, coaches, consultants, translators, internal and external service providers, etc.).

From our point of view, this new role S is crucial for the success of the Scrum performed in Network Organizations. This role should be involved in the entire software development process.

### IV. HOLISTIC NATURE OF SCRUM

#### A. Holons and Holarchies

“*Living systems are organized in such a way that they form multi-levelled structures, each level consisting of sub-systems which are wholes in regard to their parts, and parts with respect to the larger wholes.*” [10]

The idea of holon entity was introduced by Arthur Koestler in [11]. He coined the term “*holon*” for those entities, which might be simultaneously a part and a whole [2].

We can imagine that each holon has two opposite habits (tendencies): an integrative habitude to exist as a part of compound system and a self assertive habitude to preserve its individuality. Those two tendencies are complementary, although they also are opposite. The balance between habits is not static, but is adapting based on influence of two complementary tendencies. This makes the whole system open to change and very adaptive.

Thirty years after Koestler's original idea, another philosopher Ken Wilber generalized the idea of holons by highlighting its relative and conceptual nature. In [12], he considered that holon must have four basic characteristics [3]:



- Self-preservation: to maintain own structure, independently of the material that holon is made of.
- Self-adaptation (community): to adapt and link up with other superordinate holons, in order to react biologically, mechanically or intentionally to their stimuli.
- Self-transcendence: the holon has its own characteristics and qualities, which are new and emerging; new properties emerge in superordinate holons and create new classes of holons.
- Self-dissolution: the holons break up along the same vertical lines that they are formed.

Due to their nature, holons are connected to other holons in a typical vertical arborising structure known as a holarchy, which can be viewed as multilayer system, with tree-structure. From the holonic point of view, each member of the organization (e.g. Network Organization), can be considered a holon. It means that each member is a whole if observed as a separate unit and a part if looked at as a member of larger organization. Therefore, the core and ancillary Scrum roles can be interpreted as holons forming a holarchy, when taking into account communication network designed between holons.

*B. Holistic View of Scrum and Third Party Services in Network Organizations*

*“The holonic view of the world forms a middle ground between atomism and holism and the holonic structures form a middle-ground between network and hierarchic structures. The stratified order of holonic layers resembles a hierarchy of layers and allows flat networks within layer, but it is different from both. The stratified order is not about rigid transfer of control or about free interconnectedness of nodes, but it is rather about the self-organization of complexity and adaptation.”* [13]

Considering Network Organization as a network of intercommunicating elements, we can easily show that the amount of communication paths grows exponentially with addition of new elements [14].

Because network is an overly complex structure, we need some form of hierarchy with some aspects of superiority between elements. Holarchies seem to be the most suitable structures to manage complexity due to their special form of stratified hierarchy without traces of ranking between its elements (holons) and without cycles.

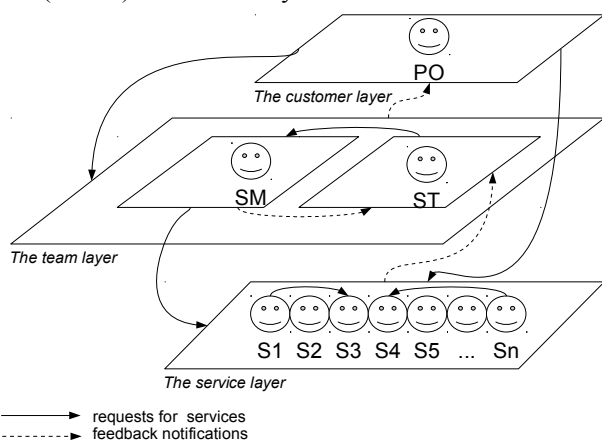


Fig 3: Scrum based model of Information Network

We propose simple three layers model of holarchy (“Fig. 3”), where default core roles(PO, SM, ST) are placed in first two layers and the new core role(S) is placed in the lowest layer. In this model, we skip ancillary roles of Managers and Stakeholders, because of low importance for further consideration.

Layers in a holarchy have an autonomy that enables them to adapt to new circumstances and changes in the environment. All communication that represents request for third party services is downward. Superordinate layers depend on the sub-ordinate layers for third party services, however not vice-versa. The lower layers inform about its state changes by providing the feedback to requesters, possibly but not necessarily from the upper layers.

Wherein our model all dependency relationships between layers are downward, and upper communication is only by providing the feedback, that helps in avoiding cycles of messages and makes communication more efficient.

The proposed model consists of three layers that refer to core Scrum roles:

- The customer layer: with Product Owner (PO) as the main requester. Third party services might be requested directly from this layer, however results (feedback) will be delivered to ST layer.
- The team layer: includes two sub-layers where entities (holons) are Scrum Master (SM) and Scrum Team (ST). ST receives results from S, requested by PO or SM.
- The service layer: this layer represents third party services and third party services providers (S). It is possible that entities in this layer will have interconnectedness between each other (e.g. some S might request services from other S).

The same solution can be used in software development of distributed projects what is shown in “Fig. 4”.

The difference occurs in additional interconnectedness between entities from middle-layer (e.g. Scrum of Scrums Meeting may be represented as SM request service from another SM). In addition to model proposed in “Fig. 3” the middle-layer is represented by two multi-entity sub-layers of SM's and ST's with interconnectedness between them. The main information flow has been kept without any changes.

We propose the model, which supports building trust, multi-culture and massive design (section II.A.). This is very important when technology and markets are changing very fast. The impact of S on the “The Team Layer” results is essential and should not be omitted (e.g. estimates proposed by ST during Sprint Planning should take into account S and dependencies associated with delivering results of third party services - delays in layer S affect results from layer ST).

*C. Scrum Artifacts*

In this paper we assume that the reader is already familiar with Scrum [1][8][15][16][17][18][19][20][21], therefore we describe only those Scrum artifacts that we propose to change to work better in Network Organizations:

- Task-feasibility instead of time-estimation: we skip using formal time-estimates and try to commit only those User Stories that we are able to deliver before next

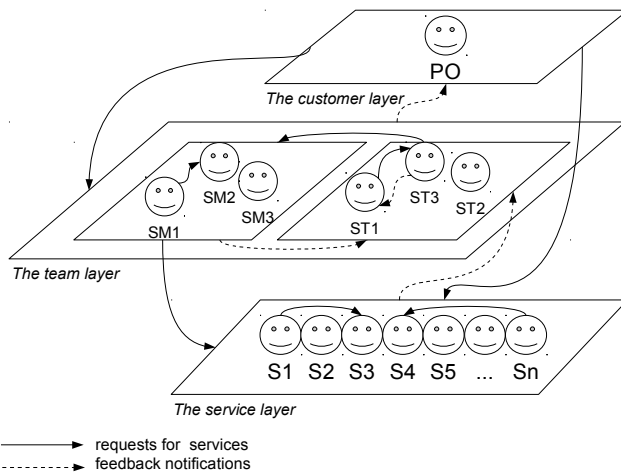


Fig 4: Scrum based model of Information Network in distributed projects

demo session. Through this change, we want to limit commitment, which we are not able to provide.

- Report Meeting instead of Sprint Review Meeting: because regular Sprint Planning session involves many resources (i.e. a lot of participants), we propose to limit participants only to representatives of the customer and the team. In our opinion that kind of meeting should be held more frequently than Sprint Review Meeting (e.g. every week) in order to improve information flow between the customer and the team.
- Planning on demand instead of Sprint Planning Meeting: because we skipped time-estimations we also propose to limit number of Sprint Planning Meetings and hold them only if really needed (i.e. when the customer needs help from the team, because is not able to prioritize Product Backlog without an additional Team's expertise).

We agree with Scrum advocates that planning up-front the Sprint is very important. However we suggest limiting this action only to prioritizing the scope (i.e. Product Backlog), so that the most important User Stories are on top of the Product Backlog list. Of course, the team is committed to delivering proposal of items, feasible for the next Sprint, but without specifying how much time will spend on each item.

Instead of regular Sprint Review Meeting held after each cycle, we propose more frequent Report Meeting (e.g. Weekly Meeting). During this meeting, the team representative is reporting actual status and points out all unsolved issues (i.e. impediments). The shortest but more frequent meetings, with reduced number of participants result in better communication and better overall understanding of project goals.

Skipping time-estimates interferes with regular Scrum Planning Meeting. Therefore, as an alternative, we propose organizing planning sessions on demand, only if required (i.e. planning the scope of next release) instead of time-consuming regular meeting before each cycle.

The proposed changes arise from the fact that we adopted a holistic view (mentioned in section IV.B.) that reduces the dependencies between the layers (e.g. no information cycles, stable workload during the Sprint, etc.). In our approach the feedback notifications from S to PO must go through "the

team layer", thus ensuring that ST and SM are fully involved in providing deliveries. For instance, the ST will not be able to deliver implementation of new wizard until they get all required translated texts from translators (i.e. S), so this implies that ST and SM must keep an eye on S and their deliveries.

#### D. Key Performance Indicators

"A performance indicator can be defined as an item of information collected at regular intervals to track the performance of a system." [22]

Within original Scrum we use only one metric (i.e. indicator). This is the time-estimate of the amount of remaining work that needs to be done versus amount of User Stories or Tasks that are set as "done" in Sprint Backlog [21]. We propose to use the following KPI's (i.e. Key Performance Indicators) that help better control software development in Network Organization:

- Reliability: to measure if the team is delivering what they said they will. We compare the difference between the amount of committed Story Points ( $c_i$ ) and delivered Story Points ( $d_i$ ) like shown in (1). The values might be presented as the percent of reliability calculated per Sprint ( $R_i$ ).

$$R_i = \frac{c_i}{d_i} * 100\% \quad (1)$$

- Productivity: to measure project velocity. We measure amount of fixed bugs ( $b_i$ ) and newly implemented requirements ( $s_i$ ) like shown in (2). The value of productivity ( $P_i$ ) should be calculated after each Sprint.

$$P_i = b_i + s_i \quad (2)$$

- Effectiveness: to measure effectiveness of testing service. The measure includes the amount of defects delivered to the customer. Based on this KPI we can calculate the effectiveness of internal testing service (like shown in (3)), by measuring the ratio between all found defects and those found by external S providing complementary testing. This shows effectiveness ( $E_i$ ) of software development team and testing services.

$$E_i = \frac{a_i - e_i}{a_i} * 100\% \quad (3)$$

We believe that the introduced KPI's are crucial to maintaining customer satisfaction. From our point of view, the required data should be collected at the end of each Sprint. We would like to point out that the same KPI's can be measured for S (section III.B.) and their findings can be used by the team for increasing customer satisfaction and for coordinating and controlling workload status.

#### V. CASE STUDY

Our approach to using Scrum in Network Organization is best illustrated by a real life example of two similar projects managed in two different variants of Scrum:

- In the project A: the pure implementation of Scrum.
- In the project B: the Scrum extended to our guidelines.

The company used in this study is a large multi-national organization (over 17000 employees all over the world) specialising in R&D services, telecommunication and mobile solutions. The customer is a large multi-national organization specialising in telecommunication and mobile solutions. The contract between companies was a typical outsourcing service.

A. Structure and Scope of Projects

The project A was a software application, dedicated for care centres for upgrading mobile device software via computer. It was a user-friendly application with an ergonomic presentation layer (UI) and very complex middle-layer to handle hundreds of different variants of mobile devices. The presentation and middle-layer text contents were localised for 40 different languages.

The project B was very similar to the project A, however it was not a stand-alone application, but a part of bigger software application (i.e. software update plug-in) dedicated for end users managing their mobile devices via computer. The plug-in was using the same middle-layer as in project A. The presentation layer was also represented by a user friendly UI localized for 40 languages.

At the time we conducted the case study, the two projects were in maintenance and support phase (i.e. about 70% of workload was bug-fixing and 30% implementation of new functionalities), so the case study did not relate to implementation from the scratch but to maintenance of the existing products.

The projects started with a short initiation phase during which the product backlog was set, general architecture was established, Scrum roles were assigned, and Scrum principles were known for all involved persons.

Iteration length was set at two week interval throughout the project, therefore after first two weeks of project initiation, the team finished the first iteration (i.e. Sprint 0).

In both cases A and B the Scrum Teams were following engineering practices (referred to in section III.A.).

B. Roles Assignment

Both projects were using services S (section III.B.), such as: translations, complementary external testing service, User eXperience expertise service.

The team composition had not changed since its inception, and for the moment of data collection, the number of persons in software development teams was the same for both projects: three software engineers and one test engineer.

The software development teams and Scrum Masters were co-located (i.e. based in Poland) that allowed building personal relationship between all team members inside their teams.

The customer (i.e. Product Owner) was remotely involved in the project, due to different location (i.e. based in Finland).

C. Meetings

The project A was following all Scrum principles; therefore, all meetings were held as defined.

The project B was performed as we suggest in this paper: instead of Sprint Review was Weekly Meeting and instead of regular Sprint Planning meeting was Planning Meeting on demand.

D. KPI's Results

We measured reliability, productivity and effectiveness values, known as KPI's (introduced in section IV.D.) to compare approaches taken by teams A and B. All presented metrics were collected for 13 consecutive Sprints (i.e. each Sprint was two weeks long). We present results of measuring reliability in "Fig.6", by comparing deviation of the reliability calculated for both projects (i.e. A and B). In both projects expected values were 100% of reliability per each Sprint.

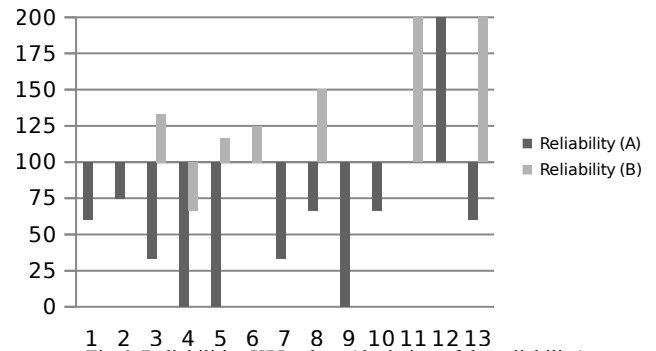


Fig 6: Reliabilities KPI values (deviation of the reliability)

We note that the approach taken by team A, was affected by large number of cases that the team promised but it was not able to deliver committed stories. We also note that our approach was not "the silver bullet". Therefore, in "Fig.6" the bars are shown to indicate when team promised less than was able to deliver.

The approach taken by team B (i.e. ~46%) was three times more accurate than approach taken by team A (i.e. ~15%).

We checked the productivity of teams A and B. The results are presented in "Fig. 5". The amount of work realized by the teams (i.e. fixed defects and implemented user stories) was comparable and their values depended on the release scope.

Because the quality was crucial for both projects we measured effectiveness of our internal testing service (i.e. internal S) and compared the amount of found defects to effectiveness of external testing service (i.e. external S). In "Fig. 7" we present the ratio of leaked defects to the number of defects found by our internal testing service.

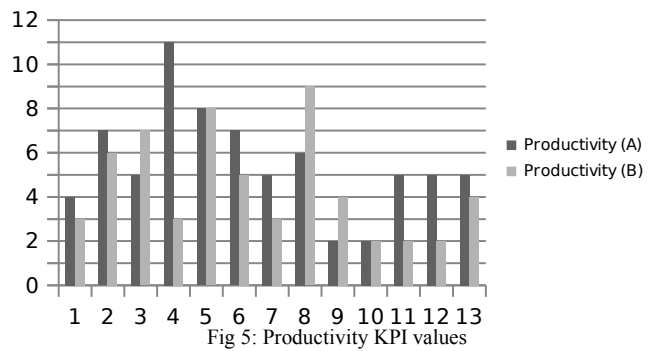
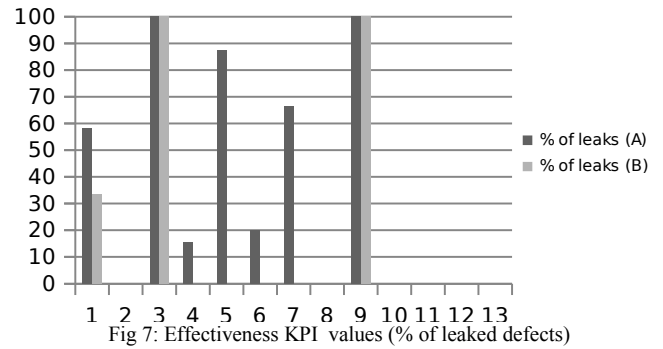


Fig 5: Productivity KPI values

TABLE I.  
CUSTOMER SATISFACTION SURVEY – QUESTION FORM

ID	Questions	Project A Scores (1-4)	Project B Scores (1-4)
<b>Project Quality:</b>			
1	What was the quality of project work in general?	3	4
2	How good was schedule estimation accuracy (timeliness)?	3	4
3	How good was effort estimation accuracy?	3	N/A
4	Tasks in the project were professionally accomplished?	3	4
5	The supplier managed the delivery well (delivery management and control)?	3	4
6	Possible changes in project personnel did not have any effect on the delivery?	N/A	4
7	Project's actual total cost corresponds to the expectations in the beginning of the project?	N/A	4
8	Work as a whole correspond to my expectations	3	4
<b>Project Personnel:</b>			
9	Project personnel accomplished their issues/tasks as promised?	3	4
10	Project personnel had a good knowledge of their own professional area?	3	3
11	Project personnel were easy to reach?	4	4
12	Project personnel worked professionally and efficiently?	4	3
13	Project personnel were genuinely interested in solving my issues and/or my problems?	4	4
14	Project personnel had the ability to cooperate?	3	4
15	Project Manager was reliable?	4	4
16	Communication between the project personnel was fluent and there were no information barriers in place?	3	4
17	Project personnel as a whole correspond to my expectations?	3	4
<b>Customer Input:</b>			
18	As the customer, we were able to accomplish our won tasks and obligations as promised?	3	3
19	As the customer, we were able to provide enough time for the delivery?	3	2
20	As the customer, we were satisfied with our won specifications in the beginning of the project?	N/A	3
21	As a customer, we were successful in guiding the third parties (e.g. the subcontractors that were our responsibility)?	4	4
<b>Average (Questions 1-17):</b>		3,26	3,88
<b>Average for customer input (questions 18-21):</b>		3,33	3,00
<b>Overall average:</b>		3,27	3,70



We found that internal service testing was more effective in project B than A. This means that the external testing service found more defects in project A than B.

#### E. Survey Results

After every six months, the customer satisfaction was monitored by internal Customer Satisfaction Survey (CSS).

During the interview the customer evaluates the level of satisfaction in three different areas (questions and areas are presented in Table I) by assigning the score from 1 (i.e. very unsatisfied) to 4 (i.e. very satisfied). The customer was able to comment on all questions, by providing additional feedback.

The scores and comments collected during CSS sessions are a kind of justification for what we have noticed during measuring KPI's. The customer feedback is always welcome and has high importance for further improvements of entire process.

The average results from both CSS's were quite impressive and show that both teams and projects were managed very well.

We note that all sections in CSS were scored higher for team B. However team A also collected high values.

#### F. Findings

To compare these two projects we used data collected regularly after each Sprint to measure KPI's (section IV.D.) and survey data collected after six months period (i.e. Customer Satisfaction Survey) to measure customer satisfaction.

We noticed that switching to "task-feasibility" from "time-estimation" (section IV.C.) and taking into account the possibility of delays caused by S (section III.B.) influence positively the amount of "empty promises".

This finding has been also noticed in high values of CSS in question 2, where the customer was more satisfied from schedule estimation accuracy in project B (e.g. the customer commented that: "it was very helpful in planning scope of software application releases").

The customer was disappointed due to bad effort estimations in project A, because that affected the scope (timeliness) of scheduled releases. In the project B the customer was satisfied and got positive feeling that schedule estimations were very good, although time-estimations were not applicable in this approach. The team limited their estimations only to committing that the proposed tasks will be finished and included in next release. Dissatisfaction of the customer

was also visible in question 9 in the section “Project Personnel” (i.e. “*Project personnel accomplished their issues/task as promised?*”), where the team A collected lower score than B – probably due to bad estimations.

In both cases (i.e. both projects A and B) the most crucial were quality and timeliness. The key incentives for the client were high visibility (transparency) and an empirical project control that Scrum delivers. Based on the presented results we can conclude that in general the customer satisfaction was higher in the project B.

## VI. RELATED WORK

During the last 30 years, many approaches have been proposed to software application development. Starting from “Code and Fix” [1][23], to the waterfall model, spiral model, rapid prototyping, incremental model, extreme programming, scrum, etc. These can be classified as representatives of general life cycle models: heavyweight, middleweight, and lightweight. We cannot say that one model is better than another, because the approach suitable for one project may not be suitable for another. However we can assume, on the basis of the research results and experience [15][16][17][18][19], that the best choice for almost all kinds of software development projects, starting from scratch and executed in continuously changing environment, is an adaptive and flexible life cycle model (i.e. Agile) and a strongly prescriptive method (e.g. Scrum, eXtreme Programming, etc.) [15][17][18]. We agree with Scrum advocates that using time-boxed delivery cycles (i.e. Sprints), visualization of the project scope (i.e. Product Backlog), prescribed roles (e.g. Scrum Master, Product Owner, Scrum Team), essential meetings (e.g. Sprint Retrospective, Sprint Review, Daily Meetings), and following Scrum rules is necessary for project's success.

However, conventional Scrum is not sufficient when daily work results depend on external third party services (e.g. Translation, Testing, Technical Support, etc.) as it is the case in “*Network Environment*”. In contemporary landscape in which Network Organizations are more and more popular and IT projects are realized in inter-cultural environment, distributed services multiply the risks and uncertainties. That makes some Scrum artifacts useless, because of dependencies between third party services and service providers.

In literature, a few researchers have already studied the way to adapt the agile practices in Network Organizations.

We observe that usually Agile software development methods are introduced as a set of principles that need to clarify a lot of different interpretations of Agile Manifesto [24]. Because of that it is very difficult to say what exactly is an Agile methodology and how to adapt it to Network Organization environment, usually based on the Scrum as an example of Agile methods.

The originators of Scrum in software development are Hirotaka Takeuchi and Ikujiro Nonaka. In [20] they defined a new approach to software development, called: rugby approach. They presented the whole process as performed by one cross-functional team across multiple phases, where the team tries to “*go the distance as a unit, passing the ball back and forth*” [20].

We consider that Scrum, as one of the representatives of Agile life cycle model, matches up perfectly to principles of “*Manifesto for Agile Software Development*” [17][18][24]. Following the Agile Manifesto principles is possible only because Scrum defines precisely the essential roles, principles and artifacts that make it very prescriptive method for managing software development teams.

We agree with Scrum advocates, but in addition we propose to add one new role (Third Party Services providers) for adapting Scrum and Third Party Services to Network Organization.

Piero Mella studied the holonic perspective in organization and management. In [3] he examined six different examples of holonic networks in terms of manufacturing systems. This paper was an inspiration to consider the Network Organization as Holonic Network, seen as “*comprised of autonomous firms that are variously located—characterized by different roles and different operations and connected through an holonic network, real or virtual, often oriented, in order to achieve a common objective through the sharing of resources, information, and necessary competencies*” [3].

Dirk S. Hovorka and Kai R. Larsen in [25] present the study that examined the influence of network organization environment on the ability to develop agile adoption practices. They use exploratory case study design to “*investigate the interactions between network structure, social information processing, organizational similarity (homophily), and absorptive capacity during the adoption of a large-scale IT system in two network organization environments*” [25]. They propose Agile Adoption Practice Model (i.e. APM) that proposes interactions within the inter-organizational network that enable Agile adoption practices. We adopted a more detailed approach, and instead of treating Agile life cycle model as a set of good practices, we proposed a more detailed analysis of one selected method of software development (i.e. Scrum) and propose Scrum-based model that suits better Network Organizations.

Wojciech Cellary and Willy Picard presented in [26] the way to achieve agility and pro-activity by introducing the model of Collaborative Network Organizations in its two forms: Virtual Organizations (VO) and Virtual Organization Breeding Environments (VOBE). They presented idea of public administration “*playing a role of Virtual Organization customer on the one hand, and a Virtual Organization member on the other hand*” [26]. This publication was a stimulus for reflection about third party service providers (i.e. S role) as an entity that might be simultaneously a part and a hole in Network Organizations.

## VII. CONCLUSION

In this paper, we proposed a holonic view based on which we adapted Scrum for 3<sup>rd</sup> part services and the Network Organization.

We developed a new service layer in the holonic structure and recommended new Scrum principles. In an industrial case study, we demonstrated the advantages of our model and method.

We believe that big differences in the level of satisfaction of the customer using our approach were caused by very prescriptive way of working with the pure Scrum.

There is no easy way to adapt Scrum software development method to work in Network Organization; however, we believe that presented results will serve to advance research and help in finding the best solution.

#### REFERENCES

- [1] V. Guntamukkala, J. H. Wen, M. J. Tarn, "An empirical study of selecting software development life cycle models", *Human Systems Management*, vol. 25, no. 4/2006, pp. 268-278, November 2006.
- [2] L. A. Maciaszek, Modeling and Engineering Adaptive Complex Systems, *Challenges in Conceptual Modeling*, in *Proc. Tutorials, Posters, Panels and Industrial Contributions to the 26th International Conference on Conceptual Modeling - ER 2007, CRPIT*, no. 83, Auckland, New Zealand, ed. J. Grundy, S. Hartmann, L. Laender, L. Maciaszek, J. Roddick, ACS, November 2007, pp. 31-38
- [3] P. Mella, "The Holonic Perspective in Management and Manufacturing", *International Management Review*, vol. 5, no. 1, pp. 19-30, 2009.
- [4] L. D. Sailer, "Structural Equivalence: Meaning and Definition, Computation and Application.", *Social Networks*, vol. 1, no. 1, pp. 73-90, 1978.
- [5] M. Van Alstyne, "The State of Network Organization: A Survey In Three Frameworks", *Organizational Computing*, vol. 7, no. 3, pp. 88-151, 1997.
- [6] M. James, D. R. White, "Structural Cohesion and Embeddedness: A Hierarchical Concept of Social Groups.", *American Sociological Review*, vol. 68, no. 1, pp. 103-127, 2003.
- [7] G. A. Rummier, A. P. Brache, *Improving Performance - How to Manage the White Space in the Organization Chart*, California, 350 Sansome Street, Jossey Bass Inc., 1995.
- [8] P. Deemer, G. Benefield, *The Scrum Primer: An Introduction to Agile Project Management with Scrum.*, GoodAgile, Version 1.04., 2007.
- [9] H. Sharp, A. Finkelstein, G. Galal, "Stakeholder identification in the requirements engineering process", in *Proc. Database and Expert Systems Applications, 1999. Proceedings.*, Florence, Italy, IEEE Computer Society Press, September 1999, pp. 387-391.
- [10] F. Capra, *The Turning Point*. Science Society, and the Rising Culture, New York, USA, Flamingo, 1982, pp. 27.
- [11] A. Koestler, *The Ghost in the Machine*, London, England, Penguin Group, 1967.
- [12] K. Wilber, *A brief history of everything*, Boston, Massachusetts 02115, Shambhala Publications, 2000.
- [13] L. A. Maciaszek, "An Investigation of Software Holons - The 'adHOCS' Approach", *Argumenta Oeconomica*, vol. 1-2, no. 19, pp. 1-40, 2007.
- [14] L. A. Maciaszek, Architecture-Centric Software Quality Management, *Web Information Systems and Technologies, WEBIST 2008, LNBIP 18*, ed. J. Cordeiro, S. Hammoudi, J. Filipe, Springer-Verlag, Berlin Heidelberg, 2009, pp. 11-26.
- [15] A. Cockburn, "Selecting a project's methodology", *IEEE Software*, vol. 4, no. 17, pp. 64-71, July - August 2000.
- [16] J. Nandhakumar, J. Avison, "The fiction of methodological development: A field study of information system development", *Information Technology and people*, vol. 2, no. 12, pp. 176-191, 1999.
- [17] K. Schwaber, M. Beedle, *Agile Software Development with Scrum*, Upper Saddle River, New Jersey, USA, Prentice Hall, 2002.
- [18] K. Schwaber, *Agile Project Management with Scrum*, Redmond, Washington, Microsoft Press, 2004.
- [19] M. Lindvall, V. Basili, B. Boehm, P. Costa, K. Dangle, F. Shull, R. Tesoriero, L. A. Williams, M. V. Zelkowitz, "Empirical Findings in Agile Methods", in *Proc. Second XP Universe and First Agile Universe Conference on Extreme Programming and Agile Methods - XP/Agile Universe 2002*, London, England, Springer-Verlag, 2002, pp. 81-92.
- [20] T. Hirotaka, N. Ikujiro, *The New New Product Development Game*, Harvard Business Review, vol. 64, January-February 1986.
- [21] N. Zabkar, V. Mahnic, "Using COBIT indicators for measuring Scrum-based software development", *WSEAS Transactions on Computers*, vol. 7, no. 10, pp. 1605-1617, 10 2008.
- [22] C. T. Fitz-Gibbon, Bera Dialogues: 2, Performance Indicators, Clevedon, England, Multilingual Matters, 1990, pp. 111.
- [23] W. Royce, "Managing the development of large software systems", in *Proc. Proceedings of the 9th international conference on Software Engineering ICSE '87*, Los Angeles, IEEE Computer Society Press, August 1970, pp. 1-9.
- [24] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, C. R. Martin, S. Mellor, K. Schwaber, J. Sutherland, D. Thomas, *Manifesto for Agile Software Development*, 2001.
- [25] D. S. Hovorka, K. R. Larsen, "Enabling agile adoption practices through network organization", *European Journal of Information Systems - Including a special section on business agility and diffusion of information technology*, vol. 15, no. 2, pp. 159-168, April 2006.
- [26] W. Cellary, W. Picard, "Agile and Pro-Active Public Administration as Collaborative Networked Organization", presented at the International Conference on Theory and Practice of Electronic Governance ICEGOV'10, ACM, NEW York, 2010, 978-1-4503-0058-2.

# Extending the Descartes Specification Language Towards Process Modeling

Joseph E. Urban<sup>3</sup>, Vinitha H. Subburaj<sup>1</sup>, Lavanya Ramamoorthy<sup>1</sup>

<sup>1</sup>Computer Science Department

<sup>3</sup>Industrial Engineering Department

Texas Tech University

Lubbock, Texas 79409 USA

{vinitha.subburaj,joseph.urban,lavanya.ramamoorthy}@ttu.edu

**Abstract**—With current complex real time software problems, the need for reliable software specification becomes crucial. This paper overviews the use of formal methods to specify requirements and the advantage of using an executable formal specification language processor to develop a process model for the development of a software system. The paper presents how a software process can be described using the Descartes specification language, an executable specification language, and the language extensions made to Descartes to make it suitable to describe a software process.

**Index terms**—software specification; software process model;

## I. INTRODUCTION

Software development is a complex and creative effort. A software process is a sequence of steps that are used to manage the development of software. A software process should be handled effectively to deliver the software on time and with good quality. Time, cost, and quality are some reasons why software development should be automated. Software process modeling focuses on what occurs during software creation and evolution. The basis for process models includes the individuals involved in the development of software, the assigned work, tools required to do the work, and the final results of the work.

Requirements analysis is a significant phase of software development because if any fault in the specification is left undetected, it can be carried over into the next phase. Thus, later correction of the fault would involve fixing the fault and fixing the effects of the fault in subsequent phases. Apart from consumption of a large amount of resources, for a change in a specification, the entire code will have to be rewritten by the developers. Also, since there is no actual system to verify the requirements provided by the user in a conventional software life cycle, development of such unverified requirements can cause errors in the program code.

The remainder of this paper is structured as follows. In Section II, related work that has been done in developing agent systems using formal methods is discussed. Section III discusses the research methodology used in this research effort. Section IV describes the extensions made to the Descartes specification language for specifying process modelling. Section V gives a comparative study with several existing methodologies. Section VI concludes the paper with a summary and future research.

## II. RELATED WORK

Software process models are expressed formally by process modeling languages (PML). Many process modeling languages have been developed. The following section briefly describes the earlier work done similar to this research effort. This section includes APER-2, CSPL, DPEL, Marvel, Merlin, VRPML, and YAWN.

APER-2 [3] is a developer centered and object-oriented process language. A process program in APER-2 is composed of classes. CSPL (Concurrent Software Process Language) [2] is a process modeling language that uses Ada95-like syntax. CSPL integrates object-oriented Ada95 for modeling support and UNIX shell scripts for enactment support.

DPEL (Decentralized Process Enactment Language) [4] is a process modeling language which is used to model activities and activity synchronization. In DPEL, modeling of a process is based on developers. The process programs are converted into DPEL segments for enactment in DPEM. Marvel [8, 9, 1] is a process modeling technique based on rules. Marvel has three types of rules: project rule set, project type set, and project tool set. Process specific issues are described by project rule set. Project type set is used to specify the data with object-oriented classes.

MERLIN [7, 6] has a knowledge base which describes a process that is built using a rule based technique. Rules and facts are interpreted as forward and backward chaining. Backward rules and facts are interpreted in a Prolog-like manner. VRPML (Virtual reality process modeling language) [12] is a process modeling language that uses graphs to specify a soft-

ware process. YAWN (Yet Another Workflow Notation) [11] is a graphical PML which uses directed graphs to represent a process interaction model.

A process model should enable effective communication, facilitate reuse, support evolution, and facilitate management [5]. Extensions were made to Descartes, such that it represents all the process entities and their relationships. The Descartes constructs are to support the basic elements of the software process.

Descartes [14] is an executable specification language that is based on three data structuring methods proposed by Hoare: direct product, discriminated union, and sequence. The language uses a tree structure notation to perform analysis and synthesis within a specification.

### III. EXTENSIONS TO THE DESCARTES SPECIFICATION LANGUAGE FOR PROCESS MODELING

In order to describe a software process, basic elements of the software process must be modeled. The basic elements of the software process are activities, products, role, human, and tool. The following describes extensions to Descartes that support software processes.

#### A. Activities and products

An activity is a step of a process that produces changes to the software product. The product is the set of artifacts developed and maintained in a project. The product is the input and output of an activity. A pre-pended unary reserve word “activity” is included in a Descartes module for declaring an activity module. After the module title, the module contains the specification for the other process elements. The input PRODUCTS consist of software products.

```
activity ACTIVITY_NAME_USING_(PRODUCTS)
  PRODUCTS
    inputs
```

#### B. Human actor

An actor is a person who is responsible for a software process activity. The reserved keyword “actor” is used to specify the person responsible for that activity. Consider the example of “modify\_design” activity which is performed by the actor “design\_eng1”. The “design\_eng1” is responsible for performing the activity “modify\_design”.

```
activity MODIFY_DESIGN_USING_(PRODUCTS)
  PRODUCTS
    requirement_change
      FILE
    design_document
      FILE
  actor
    ‘design_eng1’
```

#### C. Role

A role is the rights and responsibility of a person who is

going to perform a software process activity. The “role” construct introduced by Medina and Urban [10] is used for specifying the role of the actor in an activity.

```
activity MODIFY_DESIGN_USING_(PRODUCTS)
  PRODUCTS
    requirement_change
      FILE
    design_document
      FILE
  actor
    ‘design_eng1’
  role
    ‘design_engineer’
```

In the above example, the activity “modify\_design” is performed by a “design\_eng1” who is a “design\_engineer”.

#### D. Tool

The tools used in the software production, such as textual editors and case tools, should be represented. The reserved keyword “tools” is used to specify the tools used in that activity. Consider the example of “modify\_activity” in which “textual\_editor” is used to edit the “design\_document”.

```
activity MODIFY_DESIGN_USING_(PRODUCTS)
  PRODUCTS
    requirement_change
      FILE
    design_document
      FILE
  actor
    ‘design_eng1’
  role
    ‘design_engineer’
  tools
    ‘textual_editor’
```

#### E. Process example

In this example, the process of designing a library management system is used. Suppose a library management system consists of two subsystems ‘subsystem1’ and ‘subsystem2’.

When the process starts, subsystem1 is designed by ‘design\_eng1’ and subsystem2 is designed by ‘design\_eng2’. Then the design of subsystem1 is reviewed by ‘design\_rev1’ and the design of subsystem2 is reviewed by ‘design\_rev2’. If the subsystem1 fails the review, the changes are made to subsystem1 by ‘design\_eng1’. If the subsystem2 fails the review, the changes are made to subsystem1 by ‘design\_eng2’. Changes are made to subsystem1 and subsystem2 until the subsystems pass the review. If subsystem1 and subsystem2 passes the review, the whole system is reviewed by the ‘design\_rev1’ and ‘design\_rev2’. If the whole system fails the review, the changes are made to the system by ‘design\_eng1’ and ‘design\_eng2’. Changes are made to the system until the system passes the review.



At the start of the process, the DES\_SS1 and the DES\_SS2 activities are started concurrently as shown in the specification below. The “parallel” construct introduced by Sung [13] is used to express the concurrent execution.

DESIGN\_LIBRARY\_SYSTEM

```

return
  parallel
    DES_SS1_USING_(PRODUCTS)
    DES_SS2_USING_(PRODUCTS)

```

In the DES\_SS1 activity, the ‘design\_eng1’ takes the requirement\_document as input and designs the subsystem1\_design\_document.

The subsystem1\_design\_document is modified using the tools ‘textual\_editor’ and ‘case\_tool’. After subsystem1 is designed, the REVIEW\_SS1\_DESIGN activity module is executed.

**activity** DES\_SS1\_USING\_(PRODUCTS)

```

PRODUCTS
  products
    requirement_document
      FILE
    ss1_design_document
      FILE

```

```

actor
  ‘design_eng1’
role
  ‘design_engineer’
tools
  ‘textual_editor’
  ‘case_tool’

```

```

return
  SS1_DESIGN_DOCUMENT
  ‘designed_by’
  ACTOR
  ‘using’
  TOOLS
  REVIEW_SS1_DESIGN_USING_(PRODUCTS)

```

In the DESIGN\_SS2 activity, the ‘design\_eng2’ takes the requirement\_document as input and designs the ss2\_design\_document. The ss2\_design\_document is modified using the tools ‘textual\_editor’ and ‘case\_tool’. After the ss2 is designed, the REVIEW\_SS2\_DESIGN activity module is executed. The specification written for the DES\_SS2\_USING\_(PRODUCTS) activity is similar to the DES\_SS1\_USING\_(PRODUCTS) activity.

In the REVIEW\_SS1\_DESIGN activity, the ‘design\_rev1’ takes the requirement\_document and the ss1\_design\_document as input and reviews the design of ss1. The design of subsystem1 is reviewed using the tool ‘textual\_editor’. If subsystem1 passes the review, the

REVIEW\_SYSTEM\_USING\_(PRODUCTS) activity is called. If subsystem1 fails the review, the MODIFY\_SS1 activity module is executed.

**activity** REVIEW\_SS1\_DESIGN\_USING\_(PRODUCTS)

```

PRODUCTS
  products
    requirement_document
      FILE
    ss1_design_document
      FILE
    review+
      review_pass
        STRING
      review_fail
        STRING
    feedback
      FILE

```

```

actor
  ‘design_rev1’
role
  ‘design_reviewer’
tools
  ‘textual_editor’

```

```

return+
  REVIEW_PASS
  SS1_DESIGN_DOCUMENT
  ‘reviewed_by’
  ACTOR
  ‘using’
  TOOLS
  ‘subsystem1_passed_the_review’
  REVIEW_SYSTEM_USING_(PRODUCTS)
  REVIEW_FAIL
  SS1_DESIGN_DOCUMENT
  ‘reviewed_by’
  ACTOR
  ‘using’
  TOOLS
  ‘ss1_failed_the_review’
  MODIFY_SS1_USING_(PRODUCTS)

```

In the REVIEW\_SS2\_DESIGN activity, the ‘design\_rev2’ takes the requirement\_document and the ss2\_design\_document as input and reviews the design of ss2. The design of ss2 is reviewed using the tool ‘textual\_editor’. If ss2 passes the review, the REVIEW\_SYSTEM\_USING\_(PRODUCTS) activity is called. If ss2 fails the review, the MODIFY\_SS2 activity module is executed. Similarly we can write the specifications for REVIEW\_SS2\_DESIGN\_USING\_(PRODUCTS) activity, MODIFY\_SS1 activity, MODIFY\_SS2 activity, REVIEW\_SS\_DESIGN\_USING\_(PRODUCTS) activity, and MODIFY\_SYSTEM activity.

## IV. COMPARISON

A process modeling language can be compared through the following criteria: interface and style. Table 1 shows the comparison of several process modeling languages based on interface and styles along with the Descartes specification language used for process modeling.

Process modeling languages	Interface		Style			
	Graphical	Textual	Rule based	Object Modeling	State Transitions and petrinets	Programming language
APER-2		X		X		
CSPL		X				X
DPEL		X				X
Marvel		X	X			
MERLIN		X	X			
VRPML	X				X	
YAWN	X				X	
Descartes Specification Language		X	Formal Specification Style			

Table 1 Comparison of process modeling languages

The Descartes specification language could be used in both specification development and for describing a software process. With this advantage, the Descartes specification language has an edge over the other process modeling languages mentioned because, time and cost in training the personnel could be saved by using the Descartes specification language in both specification development and for describing a software process.

## V. SUMMARY AND FUTURE WORK

Managing a process manually will consume more time, cost more, and can result in low quality software. Thus, automation of a software process will save time and reduce extra work. This paper introduced the extended Descartes specification language for automating a software process. Extensions were made to the Descartes specification language for describing a software process. Extensions made to the Descartes specification language were identified to be helpful in modeling the basic elements of a software process.

The future research effort can focus on providing tool support for designing, modifying, analyzing, simulating, and verifying a process. Tool support would be more helpful for a user to organize the work and to keep track of what is going on in a

process. A tool could be provided for designing and modifying the specifications for a process. Simulating a process specification before it is executed could help in checking whether the process performs what is intended. At present, the Descartes specification language does not support simulation. Future research can concentrate on adding a simulation feature to the Descartes specification language. The tool support could be provided for simulation.

## REFERENCES

- [1] N. Barguthi and G. Kaiser, "Scaling up Rule-based Software Development Environment," *Proceedings of the 3<sup>rd</sup> European Software Engineering Conference (ESEC'91)*, Milan, Italy, Oct 1991, pp. 380-395.
- [2] J. J. Chen, "CSPL: An Ada95-Like, Unix-Based Process Environment," *IEEE Transactions on Software Engineering*, Vol. 23, No. 3, March 1997, pp. 171-184.
- [3] J. Y. Chen, S. C. Chou, and W. C. Liu, "APER-2: A Developer Centered, Object-Oriented Process Language," *Proceedings of the International Symposium on Multimedia Software Engineering*, December 2000, pp. 297-303.
- [4] S. C. Chou, "DPEM: A Decentralized Software Process Enactment Model," *Information and Software Technology* 46, 2004, pp. 383-395.
- [5] B. Curtis, M. I. Kellner, and J. Over, "Process Modeling," *Communications of the ACM*, Vol. 35, No. 9, September 1992, pp. 75-90.
- [6] W. Emmerich, G. Junkermann, and W. Schafer, "MERLIN: Knowledge Based Process Modeling," *Proceedings of the First European Workshop on Software Process Modeling*, Milan, Italy, May 1991, pp. 181-187.
- [7] H. Hunnekens, G. Junkermann, B. Peuschel, W. Schafer, and J. Vagts, "A Step Towards Knowledge-Based Software Process Modeling," *Proceedings of the First Conference on System Development Environments and Factories*, 1990, pp. 49-58.
- [8] G. Kaiser, N. Barguthi, and M. Sokolsky, "Preliminary Experience With Process Modeling in the Marvel SDE Kernel," *Proceedings of the IEEE 23<sup>rd</sup> Hawaii ICSS*, Software Track, 1990, pp. 131-140.
- [9] G. Kaiser and P. Feiler, "An Architecture for Intelligent Assistance in Software Development," *Proceedings of the 9<sup>th</sup> International Conference on Software Engineering* Monterey, April 1987, pp. 180-188.
- [10] M. A. Medina and J. E. Urban, "An Approach to Deriving Reactive Agent Designs from Extensions to the Descartes Specification Language," *Proceedings of the Eight International Symposium on Autonomous Decentralized Systems*, March 21-23, 2007, pp. 363-367.
- [11] D. Rossi and E. Turrini, "Using a Process Modeling Language for the Design and Implementation of Process-Driven Applications," *International Conference on Software Engineering Advances (ICSEA)*, 2007, pp. 55-61.
- [12] K. Z. Zamli, "The Design and Implementation of the VRPML Support Environments," *Malaysia Journal of Computer Science*, Vol. 18, No. 1, June 2005, pp. 57-69.
- [13] K.-Y. Sung and J. E. Urban, "A Real-Time Specification Method for Specifying and Validating Real-Time Concurrent Systems," *Proceedings of the Twelfth IEEE International Phoenix Conference on Computers and Communications*, Tempe, Arizona, March 24-26, 1993, pp. 578-584.
- [14] V. H. Subburaj and J. E. Urban, "Issues and Challenges in Building a Framework for Reactive Agent Systems," *Proceedings of the Third International Workshop on Frontiers in Complex Intelligent and Software Intensive Systems (FCISIS-2010)*, Krakow, Poland, February 15-18, 2010.

## Influence of search engines on customer decision process

Marek Zgódka

Warsaw School of Economics -  
Szkoła Główna Handlowa  
w Warszawie

Al. Niepodległości 162,  
02-554 Warsaw, Poland

Email: marek.zgodka@gmail.com

**Abstract**—This article summarizes customer decision process focusing on information search. It explains the role and use communication channels that are used during information search, particularly the internet. It describes the internet and the role of search engines during information search. It explains the use of search engines and provides better understanding of ways in which search engines support customer decision process such as reduction of information search cost, higher involvement in the search process and increased ability to search for information. It also identifies some possible disadvantages like information irrelevancy or invisible web. Paper aims to identify the influence of search engines on information search phase of customer decision process.

### I. DECISION PROCESS

ONE of important issues in research on customer behaviour is observation, analysis and search of the values that influence customers during the purchasing decision process. In order to better understand the process, there have been distinguished phases of decision making process.

According to Engel J. F., Kollat D. T. and Blackwell R. D. decision making model consist of following phases: [1]

- A. Need Recognition
- B. Information Search
- C. Evaluation of Alternatives
- D. Purchase
- E. Post-Purchase Evaluation

When unsatisfied need is recognized, the customer begins searching for information. R. E. Rice, M. McCreadie and Chang S. L. define information as a "commodity or resource, and part of the communication process." [2] G. Stigler in 1961 said that knowledge is power and one should hardly have to tell academicians that information is a valuable resource. [3] Until then, conventional economics assumed unlimited access to information and ability to obtain information at no cost. Information search is a process during which consumers look for relevant information to make a reasoned decision. Consumers want to gain a better product and/or a better price and try to make the most optimal decision. They can seek for information on prices or characteristics of service/ product. Information search can be costly. The information itself can have its price, and consumers have to pay

to access it, or the process of search implies costs and time consumption.

First model of information search was presented in 1982 by G. Punj and R. Staelin. It was based on the assumption, that consumers are looking for the best and richest source of information, in order to make satisfactory purchase decision. They have examined five variables: knowledge and experience, need and efficiency of exploration, the market environment as an acceptable set size, customer satisfaction and cost-saving as a measure of the cost of exploration and search for information.[4] In 1991, N. Srinivasan and B. Ratchford extended the above model with perceived risks and perceived benefits of information search. [5]

J. R. Bettman stated that the process of information search is made up of internal and external search for information. He claimed that consumers usually first engage in internal search for information stored in memory. Main determinants of internal search scope are: the quantity of stored information, the suitability of information and the importance of decision. Only when information available in memory is insufficient to make a decision, consumers engage in external search for information.[6] Most of searches consists both, from internal and external information search. [7]

Progress of information and communication technologies has contributed to the development of information society. Historically, the processing and distribution of information consisted of, among others: mail, newspapers, radio, telephones. A special role in information society plays the Internet, that enabled the transfer of information to match the expectations and lifestyle of the recipient. As a result of these changes, massive numbers of readers and listeners are divided into smaller groups with common goals and interests.

### II. INTERNET

The Internet has become an important source of external information, as the information traditionally available in a variety of channels, became available in a single medium - Internet, thus influencing the decision-making process [8]. According to R. A. Peterson, S. Balasubramanian and B. J. Bronnenberg Internet offers the possibility to store large amounts of information in a number of virtual locations. It allows almost unlimited access to information from anywhere. Additionally, the Internet has the advantage of efficient and effective searching, organizing and sharing gath-

ered information [12]. Provides access to information named by R. Kraut, M. Patterson, V. Lundmark, S. Kiesler as "previously unavailable" [13]. N. Nie and L. Erbring defined Internet as a "huge public library" [14].

Compared to other media, the Internet has many advantages. On the Internet, there are multiple senders of information, so we can often compare data from several sources. It should be noted that Internet traffic is highly concentrated. Only 0.5 percent of Web pages is responsible for 80 percent of Internet traffic [15].

In communication via Internet (as opposed to traditional media), the consumer (user computer) takes the initiative in the selection of information that reaches him - he decides which information is needed. Thus, in the literature, Internet is often referred to as the "pull" medium, because of the unique way of sender - recipient communication (usually unavailable in other media). Information is pulled by customers, as opposed to traditional media (television, radio, press), where the consumer is only passive recipient of message [16].

Internet provides the ubiquity of information. It allows to get information from anywhere, anytime and reduces information asymmetry by removing the existing division between the sender and the recipient, because each side is both the sender and receiver of information [17]. Information retrieval is the result of consumer active search or navigation.

### III. SEARCH ENGINES

With the development of internet technology, the amount of information available on the Internet has grown to such an extent, that navigation to desired information became difficult. These difficulties has been solved by search engines, which spider the content of the global network, and then search for Internet resources relevant to our query. According to the PWN encyclopaedia definition, search engine is a web site that allows searching web pages containing the particular keywords. R. Prytherch described the search engine as a program produced by the publisher or data provider, enabling access to its information resources by author, title or keyword [18]. M. Busby defines it as kind of browser software, which searches the resources of the Internet, identifies the contents of web pages and stores it on computer's search engine [19].

Search engines offer some additional features that improves its search capabilities. In order to offer consumers the most relevant results, search engines take into account user behaviour on the Internet – i.e. queries entered into a search engine or web pages visited. On this basis, they are able to determine the interest of a consumer and choose the correct meaning of ambiguous words. Thanks to the computer's IP number, search engines can determine its approximate location. This allows search results geo-targeting, which means returning search results that located close to an Internet user. For the purpose of this article, we will define a search engine as a tool design to find information on entered query, that takes into account the location from which the query was asked, the personal preferences of consumers, and the time of query.

Internet search engine consists of:

A. software

called a robot, which follows links on the website, updates information and adds a new documents if found. Documents that change and are modified frequently or that are very important, are visited by robots more often. This does not affect the perceived by the search engine importance of the document, only the validity of the information presented.

B. Index

that stores the documents found by the robot. Index is a repository of all documents that search engine can search. During information search, search engines do not actually search the Internet resources but search engine's index.

C. User Interface

which is responsible for the exchange of information. It is a place where customer enters a query and gets a list of search results.

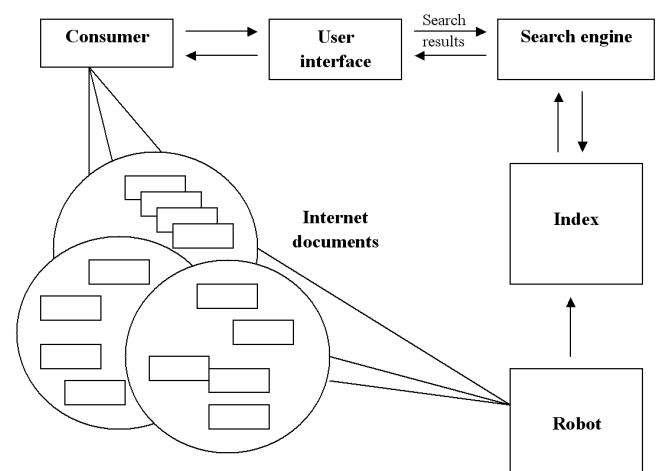


Fig. 1 Search engine architecture. Source: Own analysis

### IV. INFORMATION SEARCH ON THE INTERNET

With search engines, consumers can easily and precisely gain desired information. Use of search engine allows extremely easy and quick access to information on given topic. Internet users can find, compare and verify the information. Eric Schmidt, Google's director, described this situation as the democratization of information [21]. Through the Web, customers can very easily learn, compare and verify offers. Using so-called price search engines, they can easily compare prices and find a shop offering a selected product at the lowest price.

D.L. Hoffman i T.P. Novak distinguished two types of information search on the Internet [22]:

A. Specific information search

characterized by external motivation, focused on needs, consisting of direct search, leading to desired information.

B. General information search

characterized by internal motivation, constant involvement, consisting of indirect search and focusing on navigational choices.

## V. INFORMATION SEARCH WITH SEARCH ENGINE

Search engines enable to quickly find information on entered query, reducing the cost of information search. Yu Liang examined the long and short-term effects of reducing the information search cost. His research shows that in the short term, reduced cost of finding information will reduce the equilibrium price (The market price at which the supply of an item equals the quantity demanded). However, on the long-term basis, vendor will upgrade the product and as a result both the equilibrium price and vendors profit will increase. **Along with reduction of information search cost, consumers can look more and increase chances to find a product with better specifications.** Thus, the seller will be more motivated to offer a better product, that will have greater value for the buyer. As a result, product diversification will increase, what will reduce competition and prevent price drop to marginal cost. Reduction of information search cost will increase society welfare [25].

These advantages cause that over 85% of Internet users use search engines [26]. Typically, search engines apart from link to the website, provide some information important for the consumer. It may be the price, summarized description of a web site or phone number. This makes search engines both the source of information, as well as the way to reach more detailed information. Information access improvement by search engines depends on the importance of information obtained through the search results. Consumers use search engine to obtain the desired information. **The aim of search engine is to provide the most relevant answers to user's query.** Search engines build out their own web indexes, that collect increasing amounts of information and enhance algorithms to return the most relevant results. In a study conducted in 2000, 85% of British said that the quality of information returned by the search engine is important or very important. At the same time 31% of the examined group of respondents said that they often do not find the desired information on the web [27]. Search engine algorithms development is an ongoing process. Advances in technologies of data processing, content processing and information indexing improved information relevancy. In a study conducted in 2004, already 92% of Google users were satisfied with the results [28], what implies the constant search results improvement. **The process of information search on the web is characterized by the principle of "least effort". Users want to find information in quickest and easiest way, even at the expense of its quality** [29]. Most Internet users do not use advanced search functions, or specialized search engines, focusing on the general search engine, giving an average of two questions, consisting of two terms and only reading the first page of search results [30]. Consumers involvement in information search is associated with time spent on information access. Consumers engage in information search in order to maximize expected satisfaction with the chosen product. They compare different characteristics of products [31]. **Fast access (thanks to search engines) to desired information on the Internet may encourage the higher involvement in the search process** and offer possibility of accessing available information about the products.

Difference between search engine and web browser should be obvious. The purpose of a search engine like Google or NetSprint is to find a site for which we do not know the address. The web browser is a program to view and navigate the Web Site. We must have a browser to be able to use search engine. Nevertheless these two tools are often mistaken. In a survey conducted in six European Union countries in 2010, the question of what is a web browser, 68% of respondents said that it is a tool to access the Internet, while 40% answered that it is a search engine [32]. The above study shows that in Poland 57% of people do not distinguish between a search engine on the browser. So search engine can often replace some functionalities of web browser, giving the result of exact web site URL address and **being the first place where the consumer enters the address**, for example from a leaflet or a business card.

K. Wojcik among the determinants of consumer behaviour mentions capability of information search. The capability of information search is the consumer's own assessment of his personal capacity and motivation to perform actions associated with information search. The ability to information search is related to knowledge, experience and education of the consumer [33]. **Use of internet search engines may increase the ability to search for information** through a precise search results.

Consumers have a limited ability to process large amounts of information. Accordingly, in order to cope with information overload they make less careful choice, avoiding effort. It was confirmed by studies of K. Keller and R. Staelin [34]. They show that the quality of decisions made by consumers is deteriorating with increasing amounts of information. **Too much information on alternative choices not only does not facilitate consumer choice, but even reduces the effectiveness of this choice (information overload).** Search engines allow access to many information resources that are often not verified. This can lead to collection of large amounts of information, often difficult to process. Explained in this article disadvantages and advantages of search engines are presented in Fig. 2.

Strengths	Weaknesses
fast information access large information index reduction of information search cost increased information search ability	invisible web information irrelevancy
Opportunities	Threats
increasing information relevancy increasing information visibility	information overload

Fig. 2 Search engine influence on information search

Source: Own analysis

## VI. CONCLUSION

This article explains influence of search engine on customers decision process, particularly information search

phase. Some strengths, weaknesses, opportunities and threats have been described basing on literature, but they still require further investigation. Other possible topic that could be subject of investigation is the characteristics of different ways of search engines usage, or consumers awareness of the fact that search results hierarchy can be artificially influenced.

#### REFERENCES

- [1] Engel J.F, Kollat D.T., Blackwell R.D.: Consumer Behavior; New York; 1973; pp.58
- [2] Rice R. E., McCreedy M., Chang pp. L.: Accessing and browsing - Information and communication; Massachusetts Institute of Technology; 2001; pp. 4.
- [3] Stigler G: The economics of Information; „Journal of Political Economy”; 1961; 69; pp. 213-225
- [4] Punj G.N., Staelin R.: A Model of Consumer Information Search Behavior for New Automobiles; „Journal of Consumer Research”; 1983; 9; pp. 366-380
- [5] Srivinsan N., Ratchford B.: An Empirical Test of a Model of External Search for Automobiles; „Journal of Consumer Research”; 1991; 18; pp. 233-242
- [6] Bettman J. R., Johnson E., Payne J.: Consumer decision making, Handbook of Consumer Behavior, Prentice-Hall, 1991, pp. 107-110.
- [7] Bettman J.R., Park C.W.: Effects of Prior Knowledge and Experience and Phases of the Choice process on Consumer Decision Processes: A Protocol Analysis. “Journal of Consumer Research”; vol. 7; 1980; pp. 234-248
- [8] Klein L.R.: Evaluating the Potential of Interactive Media Thorough a New Lens: Search versus Experience Goods; “Journal of Business Research”; Vol. 41; 1998; pp. 195-203
- [9] Castells M.: Galaktyka internetu. Refleksje nad Internetem, biznesem i społeczeństwem; Dom wydawniczy Rebis; Poznań; 2003; pp. 12
- [10] Batorski D.: Wykluczenie cyfrowe w Polsce; "Studia Biura Analiz Sejmowych"; vol. 3/2009.
- [11] Peterson R.A., Merino M. C.: Consumer Information Search Behaviour and the Internet; “Psychology and Marketing”; Vol. 20 number 2; 2003
- [12] Peterson R. A., Balasubramanian pp., Bronnenberg B. J.: Exploring the implications of the Internet for consumer marketing. “Journal of the Academy of Marketing Science”; Vol. 25; 1997; pp. 329-346
- [13] Kraut R., Patterson M., Lundmark V., Kiesler pp., Mukopadhyay T., Scherlis W.: Internet paradox: A social technology that reduces social involvement and psychological well-being? “American Psychologist”; Vol. 53; 1998; pp. 1017-1031
- [14] Nie N.H., Erbring L., Internet and society: A preliminary report, The Institute for the Quantitative Study of Society, Stanford
- [15] DiMaggio P., Hargittai E., Neuman W., Robinson J.: Social implications of the internet.; “Annual Review of Sociology”; 27; 2001; pp. 307-336.
- [16] Adamczyk J.; Internet jako narzędzie komunikacji marketingowej.; T. Goban-Klas: Komunikacja marketingowa – kształtowanie społeczeństwa konsumpcyjnego?; Wyższa Szkoła Handlowa; Radom 2006; pp. 371.
- [17] Hoffman D. L., Novak T. P.; A new marketing paradigm for electronic commerce.; “The Information Society”; 13; 1997; pp.43–54.
- [18] Prytherc R.; Search Engine; Gower; 2000; pp. 657
- [19] Busby M.; Learn Google; Wordware Publishing; Plano; 2004; pp. 25
- [20] Oppenheim C., Morris A., McKnight C., Lowley S.; The evaluation of www search engines; “Journal of documentation”; 56; 2000; pp. 190-211
- [21] Schmidt E., Demokratyzacja informacji, [http://www.medianewpp.com.pl/info\\_media5119.php3](http://www.medianewpp.com.pl/info_media5119.php3).
- [22] Hoffman D. L., Novak T. P.: Marketing in hypermedia computer-mediated environments: Conceptual foundations; “Journal of Marketing”; 1996; Vol. 60; pp. 50-68
- [23] Sherman C. and Price G.: The Invisible Web: Uncovering Information Sources SearchvEngines Can’t See; CyberAge Books; Nowy Jork; 2001
- [24] Lewandowski D., Mayr P.; Exploring the academic invisible web; “Library Hi Tech”; 2006; pp. 529-539;
- [25] Mansourian Y., Ford N., Webber pp., Madden A.; An integrative model of “information visibility” and “information seeking” on the web; “Program”; vol. 42; 2008; pp. 402-417
- [26] Yu L.; Essays on search, search engine and search-based advertising; Purdue University; 02.2006; pp. 7
- [27] Neave C., Cumberland pp.; Frustration Overwhelms Internet Searchers; <http://www.ipsos-mori.com>; 07.2000
- [28] Kevin J. D.; Google's Rivals Narrow Search Gap; “Wall Street Journal – Technology Journal”; 01.2005
- [29] Griffiths J. R.; Brophy P.; Student Searching Behavior and the Web: Use of Academic Resources and Google; “Library Trends”; Vol. 53; pp. 539-554
- [30] Ford N., David M., Nicola M.; Web search strategies and retrieval effectiveness: an empirical study.; “Journal of Documentation”; Vol. 58; 2002; pp. 30-48.
- [31] Assael H.: Consumer Behavior, Wadsworth, New York 1981
- [32] Wiedza na temat wyboru przeglądarki: Ipsos MORI, 6000 respondents, 02.2010.
- [33] Wojcik K.: O niektórych teoriach informacji dla konsumentów i ich walorach aplikacyjnych.; "Narzędzia Polityki gospodarczej i społecznej w procesie kształtowania konsumpcji. Tom I. Ogólnopolska Konferencja Naukowo-Dydaktyczna Katedr i Instytutów Obrotu Towarowego i Usług Uczelni Ekonomicznych"; Ustroń, 09.1987; AE Katowice 1987, pp. 163 – 175
- [34] Keller K., Staelin R.; Effects of quality and quantity of information on decision effectiveness, “Journal of Consumer Research”, 1987, number 14, pp.200-213

# 1<sup>st</sup> International Workshop on Interoperable Healthcare Systems—Challenges, Technologies, and Trends

ACCORDING to a 2009 report published by the World Health Organization, “Globally in 2006, expenditure on health was about 8.7% of gross domestic product, with the highest level in the Americas at 12.8% and the lowest in the South-East Asia Region at 3.4%. This translates to about US\$716 per capita on the average but there is tremendous variation ranging from a very low US\$31 per capita in the South-East Asia Region to a high of US\$2636 per capita in the Americas”.<sup>1</sup>

Spending on healthcare systems worldwide continues to surge in spite of the limited number of funding bodies besides governments. These systems have to include state of the art technologies and equipments to keep up the pace with the demands of a growing population and address the risks that diseases put on the welfare of this population. The rapid widespread of some diseases and scarcity of appropriate medical facilities in some countries are examples of challenges that healthcare stakeholders face daily. In addition the lack of a common healthcare systems interoperability framework undermines regularly the efforts put into offering better services that spread over multiple stakeholders. These systems are simply not meant to collaborate making any cross-system scenario tedious and error prone.

## AIMS AND TOPICS

This workshop aims at gathering researchers from the fields of IT and healthcare to think about the obstacles that hurdle the leveraging of interoperable IT healthcare applications. We target researchers from both industry and academia to join forces in this new area. We intend to discuss the recent and significant developments in the general area of healthcare systems. In particular, we hope to identify techniques from IT like service and ubiquitous computing that will have the greatest impact on making healthcare systems collaborate.

Specific possible topics include (but not limited to):

- Service computing for interoperable healthcare systems
- Agent computing for interoperable healthcare systems
- Pervasive computing for interoperable healthcare systems
- Standards for interoperable healthcare systems
- Methods for healthcare systems design
- Semantic technologies for healthcare systems
- Privacy and security in healthcare systems
- Mobile healthcare systems
- Context management for healthcare systems

- Protection of individual privacy for aggregate anonymous data
- Healthcare security and privacy policies
- Artificial intelligence technologies to support complex decision making in healthcare systems
- Case studies

## AUDIENCE

This workshop will be of particular interest to IT researchers who are working in the field of healthcare systems, those interested in developing open systems, in tracking and developing standards, and of general interest to anyone using IT for interoperable software development. We also believe that the Workshop's topic area will be of significant interest to the wider IT community and expect industry participation.

## FORMAT

The format of the workshop in terms of number of sessions, types of papers (long or short), keynote speakers, and last but not least panel discussions will be set upon completing paper review and author notification. The workshop format will be designed to foster discussion and developing action outcomes on key issues relating to developing interoperable healthcare systems.

## PROGRAM COMMITTEE

**Sohaib Majzoub**, AUD, U. A. E  
**Bassillio Dahlan**, AUD, U. A. E  
**Rabeb Mizouni**, Khalifa University, U. A. E.  
**Paul D. Yoo**, KUSTAR, U. A. E.  
**Hicham Elzabadani**, AUD, U. A. E  
**Lay-Ki Soon**, Soongsil University, Republic of Korea,  
**Elie Abi-Lahoud**, Bourgogne University, France  
**Ji Ruan**, Uni Sydney, Australia,  
**Nilmini Wickramasinghe**, RMIT, Australia  
**Mohy Mohyuddin**, King Abdullah International Medical Research Center, Saudi Arabia

## ORGANIZING COMMITTEE

**Zakaria Maamar**, Zayed University, Dubai, U. A. E., zakaria.maamar@zu.ac.ae, maamarz@gmail.com  
**Christian Guttmann**, Etisalat British Telecom Innovation Centre, Khalifa University, Abu Dhabi Campus, Abu Dhabi, U. A. E., christian.guttmann@kustar.ac.ae  
**Osama Elhassan**, TECOM Investment, Dubai, U. A. E.  
**Wathiq Mansoor**, American University in Dubai, Dubai, U. A. E., wmansoor@aud.edu  
**Fahim Akhter**, Zayed University, Dubai, U. A. E., fahim.akhter@zu.ac.ae

<sup>1</sup> [http://www.who.int/whosis/whostat/EN\\_WHS09\\_Table7.pdf](http://www.who.int/whosis/whostat/EN_WHS09_Table7.pdf)





# The Intersection of Clinical Decision Support and Electronic Health Record: A Literature Review

Hajar Kashfi

Department of Applied Information Technology, Chalmers University of Technology  
SE-412 96 Gothenburg, Sweden, Email: hajar.kashfi@chalmers.se

**Abstract**—It is observed that clinical decision support (CDS) and electronic health records (EHR) should be integrated so that their contribution to improving the quality of health care is enhanced. In this paper, we present results from a review on the related literature. The aim of this review was to find out to what extent CDS developers have actually considered EHR integration in developing CDS. We have also investigated how various clinical standards are taken into account by CDS developers.

We observed that there are few CDS development projects where EHR integration is taken into account. Also, the number of studies where various clinical standards are taken into consideration in developing CDS is surprisingly low especially for *openEHR*, the EHR standard we aimed for. The reasons for low adoption of *openEHR* are issues such as complex and huge specifications, shortcomings in educational aspects, low empirical focus and low support for developers. It is concluded that there is a need for further investigation to discover the reasons why the rate of integration of EHRs and CDS is not at an optimum level and mostly to discover why CDS developers are not keen to adopt clinical standards.

## I. INTRODUCTION

**E**VEN though more than 50 years of research have been put into the clinical decision support (CDS) field, the adoption rate of these systems is still low [1], [2], [3], [4], [5], [6]. Various researchers have investigated the factors that should be considered by developers of such systems in order to result in higher adoption. One of these factors is the integration of CDS into the electronic health record (EHR) systems. Different benefits are associated with the integration of CDS into EHRs. For instance, integration facilitates real time access to the knowledge provided by CDS at point of care, it also eliminates tedious duplicate patient data entry since the pre-existing digital patient data in the EHR system can be utilized for the purpose of providing decision support [1], [7], [8].

The aim of this study is to answer this research question: *is integration of clinical decision support into electronic health record taken into consideration by developers of clinical decision support?* The related literature was reviewed not only to explore CDS developers' attitude towards integration of EHR and CDS, but also to discover the status of EHR standards in this field.

The structure of the paper is as follows. We start with the background information including the motivation for integration of CDS and EHRs in Section II. In Section III the literature review search strategy is given. The results of the review are presented in Section IV. Section V includes the discussion of the findings along with our reflection on the low adoption rate of the *openEHR* EHR standardization approach. Finally, we end with a conclusion and future directions of the study in Section VI.

## II. BACKGROUND

The idea of computerized medical records has been around as one of the key research areas in medical informatics for more than 20 years. Iakovidis defines EHR as “digitally stored health care information about an individual’s lifetime with the purpose of supporting continuity of care, education and research, and ensuring confidentiality at all times” [9]. EHRs include the whole range of patient-related data such as demographic information, medical history, medication, and allergies [10].

The main aim of EHRs is to make distributed and cooperating health information system and health networks a reality [10].

Several reasons have been identified for the low adoption rate of EHRs in small hospitals and office practices. This includes high implementation and maintenance costs, additional time and effort and finally the difficulty in choosing among available systems on the market due to a lack of standardization [1].

Improving the quality of health care is the ultimate goal of the EHR research domain, but it is in doubt whether EHRs have the ability to fulfill this goal [5]. EHRs need to be supported by other services in order to improve the quality of care [5], [11], [12], [13]. To reach the goal of improved health care quality, it is central to have CDS [5], [14], [3], [2], [6], [12], [15].

It has been observed that if there is no decision support service, the clinical knowledge needed for making a decision is not always available or applied [16]. Therefore, it is recommended that clinicians be automatically supported by

timely access to clinical decision support tools [7], [8]. The emphasis in the current application of EHRs is on timely access to patient data, patient tracking and providing decision support with the aim of improving quality of care [13]. In spite of this fact, the usage of decision support among EHR users is still quite low and there are still many EHR systems that do not include any CDS features [5]. Nonetheless, interest in applying CDS in various health care organizations to improve quality of health care has recently shown an increase [17], [18]. The CDS these organizations are looking for should provide support in patient specific assessments [17], [1].

#### A. Low Adoption of Clinical Decision Support

Results from several studies that deal with the question: *which factors should be considered in the design and development of CDS to result in an acceptable and effective CDS?* are summarized in [19]. These studies focus on developing such systems that lead to wider adoption of CDS and consequent improvement in quality of health care. According to these studies and those reviewed in this section, three of the main challenges in design and development of CDS are:

- human-related factors that are related to the way CDS systems are designed, evaluated and introduced to the users
- technical factors that are mainly related to knowledge representation and reasoning in CDS systems
- Integration to the EHR systems available in health organizations

#### B. Integration of Clinical Decision Support into Electronic Health Records

It is recommended that CDS be integrated into other information systems in the clinical domain and it has been demonstrated that an integrated system has better effects on the care process [20]. Different clinical applications such as computerized physician order entry (CPOE), electronic prescribing, e-prescribing (eRX) and personal health records (PHR) are valuable underlying platforms for CDS [16], [1]. Several studies discuss how delivery of decision support through EHRs can improve the quality of care [4], [3], [21], [22]. Moreover, integration of CDS into EHR systems has been advocated in several studies as being helpful to the wider adoption of CDS [2], [1], [5], [4], [23], [16], [24]. Overall, EHR is considered as leverage for CDS [6], [1].

Several studies have observed that manual data entry into CDS acts as a barrier for broad adoption of CDS. It is recommended that the CDS be provided at the point of care and without any additional effort to invoke it or utilize it [1], [17]. One sample scenario for an integrated CDS feature would be prompts or alerts that appear on the screen in order to inform the clinician about a drug-drug or drug-allergy interaction for one specific patient while reviewing/editing the patient's health record.

Manual data entry which is a time consuming task and a burden for clinicians can be removed by integrating CDS into EHR systems and utilizing the data which is already in an

electronic, computer-readable format. In this case, there is no need for duplicate data entry and the system can query related information from the EHR system [2], [1], [23], [25], [6]. Therefore, implementation of CDS is facilitated by EHRs. If there is no integration, data must be extracted from EHRs to be applied in the CDS. Moreover, if CDS is not integrated into EHRs, that part of the domain knowledge which is included in EHR is not applied properly [1].

#### C. Interoperability of EHR systems

EHR systems are being developed by various vendors, so they might be stored in different formats. This results in systems that are not interoperable, and makes sharing EHRs among different health organizations difficult. To overcome this problem, and to support secure and timely access to EHRs, national and international EHR standards are developed [26], [27]. *openEHR* [28] and *health level 7* (HL7) [29] are two of the well-known interoperability standards. A description of these two standards follows:

1) *openEHR*: *openEHR* is an open standard specification. The *openEHR* specification describes how health data, i.e. EHRs, are managed, stored, retrieved and exchanged [30]. Three main concepts defined in *openEHR* are (i) the two-level software architecture (ii) archetypes (iii) templates. The two-level architecture for clinical applications deals with separation of knowledge and information levels in order to overcome the problems caused by the ever-changing nature of clinical knowledge. This is realized by using *openEHR* archetypes. Archetypes and templates are used for data validation and sharing [28]. Beale et al. in [31] define archetype and template as follows:

- Archetype is “a computable expression of a domain content model in the form of structured constraint statements, based on a reference (information) model. *openEHR* archetypes are based on the *openEHR* reference model. Archetypes are all expressed in the same formalism. In general, they are defined for wide re-use, however, they can be specialized to include local particularities. They can accommodate any number of natural languages and terminologies.”
- Template is “a directly locally usable definition which composes archetypes into a larger structures often corresponding to a screen form, document, report or message. A template may add further local constraints on the archetypes it mentions, including removing or mandating optional sections, and may define default values.”

2) *Health Level 7*: HL7 is an EHR standard that focuses on communicating health data, i.e. EHRs, [10]. According to HL7 website<sup>1</sup>: “Health Level Seven International (HL7) is a not-for-profit, ANSI-accredited standards developing organization dedicated to providing a comprehensive framework and related standards for the exchange, integration, sharing, and retrieval of electronic health information that supports

<sup>1</sup><http://www.hl7.org/>

clinical practice and the management, delivery and evaluation of health services". In HL7 version 3 a comprehensive Reference Information Model (RIM) is introduced [10]. HL7 clinical document architecture (CDA) templates are analogous to *openEHR* archetypes [32].

3) *Other Standards in the Clinical Domain*: There are different approaches to support the interoperability among heterogeneous clinical systems. Other than EHR interoperability standards that concentrate on standardizing the clinical information model, the initiative has been taken to standardize other concepts in the clinical domain such as clinical guidelines and clinical terminologies to improve shareability and reusability of them among health institutions.

- **Communicating the Clinical Terminology** The language is not uniform in the clinical domain and clinicians may use different terms to refer to the same concepts. Therefore, there is a need to standardize the clinical terminology to enable communicating it [33]. SNOMED CT (Systematized Nomenclature of Medicine – Clinical Terms) is an advanced clinical terminology and coding system [33]. SNOMED CT concepts are usually referred to by an information model such as *openEHR* and HL7 [34].

ICD (International Classification of Diseases) is a coding system that is designed to "promote international comparability in the collection, processing, classification, and presentation of mortality statistics" [35]. This classification standard is suitable for statistical reporting rather than clinical documentation as is supported by SNOMED CT. There is a map between SNOMED CT terms and the equivalent ICD codes [34].

- **Sharing Clinical Guidelines** Developing clinical guidelines involves a lot of effort. Therefore, there have been initiatives to enable reusability and shareability of clinical guidelines among various health organizations. The first step to support reusability and shareability of clinical guidelines is to define a common format for representing them [36]. One well-known language for this purpose is the one developed by the InterMed Collaboratory named GLIF (the GuideLine Interchange Format) [36].

### III. METHODS AND MATERIALS

The search was conducted in the Scenedirect<sup>2</sup> database that includes the major journals in medical informatics. The search strategy is depicted in Figure 1 and explained in more details in the following.

- searching the combination of phrases "clinical decision support" and "electronic health record" returned 48 articles where 37 of them were selected for further studies.
- searching the combination of phrases "clinical decision support" and "medical health record" (excluding the papers that had the phrase "electronic health record") returned 50 articles where 37 of them were selected for further studies.

<sup>2</sup><http://sciencedirect.com>

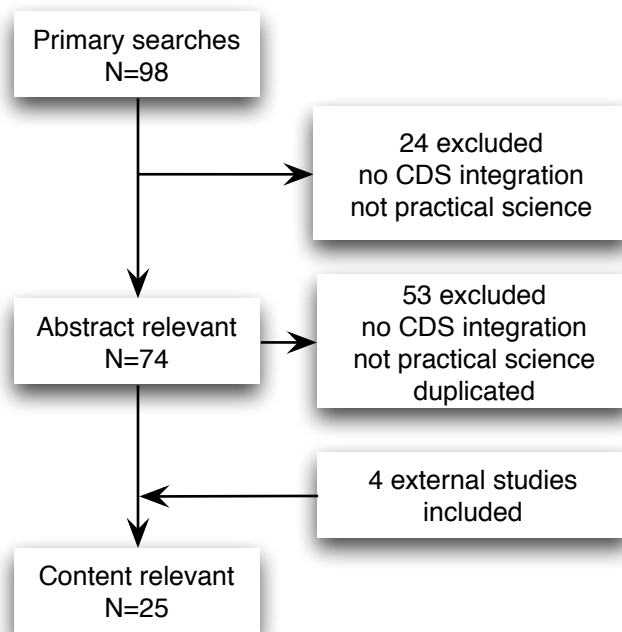


Fig. 1. Search process.

Of these 74 studies, only 21 turned out to be relevant to the review. In addition to these 21 studies, 4 more studies that the author had found were included in the review.

Inclusion criteria for the papers were positive answers to these questions based on their titles and abstracts:

- Is the study discussing development and/or evaluation of an EHR or a CDS system (i.e. practical science)?
- If Have the authors considered integration of CDS into EHRs or a related application (i.e. integration)?

Since, we were particularly interested in *openEHR*, further searches were carried out in ScienceDirect and PubMed<sup>3</sup> specifically on *openEHR* to find out if any development of an *openEHR*-based CDSS is documented in the literature:

- In ScienceDirect, searching the combination of phrases "clinical decision support" and *openEHR* resulted in 1 article that was reviewed before (the study by Greenes [1]).
- In PubMed, searching the combination of phrases "clinical decision support" and *openEHR* resulted in 2 articles by the author of this paper [37], [38] (these papers are not included in the review).

### IV. RESULTS

This section includes the preliminary findings from the literature review. Analysis of the findings are given in the next section.

The 25 selected articles were reviewed in order to find out whether they consider any of the clinical standards (i.e. EHR standards, guideline representation standards, and terminology

<sup>3</sup><http://pubmed.org>

TABLE I  
THE SUMMARY OF THE FINDINGS.

Who	Year	Integr- ation	Standards		
			EHR	Guideline	Terminology
Stair [39]	1998	✓	✗	✗	✗
Gadd et al. [40]	1998	✓	✗	✗	✗
Panzarasa et al. [41]	2002	✓	✗	✗	✓
Young et al. [42]	2004	✓	✗	✗	✗
Shiffman et al. [43]	2004	✓	HL7	✓	✗
Rosenbloom et al. [44]	2004	✓	✗	✗	✗
Galanter et al. [45]	2005	✓	✗	✗	✗
Haller et al. [46]	2007	✓	✗	✗	✗
Stutman et al. [47]	2007	✓	✗	✗	✗
Wilson et al. [24]	2007	✓	✗	✗	✗
Lobach et al. [48]	2007	-	HL7	✗	✗
Graham et al. [49]	2008	-	HL7	✗	✗
Marcy et al. [50]	2008	✓	✗	✗	✗
Wright et al. [51]	2008	✓	HL7	✗	SNOMED CT,ICD
Gerard et al. [52]	2008	✓	✗	✗	✗
Field et al. [53]	2008	✓	✗	✗	✗
Schnipper et al. [54]	2008	✓	✗	✗	✗
Peleg et al. [55]	2009	✓	✗	GLIF	✗
Saleem et al. [56]	2009	✓	✗	✗	✗
Field et al. [57]	2009	✓	✗	✗	✗
Chen et al. [58]	2010	✓	✗	✓	✗
Galanter et al. [59]	2010	✓	✗	✗	SNOMED CT,ICD
Noormohammad et al. [60]	2010	✓	HL7	✗	✓
Trafton et al. [61]	2010	✓	✗	✗	✗
Were et al. [62]	2010	✓	HL7	✗	✗

or vocabulary standards). The summary of the results is shown in Table I. The Integration column indicates if the integration of EHRs and CDS is taken into consideration in the study (✓) or not (✗), there are cases where the authors did not reveal any information in this regard (-). If any sorts of standards is applied in the study, the corresponding column is marked with ✓, and in cases where an international standard is used with the name of the standard e.g. HL7 for EHR, SNOMED CT for terminology.

As evident from Table I, there are various studies that have applied EHR standards (not including *openEHR*) in developing EHRs with CDS functionalities. HL7 is used in 7 studies, GLIF in 1, and SNOMED CT/ICT in 2 studies. There are also studies in which local representations or terminologies were used for representing clinical guidelines or clinical terms [60], [41]. Most of the CDS services were documented to be integrated into a CPOE system. The summary of findings is presented in Figure 2.

#### A. HL7 versus *openEHR*

While searching the combination of phrases “clinical decision support” and HL7 resulted in 41 papers<sup>4</sup>, we did not find any study that reports on implementation of a CDS applying *openEHR*<sup>5</sup>.

## V. DISCUSSION

Theory supports the benefits offered by integrating CDS into EHR, but this concept is still appreciated more in theory

<sup>4</sup>Not all of these studies are included in the review.

<sup>5</sup>The search was done in mid 2010. However, in a new search in 2011, we found more new studies related to *openEHR*. These studies are discussed more in the discussion section.

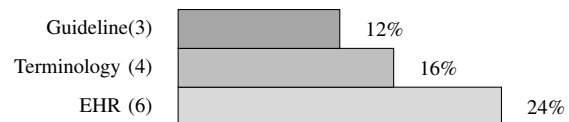


Fig. 2. Various standards reported in the reviewed studies. All of the EHR standards that were applied in studies were HL7. *openEHR* was not adopted in any of the studies.

than in practice. Only 25 related studies were discovered in this database while around 100 studies are documented in the literature that, based on their titles and abstracts, are about developing a CDSS. Nonetheless, the publication years of these 25 studies are an indication that in recent years, there has been an increase in consideration of EHR integration in development of CDS.

Moreover, it is observed that taking standards into consideration in any clinical application (and generally any information system) is very important [11]. In case of CDSSs, since such systems operate by utilizing both patient/organizational-specific data and clinical knowledge, it is important to consider the standards that support each of these areas [11]. This however is observed to still be in need of further improvements. Of these 25 studies, only 6 had considered EHR standardization, and 3 had considered terminology standards which are both surprisingly small numbers.

Finally, one can conclude that based on the literature, HL7 has a higher level of adoption than *openEHR* and that applying *openEHR* in development of clinical applications specially CDS is yet rare. This brings the question that regardless of the advances in theory why *openEHR* is suffering from a low adoption rate in practice. This issue is discussed more in the following section.

#### A. Low Adoption of *openEHR*

Below is a list of problems that we or others have faced using *openEHR*<sup>6</sup>. These issues are considered as barriers to higher adoption of *openEHR*<sup>7</sup>:

1) *Being huge and complex\**: *openEHR* is naturally complex, and this complexity is not unexpected since *openEHR* is considered to be a solution for a complicated problem (i.e. interoperable future-proof EHR) in a complex domain (i.e. the clinical domain). For instance the powerful archetype model allows expressing complex clinical concepts, therefore, an inexperienced archetype developer should expect to spend some time on learning the *openEHR* concepts. Additionally, getting a grip on the current specifications (more than 1000 pages), UML diagrams and code documentation is challenging. At the same time, it is notable that this complexity is intensified by

<sup>6</sup>In November 2010, there was a discussion on *openEHR* mailing list with the same topic. This shows that even people in the *openEHR* community have noticed that the adoption rate is low and some actions should be taken in order to improve it. Especially, it is noticeable that the amount of attention to *openEHR* is much less than HL7 in various domains i.e. government, academy and industry.

<sup>7</sup>Some of the issues presented here are the result of investigating the discussions in the *openEHR* mailing list, even though some others had been experienced in this study. Those issues are marked with an asterisk.

some other aspects such as improper educational support and limited internationalization.

2) *Shortcoming in educational aspects\**: Understanding a concept is the first step to be able to adopt it and this is even more valid for such complex concepts like those in *openEHR*. Unfortunately, no formal tutorial document exists for *openEHR*, formal training sessions are rare and even worse, not so many *openEHR* trainers exist around the world. Easy to understand tutorials are needed to help novice developers get a grip on *openEHR*.

3) *Low government and industry penetration*: Many of those who are interested in *openEHR*, in spending time on learning it or adopting it, are from the academic world (the main of which is University College London<sup>8</sup>). So far, there are very few companies that are adopting *openEHR* and to our knowledge these companies are considered to be a part of the core *openEHR* community. The main companies are Oceaninformatics<sup>9</sup>, Cambio<sup>10</sup>, and Zilics<sup>11</sup>. But what about “ordinary audience”? On the other hand, low support from the governmental agencies lead to low industry penetration. Considering the complication and the cost imposed by the *openEHR* approach, and also limited documentations and guidelines, applying *openEHR* is not still cost-efficient and yet commercial companies show a lot of hesitation to accept risks imposed by adopting this immature standard.

4) *Shortcoming in internationalization aspects*: In order to reach an international-wide adoption, it is suggested that establishing regional communities would be helpful; nevertheless, there are other concerns in this regard. *openEHR* community should consider issues such as supporting and providing guidelines for regional communities all around the world. It is also beneficial to publish *openEHR* specifications in various languages in order to speed up the process of learning for various people. Regional events such as educational sessions, gatherings and so on are also valuable to influence collaboration. As an example, in Sweden, there are around 4 groups of people<sup>12</sup> doing research on or adopting *openEHR*, but collaboration among them is at a very low level.

5) *Low empirical focus\**: *openEHR* should not just be about complex theoretical specifications and reference models, but also about implementation and practice. Semantic interoperability, two-level modeling and involving clinicians are interesting concepts, but so far these have been far from the practice. Currently, there are just a few empirical efforts on *openEHR*. Most of the focus of *openEHR* community has been on representation of domain concepts and theoretical aspects of the approach. Still, there is a huge need for supporting developers to make *openEHR* more practical.

6) *Limited tools and implementations\**: As mentioned above, developers needed to be supported in order to improve

adoption of *openEHR*. One way of delivering this support is by providing frameworks and application programming interfaces (API). At this time, the *openEHR* reference model implementation is still immature and lacks important parts like templates, persistence, and services.

#### B. Recent Advances in The *openEHR*-based CDS

When it comes to CDS, there are a few studies that deal with how *openEHR* offers opportunities for CDS. Most of these efforts however, seem to be more focused on integrating clinical guidelines into *openEHR* archetypes or utilizing archetypes for representing clinical guidelines [63], [25], [64] or to integrate reasoning and clinical archetypes (enhance archetypes by including knowledge representation capabilities to them) [65]. To our knowledge there is almost no study that has been focused on benefiting from the well-structured *openEHR*-based patient data for adopting data intensive reasoning methods in CDSSs or methods that rely on previous cases to carry out the reasoning process.

#### C. Why Are Clinical Standards Important for CDS?

According to the discussion in Section II, enough motivation exists to integrate CDS into EHRs. There is still a question whether integration of CDS into EHRs can be done without taking EHR related standards into account. If EHR standards are not considered in CDS development, all clinical data should be translated to a format understandable for the CDS system. This is not an efficient way for CDS and the EHR system to communicate. Moreover, there is an increasing interest in the medical informatics community to share clinical knowledge. This can also be supported if CDS is developed based on EHR standards. For instance, by enriching standard compatible EHRs with the reasoning knowledge, EHR sharing will also result in sharing and reusing the embedded knowledge. In cases where general domain knowledge including clinical guidelines are integrated into EHRs, the reusability and sharing of knowledge can be achieved as well.

## VI. CONCLUSION

Researchers in the area of CDS and also EHR have argued that by integrating CDS into EHRs, the improvement in the quality of health care would be higher than when the systems operate separately. The integration will be more efficient if the standards related to EHRs are considered in developing CDS. The possibility to share the domain knowledge, especially the reasoning knowledge, in decision making is another motivation for taking standards into account in developing CDS.

Nevertheless, a review of the related literature indicates that not all of CDS developers take integration into account, also there are very few of them who consider standards in developing CDS. Discovering the reasons for this however needs further investigation and has not been in the scope of this review.

## ACKNOWLEDGMENT

I would like to give thanks to Olof Torgersson who provided helpful suggestions to improve the paper.

<sup>8</sup><http://ucl.ac.uk>

<sup>9</sup><http://www.oceaninformatics.com>

<sup>10</sup><http://cambio.se>

<sup>11</sup><http://www.zilics.com.br>

<sup>12</sup>Chalmers university of technology, Linköping university, Cambio company and The Swedish NHS.

## REFERENCES

- [1] R. Greenes, *Clinical decision support: the road ahead*. Academic Press, 2007.
- [2] T. Wendt, P. Knaup-Gregori, and A. "Decision Support in Medicine: A Survey of Problems of User Acceptance," *Stud Health Technol Inform*, vol. 77, pp. 852–856, 2000.
- [3] B. Chaudhry, J. Wang, S. Wu, and M. Maglione, "Systematic review: impact of health information technology on quality, efficiency, and costs of medical care," *Annals of Internal Med*, vol. 144, no. 10, pp. 742–752, 2006.
- [4] A. Garg, N. Adhikari, H. McDonald, and M. Rosas, "Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review," *Journal of the American Medical Association*, vol. 293, no. 10, pp. 1223–1238, 2005.
- [5] D. F. Sittig, A. Wright, J. a. Osheroff, B. Middleton, J. M. Teich, J. S. Ash, E. Campbell, and D. W. Bates, "Grand challenges in clinical decision support." *Journal of Biomedical Informatics*, vol. 41, no. 2, pp. 387–92, Apr. 2008.
- [6] J. Osheroff, J. Teich, B. Middleton, E. Steen, A. Wright, and D. Detmer, "A roadmap for national action on clinical decision support," *Journal of the American Medical Informatics Association*, vol. 14, no. 2, p. 141, 2007.
- [7] "Patient Safety: Achieving a New Standard of Care," Washington DC, 2003.
- [8] "Crossing the Quality Chasm: A New Health System for the 21st Century," Website, Washington DC, 2001, [http://journals.lww.com/qmhcjournal/Citation/2002/10040/Crossing\\_the\\_Quality\\_Chasm\\_A\\_New\\_Health\\_System.10.aspx](http://journals.lww.com/qmhcjournal/Citation/2002/10040/Crossing_the_Quality_Chasm_A_New_Health_System.10.aspx).
- [9] I. Iakovidis, "Towards personal health record: current situation, obstacles and trends in implementation of electronic healthcare record in Europe." *International Journal of Medical Informatics*, vol. 52, no. 1-3, pp. 105–15, 1998.
- [10] B. Blobel, "Advanced and secure architectural EHR approaches." *International Journal of Medical Informatics*, vol. 75, no. 3-4, pp. 185–90, 2006.
- [11] J. Osheroff, E. Pifer, J. Teich, D. Sittig, and R. Jenders, *Improving outcomes with clinical decision support: An implementer's guide*. HIMSS, 2005.
- [12] R. Greenes, M. Sordo, D. Zaccagnini, M. Meyer, and GJ, "Design of a standards-based external rules engine for decision support in a variety of application contexts: report of a feasibility study at Partners HealthCare System," *Medinfo*, 2004.
- [13] L. Zhou, C. S. Soran, C. a. Jenter, L. a. Volk, E. J. Orav, D. W. Bates, and S. R. Simon, "The relationship between electronic health record use and quality of care over time." *Journal of the American Medical Informatics Association*, vol. 16, no. 4, pp. 457–64, 2009.
- [14] J. Anderson, "Increasing the acceptance of clinical Information," *MD computing: Computers in Medical Practice*, vol. 16, no. 1, p. 62, 1999.
- [15] K. Kawamoto and D. Lobach, "Proposal for fulfilling strategic objectives of the US roadmap for national action on decision support through a service-oriented architecture leveraging HL7 services." *Journal of the American Medical Informatics Association*, pp. 146–155, 2007.
- [16] I. Cho, J. Kim, J. H. Kim, H. Y. Kim, and Y. Kim, "Design and implementation of a standards-based interoperable clinical decision support architecture in the context of the Korean EHR." *International Journal of Medical Informatics*, vol. 9, pp. 611–622, Jul. 2010.
- [17] K. Kawamoto, C. A. Houlihan, E. A. Balas, and D. F. Lobach, "Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success." *BMJ (Clinical research ed.)*, vol. 330, no. 7494, p. 765, 2005.
- [18] M. Trivedi, J. Kern, A. Marcee, B. Grannemann, B. Kleiber, T. Bettinger, K. Altshuler, and A. McClelland, "Development and Implementation of Computerized Clinical Guidelines : Barriers and Solutions," *Methods of Information in Medicine*, vol. 41, no. 5, pp. 435–442, 2002.
- [19] H. Kashfi, "Towards Interaction Design in Clinical Decision Support Development: A Literature Review," *International Journal of Medical Informatics*, 2011, in review article.
- [20] R. A. K. Horasani, M. I. T. Anasijevic, B. L. M. Iddleton, and M. S. C, "Ten Commandments for Effective Clinical Decision Support: Making the Practice of Evidence-based Medicine a Reality," *Journal of the American Medical Informatics Association*, vol. 10, pp. 523–530, 2003.
- [21] D. Hunt, R. Haynes, S. Hanna, and K. Smith, "Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review," *Journal of the American Medical Association*, vol. 280, no. 15, p. 1339, Oct. 1998.
- [22] M. Johnston, K. Langton, and R. Haynes, "Effects of computer-based clinical decision support systems on clinician performance and patient outcome. A critical appraisal of reserach," *Ann Intern Med*, vol. 120, no. 2, pp. 135–142, 1994.
- [23] E. Berner, *Clinical Decision Support Systems: Theory and Practice (Health Informatics)*. New York, NY 10013, USA: Springer, 2007.
- [24] A. Wilson, A. Duszynski, D. Turnbull, and J, "Investigating patients' and general practitioners' views of computerised decision support software for the assessment and management of cardiovascular risk," *Informatics in Primary Care*, vol. 15, pp. 33–44, 2007.
- [25] R. Chen, P. Georgii-Hemming, and H. Ahlfeldt, "Representing a chemotherapy guideline using openEHR and rules." *Studies in Health Technology and Informatics*, vol. 150, pp. 653–7, Jan. 2009.
- [26] V. Stroetmann, D. Kalra, P. Lewalle, J. Rodrigues, and KA, "Semantic Interoperability for Better Health and Safer Health Care," *Deployment and Research*, no. January, 2009.
- [27] P. Schloeffel, T. Beale, G. Hayworth, S. Heard, and H. Leslie, "The relationship between CEN 13606, HL7, and openEHR," in *In Health Informatics Conference*, vol. 7. Health Informatics Society of Australia, 2006, p. 24.
- [28] T. Beale and S. Heard, "openehr architecture overview," Website, 2008, <http://www.openehr.org/releases/1.0.2/architecture/overview.pdf>.
- [29] "HL7," Website, 2010, <http://hl7.org>.
- [30] "openEHR," Website, 2010, <http://en.wikipedia.org/wiki/openehr>.
- [31] T. Beale and S. Heard, "Archetype Definitions and Principles," Website, 2007, [http://www.openehr.org/releases/1.0.2/architecture/am/archetype\\_principles.pdf](http://www.openehr.org/releases/1.0.2/architecture/am/archetype_principles.pdf).
- [32] M. Eichelberg, T. Aden, J. Riesmeier, A. Dogac, and G. B. Laleci, "A survey and analysis of Electronic Healthcare Record standards," *ACM Computing Surveys*, vol. 37, no. 4, pp. 277–315, Dec. 2005.
- [33] K. Donnelly, "SNOMED-CT: The advanced terminology and coding system for eHealth." *Studies in Health Technology and Informatics*, vol. 121, pp. 279–90, Jan. 2006.
- [34] "SNOMED CT," Website, 2010, [http://en.wikipedia.org/wiki/SNOMED\\_CT](http://en.wikipedia.org/wiki/SNOMED_CT).
- [35] "ICD," Website, 2010, <http://www.cdc.gov/nchs/icd.htm>.
- [36] L. Ohno-Machado, J. H. Gennari, S. N. Murphy, N. L. Jain, S. W. Tu, D. E. Oliver, E. Pattison-Gordon, R. a. Greenes, E. H. Shortliffe, and G. O. Barnett, "The guideline interchange format: a model for representing guidelines." *Journal of the American Medical Informatics Association*, vol. 5, no. 4, pp. 357–72, 1998.
- [37] H. Kashfi, "An openEHR-based clinical decision support system: a case study." in *Studies in health technology and informatics*, vol. 150, Jan. 2009, p. 348.
- [38] —, "Applying a user centered design methodology in a clinical context," in *Studies in health technology and informatics*, vol. 160, no. Pt 2, Jan. 2010, pp. 927–31.
- [39] T. Stair, "Reduction of Redundant Laboratory Orders by Access to Computerized Patient Records," *The Journal of Emergency Medicine*, vol. 16, no. 6, pp. 895– 897, 1998.
- [40] C. Gadd, P. Baskaran, and D. Lobach, "Identification of design features to enhance utilization and acceptance of systems for Internet-based decision support at the point of care." in *Proceedings of the AMIA*, Jan. 1998, pp. 91–5.
- [41] S. Panzarasa, S. Maddč, and S. Quaglini, "Evidence-based careflow management systems: the case of post-stroke rehabilitation," *Journal of Biomedical*, vol. 35, no. 2, pp. 123–139, Apr. 2002.
- [42] A. S. Young, J. Mintz, A. N. Cohen, and M. J. Chinman, "A network-based system to improve care for schizophrenia: the Medical Informatics Network Tool (MINT)." *Journal of the American Medical Informatics Association*, vol. 11, no. 5, pp. 358–67, 2004.
- [43] R. N. Shiffman, G. Michel, A. Essaihi, and E. Thorquist, "Bridging the guideline implementation gap: a systematic, document-centered approach to guideline implementation." *Journal of the American Medical Informatics Association*, vol. 11, no. 5, pp. 418–26, 2004.
- [44] S. T. Rosenbloom, D. Talbert, and D. Aronsky, "Clinicians' perceptions of clinical decision support integrated into computerized provider order entry." *International Journal of Medical Informatics*, vol. 73, no. 5, pp. 433–41, Jun. 2004.

- [45] W. L. Galanter, R. J. Didomenico, and A. Polikaitis, "A trial of automated decision support alerts for contraindicated medications using computerized physician order entry." *Journal of the American Medical Informatics Association*, vol. 12, no. 3, pp. 269–74, 2005.
- [46] G. Haller, P. S. Myles, J. Stoelwinder, M. Langley, H. Anderson, and J. McNeil, "Integrating incident reporting into an electronic patient record system." *Journal of the American Medical Informatics Association*, vol. 14, no. 2, pp. 175–81, 2007.
- [47] H. Stutman, R. Fineman, and K. Meyer, "Optimizing the acceptance of medication-based alerts by physicians during CPOE implementation in a community hospital environment," in *AMIA Annual Symposium*, 2007, pp. 701–705.
- [48] D. F. Lobach, K. Kawamoto, K. J. Anstrom, M. L. Russell, P. Woods, and D. Smith, "Development, deployment and usability of a point-of-care decision support system for chronic disease management using the recently-approved HL7 decision support service standard." *Studies in Health Technology and Informatics*, vol. 129, no. Pt 2, pp. 861–5, Jan. 2007.
- [49] T. Graham, A. Kushniruk, M. Bullard, B. Holroyd, D. Meurer, and B. Rowe, "How usability of a web-based clinical decision support system has the potential to contribute to adverse medical events," in *AMIA Annual Symposium Proceedings*, vol. 2008. American Medical Informatics Association, Jan. 2008, p. 257.
- [50] T. W. Marcy, B. Kaplan, S. W. Connolly, G. Michel, R. N. Shiffman, and B. S. Flynn, "Developing a decision support system for tobacco use counselling using primary care physicians." *Informatics in Primary Care*, vol. 16, no. 2, pp. 101–9, Jan. 2008.
- [51] A. Wright and D. F. Sittig, "SANDS: a service-oriented architecture for clinical decision support in a National Health Information Network." *Journal of Biomedical Informatics*, vol. 41, no. 6, pp. 962–81, 2008.
- [52] M. N. Gerard, W. E. Trick, K. Das, M. Charles-Damte, G. A. Murphy, and I. M. Benson, "Use of clinical decision support to increase influenza vaccination: multi-year evolution of the system." *Journal of the American Medical Informatics Association*, vol. 15, no. 6, pp. 776–9, 2008.
- [53] T. S. Field, P. Rochon, M. Lee, L. Gavendo, S. Subramanian, S. Hoover, J. Baril, and J. Gurwitz, "Costs associated with developing and implementing a computerized clinical decision support system for medication dosing for patients with renal insufficiency in the long-term care setting." *Journal of the American Medical Informatics Association*, vol. 15, no. 4, pp. 466–72, 2008.
- [54] J. L. Schnipper, J. A. Linder, M. B. Palchuk, J. S. Einbinder, Q. Li, A. Postilnik, and B. Middleton, "'Smart Forms' in an Electronic Medical Record: documentation-based clinical decision support to improve disease management." *Journal of the American Medical Informatics Association*, vol. 15, no. 4, pp. 513–23, 2008.
- [55] M. Peleg, A. Shachak, D. Wang, and E. Karnieli, "Using multi-perspective methodologies to study users' interactions with the prototype front end of a guideline-based decision support system for diabetic foot care." *International Journal of Medical Informatics*, vol. 78, no. 7, pp. 482–93, Jul. 2009.
- [56] J. Saleem, L. Militello, N. Arbuckle, and M. Flanagan, "Provider Perceptions of Colorectal Cancer Screening Clinical Decision Support at Three Benchmark Institutions," in *AIMIA Symposium Proceedings*, 2009, pp. 558–562.
- [57] T. S. Field, P. Rochon, M. Lee, L. Gavendo, J. L. Baril, and J. H. Gurwitz, "Computerized clinical decision support during medication ordering for long-term care residents with renal insufficiency." *Journal of the American Medical Informatics Association*, vol. 16, no. 4, pp. 480–5, 2009.
- [58] C. C. Chen, K. Chen, C.-y. Hsu, and Y.-c. J. Li, "Developing guideline-based decision support systems using protégé and jess." *Computer Methods and Programs in Biomedicine*, vol. in print, pp. 1–7, Jun. 2010.
- [59] W. L. Galanter, D. B. Hier, C. Jao, and D. Sarne, "Computerized physician order entry of medications and clinical decision support can improve problem list documentation compliance." *International Journal of Medical Informatics*, vol. 79, no. 5, pp. 332–8, May 2010.
- [60] S. F. Noormohammad, B. W. Mamlin, P. G. Biondich, B. McKown, S. N. Kimaiyo, and M. C. Were, "Changing course to make clinical decision support work in an HIV clinic in Kenya." *International Journal of Medical Informatics*, vol. 79, no. 3, pp. 204–10, Mar. 2010.
- [61] J. Trafton, S. Martins, M. Michel, and E. Lewis, "Evaluation of the Acceptability and Usability of a Decision Support System to Encourage Safe and Effective Use of Opioid Therapy for Chronic, Noncancer Pain by Primary Care Providers." *Pain Medicine*, vol. 11, pp. 575–585, 2010.
- [62] M. C. Were, C. Shen, M. Bwana, N. Emenyonu, N. Musunguzi, F. Nkuyahaga, A. Kembabazi, and W. M. Tierney, "Creation and evaluation of EMR-based paper clinical summaries to support HIV-care in Uganda, Africa." *International Journal of Medical Informatics*, vol. 79, no. 2, pp. 90–6, Mar. 2010.
- [63] M. Marcos and B. n. Martínez-Salvador, "Towards the interoperability of computerized guidelines and electronic health records: an experiment with openEHR archetypes and a chronic heart failure guideline," in *Proceedings of the ECAI 2010 conference on Knowledge representation for health-care*, ser. KR4HC'10. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 101–113.
- [64] L. Xiao, G. Cousins, L. Hederman, T. Fahey, and B. Dimitrov, *The design of an EHR for clinical decision support*. IEEE, Oct. 2010, no. Bmei.
- [65] L. Lezcano, M.-A. Sicilia, and C. Rodríguez-Solano, "Integrating reasoning and clinical archetypes using OWL ontologies and SWRL rules." *Journal of Biomedical Informatics*, vol. 44, no. 2, pp. 343–53, Apr. 2011.





# EMeH: Extensible Mobile Platform for Healthcare

Tomasz Biały, Jacek Kobusiński, Maciej Małecki and Krzysztof Stefaniak

Poznań University of Technology

Poznań, Poland

Email: {tbialy,jkobusinski,mmalecki,kstefaniak}@cs.put.poznan.pl

**Abstract**—Rapid development of mobile technology and growing number of users open new possibilities in the context of using mobile devices in the healthcare. Despite that, the resources and computing power of these devices are still far less than desktop computers. Thus, one can take into account these limitations when designing applications for such devices. On the other hand, the increasing functionality of mobile devices makes them a real alternative for a traditional PC in certain areas. Nowadays, the mobility becomes one of the most required feature for the information system users. They require to be connected and have access to data all the time at any location. This is also true in the context of hospital environment where the medical staff must collect, update and retrieve various type of data. EMeH Platform is a solution that provides the possibility to create efficient distributed applications based on the SOA paradigm. It offers universal environment that allows to integrate PDAs, smartphones, tablets as well as specialized barcode scanners and RFID readers with the existing hospital information system. Layered and flexible design that reflect real usage scenarios, efficient and economic resource usage and consistent dynamically generated user interface makes EMeH Platform interesting extension to the traditional hospital information systems.

## I. INTRODUCTION

**H**EALTHCARE domain is one of the most dynamic and promising field of the IT appliance nowadays. As studies shows the IT infrastructure and software can increase the productivity and revenue of hospitals [1], [2], however, the medical personnel spend more time on data entry at the front of PC instead being with patients. The use of mobile devices in medical information systems creates the possibility of reducing this negative effect [3].

In the recent years the mobile devices became more universal and popular. The PDA, smartphone or tablets offer increasing functionality and power. There are also specialized mobile devices equipped with barcode scanners, RFID readers and GPS receivers. Currently, these devices, though still weaker than desktop PC, have enough power to become interesting alternative to them and can play the role of complex IT system terminals.

Implementing the entire functionality of a hospital information system (HIS) as mobile application is pointless due to the imposed limits of the devices, eg. entering of a long text is still inconvenient and time consuming. However, in some cases, the mobility becomes one of the most important feature. Barcode scanners and RFID readers can greatly accelerate the process of collecting data about medical equipment, drugs and patients. The tablets can be used by medical staff to present patient health record (Electronic Health Record, EHR) close to

his bed. Emergency situations are the next field where mobility becomes important and useful.

As mentioned previously, there are many possible mobile services that can enrich HIS. Typically they are realized as dedicated and separated applications, which makes the process of software development and management more difficult and expensive. The need to create many simple applications raises the idea of using service-oriented architecture (SOA) [4]. According to the SOA, applications will be replaced by web services used by the client's applications. To minimize the resource requirements imposed on mobile device, such a client application responsible for providing user interface should be implemented using thin-client paradigm.

In this paper we describe the EMeH Platform that allows to utilize various mobile devices in the healthcare environment. What is more important the EMeH Platform allows full integration with existing HIS system and follows newest trends in building distributed applications for network environment.

The paper is organized as follows. Section II briefly presents current trends on mobile device markets. The concept of the EMeH Platform is described in Section III. Detailed description of the client application is presented in Section IV, while Section V contains a description of the service environment. Related works are discussed in Section VI. Finally, concluding remarks are presented in Section VII.

## II. MOBILE PLATFORMS

In the recent years mobile devices market has dynamically developed. Therefore, a number of available mobile platforms and operating systems is considerable. Table I shows summary of the most popular mobile operating systems in the space of last two years.

Currently Google Android is the most popular operating system among all mobile platforms. It's market share has increased four-times last year, thereupon at the moment it has more users than the previous leader of ranking - Symbian operating system. Second mobile platform, which increased it's market share last year is Apple iOS. All other platforms have lost some of their popularity.

Regardless of the rapid development of the most recent platforms, there is a limited group of mobile devices, created precisely for specialized usage and amongst which vast majority are based on operating systems from Windows Mobile family. These devices, called mobile computers (e.g. handheld computers, wearable computers, vehicle-mounted mobile computers) can have various additional modules or peripherals

TABLE I  
TOP MOBILE PLATFORMS - GARTNER, MAY 2011 [5]

Mobile platform	Market Share 2010 / 2011 [%]	Trend
Android	9.6 / 36.0	+
Symbian	44.2 / 27.4	-
Apple iOS	15.3 / 16.8	+
RIM Blackberry	19.7 / 12.9	-
Windows Mobile/Phone 7	6.8 / 3.6	-
Other OS	4.4 / 3.3	-

such as barcode scanners, RFID readers or can be adjusted to heavy duty (rugged mobile computer).

This kind of specialized devices can be extensively applicable in modern healthcare systems, hence the client application for the EMeH Platform was created for this mobile platform above all. However, keeping in mind actual statistics, Android version of client application was implemented to reach wider group of users and devices as well.

### III. GENERAL CONCEPT

The EMeH Platform was built to verify the capabilities of modern mobile devices in terms of creating e-healthcare [6] platform. E-healthcare associated with mobile devices is commonly called in the literature as mobile healthcare—*m-healthcare* or *m-health* [7]. The EMeH Platform is composed of client application for mobile devices and web service environment, which runs medical applications as services. Platform architecture was based on the SOA paradigm, which can be described as a modular, scalable and interoperable approach for building systems architecture. Thanks to these features, the SOA is the appropriate approach for placing a variety of simple applications in distributed network as web services. Basic client application was built using thin-client architecture with adaptive user interface. Furthermore, an effort was made to apply SOA guidelines to mobile devices—special services located at client's side are also available for created application.

The SOA paradigm is a crucial idea, which affected the design of this *m-healthcare* system. Frequently the infrastructure of medical facilities is expanded and its elements have to cooperate with each other. Hence its necessary to create separated modules with diverse functionalities on various devices and to provide communication between them. The SOA favors modularity of systems and ensures easy scalability. In addition usage of common interface (such as REST [8]) for accessing services assures interoperability of created applications. Such an architecture enables creation of web services which will be available to many users on various devices. Moreover, it is easy to access these services, because users need to install only client applications on their devices. The web service environment was created as a runtime environment for web services to simplify design and deployment of applications.

The assumption that every module of the system can be a separate web service in the distributed network, enforces client-server architecture of interaction between mobile devices and applications. Thus, the EMeH Platform can be

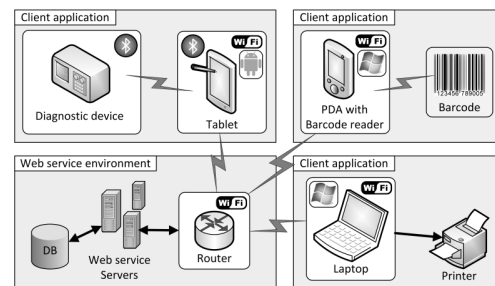


Fig. 1. General system idea

divided into two basic parts:

- *Client application* - created as a thin-client application working on mobile device,
- *Web service environment* - real medical applications consumed by client application and running in prepared runtime environment.

General idea of the system is presented in the Figure 1. To communicate with web service, every device needs only client application installed on the device. Main medium of communication between mobile device and web service is wireless network. However, it is possible to use Bluetooth technology or other ones to communicate with certain additional devices (at client's side) as well.

#### A. Client application

Client applications installed on mobile devices act similarly to web browsers, since they are thin clients. What makes difference between them is the fact, that web browsers connect to servers to browse web pages, whereas discussed client applications communicate with web services located on servers dedicated for purposes of medical institutions. Graphical user interface in the client application is generated based on data obtained from services. It determines web services' ability to control the activity of client application. Client application as a thin client makes it possible to consume various services simultaneously and the effect of such actions depends solely on service creator.

Additionally, client applications have the possibility to use services located at client's side - *local services*. This kind of service is created as an extra library and extends functionality of client application, therefore it can be called *plugin*. Plugins allow client application to use device specific peripherals such as barcode scanners, Bluetooth communication modules, RFID reader modules, etc. It is a significant mechanism of easy extending functionality with virtually no restrictions. Moreover, it provides extensions to client application without necessity to reinstall it. This gives web services the ability to command mobile device to do extra operations such as reading barcode after defined user action (e.g. pressing the button) or communicating with other devices using Bluetooth module.

Main tasks of client application running on mobile device are invocation of services by sending requests, processing of received responses from services and presentation of processed

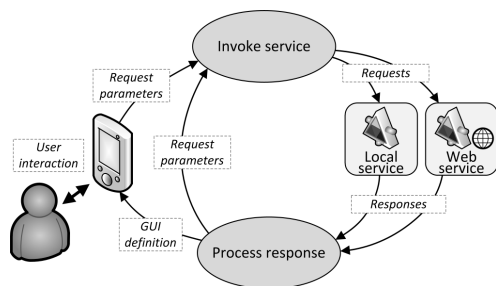


Fig. 2. Client-Service interaction flow

data to the user. It needs to be emphasized that client application does not interfere with content of presented data in any way, it only affects the way of displaying data on the screen.

The general application flow of client application is shown in the Figure 2. Every user action on mobile device triggers invocation of certain service, which in consequence can change the data displayed on device's screen. Description of mappings between user actions and service invocations is included in responses received by client application. In short, such behavior can be described as a feedback loop.

Invocation of service can be parameterized by additional parameters obtained from user interface or provided in preceding responses from services. User actions can be explained as any interaction with client application - for example it can be pressing the button, inserting text into textbox or even changing the orientation of the screen (rotating device). On the other hand, service invoked by client application, in general, returns commands in order to create or modify the graphical user interface, or to send request to some services. Summarizing, the information flow in client application is *event-driven* - it is triggered and controlled by actions performed by user.

### B. Services

Services are the part of system responsible for data processing. There are two kinds of services:

- Web services – installed in runtime environment for web services and located on servers, accessed by client applications using Wireless Local Area Network (WLAN). According to the SOA paradigm, medical applications for the EMeH Platform are designed as web services;
- Local services for mobile device – client application can utilize also services located directly on mobile device. This element expands the interpretation of the SOA paradigm through having services located not only on servers, but also as software (especially libraries) working at the client's side.

Figure 3 presents the distribution of services, which can be accessed by client application by sending requests. It shows clear division to local services and web services. Local services can do various operations on mobile device, inter alia scanning barcodes, communicating via Bluetooth module or using GPS receiver. Whereas web services are medical applications situated in runtime environment.

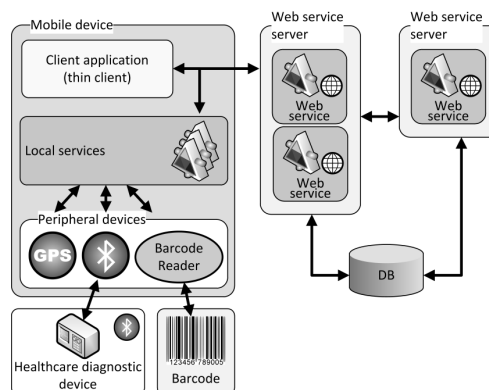


Fig. 3. Service types

In general, the network communication in general causes many security threats [9]. E-healthcare systems in many instances use confidential data (such as passwords, patient personal data, etc.), hence to protect against intrusive access of unauthorized parties, the system has to provide safe data transfer. To protect from unauthorized access, the EMeH Platform encrypts transferred data using HTTPS protocol with Secure Socket Layer (SSL) certificates. Besides in runtime environment there is a user session state management, which goal is to maintain information about user identity and activities.

## IV. CLIENT APPLICATION

In this section we describe architecture of the client application for the EMeH Platform in detail. Further, we discuss self-updating application model applied to the client application.

### A. Client architecture

As it was mentioned in the previous section, application running on mobile device is based on the thin client architecture. Thereupon, its work is focused on effective service invocation and proper presentation of received data. However, it's important to note that while processing responses from services application neither does not interfere, nor modifies the content of these responses, but only transforms them to graphical user interface.

There can be distinguished three basic layers of the client architecture: presentation layer, logic layer and communication layer. Each of them is described below.

#### Presentation layer

Client application was designed to work with various mobile devices to achieve interoperability in the EMeH Platform. These devices can differ in many ways, starting with screen parameters, such as screen resolution, aspect ratio or pixel density. Apart from screen differences, devices can run on diverse operating systems, what has an influence on methods of user interface generation. To obtain uniform way of defining user interface, client applications use common XML-based protocol of screen definitions. First client application

implementations were made for Windows Mobile and Google Android platforms, and both use the same protocol for user interface generation.

If the client application sends request to web service, then it expects response containing description of elements, which will be shown on the screen. Web service formats responses in such a way, that it does not rely on mobile device characteristics (screen parameters, operating system, etc.), thus screen definition protocol is platform independent by design. Therefore it is fundamental to transform every response to representation suitable for given device. These operations are performed by client's presentation layer.

Discussed transformation of the response and the whole user interface management itself involves some time overhead noticeable to the user. For that matter, we applied double buffering of user interface, which significantly improves screen content replacement time. It can be described shortly as two image buffers - first for current user interface content (visible to the user) and second not visible at the moment. When response from service is being processed, content of the first buffer is displayed on the screen. While in the background transformation of user interface is in progress. Eventually, when processing finishes, buffers are swapped - second buffer becomes the first one (is visible on the screen), and the first buffer goes into the background.

#### *Logic layer*

Logic layer is a part of the system where messages from services are processed to create user interface. Besides that logic layer manages events fired by user interaction with presentation layer.

Message processing means identifying type of response contained in the message and executing adequate operation depending on that type. There are three types of service responses:

- response demanding from client application to create new user interface and display completely new data on the screen (whole application window has to be recreated),
- response demanding update of certain element on the screen, i.e. change of any attribute of whichever element (for example text or color substitution on chosen label),
- response demanding client application to invoke some service with given parameters.

The message from service may contain several responses of different types - this allows to receive many screen modification or service invocation orders. Furthermore, messages can include additional data affecting client application behavior - *metadata*. Metadata can show message window, store temporary value on the device or even quit the application.

In addition to service response interpretation, logic layer has to acquire the information about events fired by user interactions (when and where was text entered, button pressed, screen touched etc.). It's essential, because process of sending request from client application to service is executed as a reaction to certain user action - as it was described in the Figure 2. To control how application should react for user

interaction (make decision which services should be called and with which parameters), the logic layer has to monitor every event fired in the presentation layer. It indicates that logic layer is responsible for interpreting user actions, formulating requests to services linked with this actions and transferring required information to communication layer, which connects directly to appropriate services.

#### *Communication layer*

Communication layer is responsible for invocation of services. As it was described in section III-B services are divided into two types: web services and local services.

Web services are basically applications deployed in runtime environment for web services, based on the SOA and having the common REST-style interface as an access method.

On the other hand local services are services located physically on mobile device. Local services are similar to *plugins* located e.g. in shared libraries, which can perform operations on client application requests. They can serve as e.g. library for device I/O operations or peripherals management - such as barcode scanners, RFID transceivers, GPS receivers, etc. To attach local service to the client application, one only has to add new entry to the application configuration file. Usage of local service does not require network connection (unless it uses it). Besides standard synchronous invocation of web services and local services, there is a possibility to invoke local services asynchronously, what allows continuous processing without waiting for response from service.

#### *B. Update mechanism*

To spare users and administrators contingent problems related to updating version of application on every single mobile device with the EMeH Platform, a mechanism called *Updater* was created. It is an implementation of self-updating application model, which utilizes a small update application (*Updater*) to alter main client application. Every client application before start uses this additional application to connect to specially exposed web service containing all versions descriptions to check whether there are any updates particular for it's device. If newer version of application or some local services (created for certain device) are available, they are automatically downloaded and installed on mobile device. It guarantees consistency between all client applications on every mobile device working within the EMeH Platform, compatibility with all web services and simple maintenance of versions in network with many different mobile devices.

## V. WEB SERVICE ENVIRONMENT

Runtime environment of web services was created to provide web services with the set of common functionalities. There are two general parts of the environment:

- web services, i.e. applications running in runtime environment,
- preprocessing modules, providing web services with essential system features, i.e. authentication etc.

Figure 4, discussed in detail below, presents these elements with regard to control flow among them (marked with arrows).

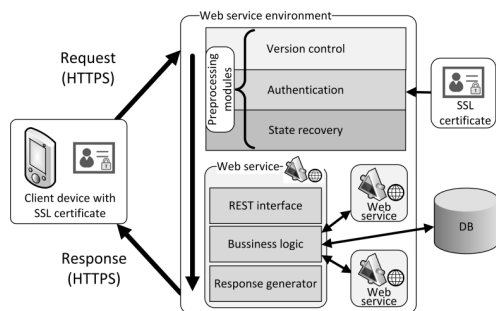


Fig. 4. Request processing flow

### A. Common preprocessing modules

In web service environment, the following preprocessing modules are implemented:

- module for controlling version of client application,
- user authentication module,
- state recovery module, used e.g. after mobile device failure

*Version control* module verifies version of application installed on mobile device that connects to the web service environment. It is important, because inconsistent versions of client-side application and web service-side environment can ensue unpredictable errors during communication. Every request sent from client to web services contains information about application version. If application version is not compatible with web service environment version, then client receives a response, which demands necessary updates to be made. Afterwards, the *updater* application (see: IV-B) on device downloads new version from the server and installs it automatically.

After verification of the client application version, the request is advanced to *authentication* module. SSL certificates are used to assure secure communication between client and services. These certificates, installed on both sides of connection, provide data encryption and identity verification for both client application, and web service environment. To positively pass authentication process, device certificate has to be compatible with corresponding certificate of web service environment.

*State recovery* module serves as mechanism of restoring interrupted user session. Whereas logging to the system, state recovery module checks whether previous user session was terminated properly. Negative result means that there exists high risk, that session was interrupted during users' work, hence module tries to recover it to the user. What is the most important, session is associated to a specific user - not to device which was used to send requests to web service environment. It gives possibility to quickly end up working on broken device and continue working on another, without losing any effects of work.

### B. Web service structure

Structure of web services working in runtime environment can be divided into three main parts:

- REST interface used for communication,
- business logic of application,
- response generator for client application.

Every web service deployed in web service environment can be accessed by clients via *REST interface*. REST is architecture style for stateless communication using HTTP protocol (or encrypted HTTPS) in the EMEH Platform. According to the REST principles, access to the service is enabled by methods of HTTP protocol: GET, POST, PUT and DELETE, meaning getting, creating, modifying and deleting data respectively. Messages exchanged between clients and web services have structure of XML documents as a result of platform independent protocol described in section IV-A.

*Business logic* is the only element of web service environment which provides certain functionalities directly to client applications. It defines specifications for client application: which events caused by the user of application will be linked to which services (web services as well as local services) and how the result will be presented i.e. which elements will be displayed on the screen. It is neither not dictated in any way, nor limited, and it depends solely on the author's concept—communication with other available services, databases, filesystem or server software is allowed. In other words, business logic module of web service has strict influence on how client-service interaction presented on Figure 2 will look like.

It is important, that all data prepared in business logic need to be passed to the *response generator* module, because it creates responses for mobile devices using platform independent protocol. This module is responsible for transformation of internal representation of data obtained from business logic of web service into format comprehensible for client applications and based on three types of responses discussed in section IV-A.

## VI. RELATED WORK

In the last decade m-healthcare systems have become a popular subject of scientific researches [10], [3], [11]. Wide spectrum of work constitutes systems for continuous monitoring of patients, what can be achieved by wearing special devices [12]. Those systems are based on so-called Body Area Networks [13]. This kind of network is dynamically developing sector of IT, in which mobile devices are used to e.g. monitor of vital signs [14] and report to a doctor every potential threat for patient's health [15].

Another issue, which is highlighted in the literature is using mobile devices in everyday work of medical staff [16]. Attempts were made to use RFID or barcode technologies to identify patients on hospital wards [17], [18], [19] or medicaments in hospital pharmacy [20]. Mobile devices with the GPS receivers were used to track elderly patients outside hospitals [21], [22]. Wireless access to basic functions of existing hospital information systems is also a big challenge

[23]. Mobile devices are used as terminals for collecting and viewing patient's medical documentation [24] or to prescribe medicines [25], [26].

The SOA assures interoperability [27] and ease of data exchange between separate medical systems [28]. In [29], [30] the SOA-based e-healthcare systems were presented, [31] contains description of the prototype SOA-based m-healthcare system for cellular phones, whereas [32], [33] show attempts to use REST interface in healthcare systems.

In the literature there were widely described thin-client architecture [34], [35] and automatic generation of user interface. There also appeared positions about adaptive user interface [36], context-aware user interface [37] and dynamical user interface [38], [35], however none of mentioned was related to medical applications.

## VII. CONCLUSIONS

This article presents a new m-healthcare platform, which utilizes modern mobile devices for e-healthcare applications. Among basic requirements for distributed system for mobile devices the most important are: modularity, accessibility, scalability and interoperability. Therefore platform was built using service oriented architecture and the access to services was assured by the RESTful interface, which enables stateless communication over HTTP protocol.

The EMeH Platform combines features of the systems mentioned in VI. It consists of client application and runtime environment for web services. Client application is a thin-client application with user interface generated dynamically of data received from services. Services can be divided into web services and local services. Web services are applications settled in runtime environment, which ensures common mechanisms for authentication, version control and user session recovery. On the other hand, local services are essentially plugins available on mobile devices. The idea of services located at client's side arises as an interpretation of the SOA paradigm in terms of a single mobile device. Local services are capable of interacting with e.g. mobile device's peripherals, such as Bluetooth, RFID or barcode modules, thus can be highly suitable for hospital information systems.

Great emphasis was put on ensuring interoperability of the created platform. To use the EMeH Platform on mobile device with other than Microsoft Windows Mobile or Google Android operating systems installed, it is needed to implement new client application for displaying user interfaces (defined by services) in new system specific manner. However, communication protocol and activity of services in web service environment is universal for every mobile platform due to the common RESTful interface of invoking services.

The EMeH Platform was implemented as a part of the *Hospital Information System Eskulap, Poznan University of Technology, Poland* [39]. In web service environment there are already some fully implemented applications, e.g. mobile dispensary in pharmacy or application for monitoring of patients on hospital wards, which currently are being performed acceptance tests in hospitals. On mobile devices there are

available client application versions for Microsoft Windows Mobile 5.0+ and Android 2.2+ operating systems.

In the future there will be more client applications implemented to extend the scope of the platform on additional operating systems, such as Apple iOS or Microsoft Windows Phone 7. Besides that, there will be created a new tool for graphical screen definition creation in web services. There will be also implemented more web services for the EMeH Platform, based on existing software of the Eskulap System. Reason is that many of existing applications for desktop computers could be easily extended by the access from mobile devices, such as: laboratory, medical imaging or ambulance service applications.

## REFERENCES

- [1] N. M. Menon, B. Lee, and L. Eldenburg, "Productivity of information systems in the healthcare industry," *Information Systems Research*, 2000.
- [2] N. Oriol *et al.*, "Calculating the return on investment of mobile healthcare," *BMC Medicine*, vol. 7, no. 1, 2009.
- [3] J. H. Wu, S. C. Wang, and L. M. Lin, "Mobile computing acceptance factors in the healthcare industry: A structural equation model," *International Journal of Medical Informatics*, vol. 76, no. 1, pp. 66–77, jan 2007.
- [4] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2005.
- [5] R. Cozza *et al.*, "Market share analysis: Mobile devices, worldwide, 1q11," Gartner, Tech. Rep., may 2011.
- [6] G. Eysenbach, "What is e-health?" *Journal of Medical Internet Research*, vol. 3, no. 2, June 2001.
- [7] R. Istepanian, S. Laxminarayan, and C. S. Pattichis, *M-Health: Emerging Mobile Health Systems*, ser. Topics in Biomedical Engineering. Springer, 2006.
- [8] R. T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, University of California, Irvine, 2000.
- [9] D. Welch and S. Lathrop, "Wireless security threat taxonomy," in *IEEE Systems, Man and Cybernetics Society Information Assurance Workshop*, jun 2003, pp. 76–83.
- [10] U. Varshney, "Using wireless technologies in healthcare," *International Journal of Mobile Communications*, vol. 4, no. 3, pp. 354–368, feb 2006.
- [11] C. Free, G. Phillips, L. Felix, L. Galli, V. Patel, and P. Edwards, "The effectiveness of m-health technologies for improving health and health services: a systematic review protocol," *BMC Research Notes*, vol. 3, no. 1, pp. 1–7, 2010.
- [12] V. Jones *et al.*, "Mobihealth: Mobile services for health professionals," in *M-Health*, ser. Topics in Biomedical Engineering. Springer US, 2006, pp. 237–246.
- [13] H. B. Li and R. Kohno, "Body area network and its standardization at IEEE 802.15.BAN," in *Advances in Mobile and Wireless Communications*, ser. Lecture Notes in Electrical Engineering. Springer Berlin Heidelberg, 2008, vol. 16, no. 4, pp. 223–238.
- [14] E. Monton *et al.*, "Body area network for wireless patient monitoring," *IET Communications*, vol. 2, no. 2, pp. 215–222, feb 2008.
- [15] V. Gay, P. Leijdekkers, and E. Barin, "A mobile rehabilitation application for the remote monitoring of cardiac patients after a heart attack or a coronary bypass surgery," in *Proceedings of the 2nd International Conference on Pervasive Technologies Related to Assistive Environments*. Corfu, Greece: ACM, 2009, pp. 21:1–21:7.
- [16] Y. C. Lu, Y. Xiao, A. Sears, and J. A. Jacko, "A review and a framework of handheld computer adoption in healthcare," *International Journal of Medical Informatics*, vol. 74, no. 5, pp. 409–422, 2005.
- [17] L. Cheng-Ju, L. Li, C. Shi-Zong, W. C. Chen, H. Chun-Huang, and C. Xin-Mei, "Mobile healthcare service system using RFID," in *IEEE International Conference on Networking, Sensing and Control*, vol. 2, Taipei, Taiwan, mar 2004, pp. 1014–1019.
- [18] A. T. van Halteren *et al.*, "Mobile patient monitoring: The Mobihealth system," *The Journal on Information Technology in Healthcare*, vol. 2, no. 5, pp. 365–373, 2004.

- [19] D. C. Baumgart, "Personal digital assistants in health care: experienced clinicians in the palm of your hand?" *The Lancet*, vol. 366, no. 9492, pp. 1210–1222, October 2005.
- [20] J. M. Rothschild, T. H. Lee, T. Bae, and D. W. Bates, "Clinician use of a palmtop drug reference guide," *Journal of the American Medical Informatics Association*, vol. 9, pp. 223–229, 2002.
- [21] L. Chung-Chih, L. Ping-Yeh, L. Po-Kuan, H. Guan-Yu, L. Wei-Lun, and L. Ren-Guey, "A healthcare integration system for disease assessment and safety monitoring of dementia patients," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, no. 5, pp. 579–586, 2008.
- [22] S. H. Chew, P. A. Chong, E. Gunawan, K. W. Goh, Y. Kim, and C. B. Soh, "A hybrid mobile-based patient location tracking system for personal healthcare applications," in *28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, New York, NY, sept 2006, pp. 5188–5191.
- [23] B. Lin and J. A. Vassar, "Mobile healthcare computing devices for enterprise-wide patient data delivery," *International Journal of Mobile Communications*, vol. 2, no. 4, pp. 343–353, dec 2004.
- [24] J. Bergmann, O. J. Bott, D. P. Pretschner, and R. Haux, "An e-consent-based shared EHR system architecture for integrated healthcare networks," *International Journal of Medical Informatics*, vol. 76, no. 2-3, pp. 130–136, March 2007.
- [25] E. G. Poon *et al.*, "Medication dispensing errors and potential adverse drug events before and after implementing bar code technology in the pharmacy," *Annals of Internal Medicine*, vol. 145, no. 6, pp. 426–434, 2006.
- [26] F. Kart, G. Miao, L. E. Moser, and P. M. Melliar-Smith, "A distributed e-healthcare system based on the service oriented architecture," in *IEEE International Conference on Services Computing*, July 2007, pp. 652–659.
- [27] S. Daskalakis and J. Mantas, "The impact of SOA for achieving healthcare interoperability," *Methods of Information in Medicine*, vol. 48, no. 2, pp. 190–195, 2009.
- [28] J. Brzeziński, S. Czajka, J. Kobusiński, and M. Piernik, "Healthcare integration platform," in *2011 5th International Symposium on Medical Information Communication Technology*, Motreux, March 2011, pp. 52–55.
- [29] F. Kart, L. E. Moser, and P. M. Melliar-Smith, "Building a distributed e-healthcare system using SOA," *IT Professional*, vol. 10, no. 2, pp. 24–30, mar-apr 2008.
- [30] W. S. Ng, J. C. M. Teo, W. T. Ang, S. Viswanathan, and C. K. Tham, "Experiences on developing SOA based mobile healthcare services," in *IEEE Asia-Pacific Services Computing Conference*, Singapore, dec 2009, pp. 498–501.
- [31] M. Savini, A. Ionas, A. Meier, C. Pop, and H. Stormer, "The eSana framework: Mobile services in eHealth using SOA," in *European Conference on Mobile Government*, 2008.
- [32] L. Griffin, C. Foley, and E. de Leastar, "A hybrid architectural style for complex healthcare scenarios," in *IEEE International Conference on Communications Workshops*, Dresden, June 2009, pp. 1–6.
- [33] L. W. F. Andry and D. Nicholson, "A mobile application accessing patients' health records through a REST API," in *4th International Conference on Health Informatics*, 2011, pp. 27–32.
- [34] R. A. Baratto, L. N. Kim, and J. Nieh, "THINC: a virtual display architecture for thin-client computing," *ACM SIGOPS Operating Systems Review-SOSP '05*, vol. 39, no. 5, pp. 277–290, December 2005.
- [35] J. Kim, R. A. Baratto, and J. Nieh, "pTHINC: a thin-client architecture for mobile wireless web," in *Proceedings of the 15th international conference on World Wide Web*. Edinburgh, Scotland: ACM, 2006, pp. 143–152.
- [36] M. Al-Turkistany, A. Helal, and M. Schmalz, "Adaptive wireless thin-client model for mobile computing," *Wireless Communications and Mobile Computing*, vol. 9, no. 1, pp. 47–59, January 2009.
- [37] T. Butter, M. Aleksy, P. Bostan, and M. Schader, "Context-aware user interface framework for mobile applications," in *27th International Conference on Distributed Computing Systems Workshops*, Toronto, Ont., jun 2007.
- [38] K. Luyten and K. Coninx, "An XML-based runtime user interface description language for mobile computing devices," in *Interactive Systems: Design, Specification, and Verification*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2001, vol. 2220, pp. 1–15.
- [39] "Eskulap-hospital information system," Poznan University of Technology. [Online]. Available: <http://www.systemeskulap.pl/>





# Semantic Interoperability for Infectious Diseases Reporting System

Murugavell Pandiyam  
Research Student,  
Kalasalingam University,  
Krishnan koil, India.  
Email:

murugavell.pandiyam@tecom.ae

Osama El Hassan  
Sr. Technical Consultant  
TECOM Investments,  
Dubai, UAE.  
Email:

osama.elhassan@tecom.ae

Zakaria Maamar  
Professor,  
Zayed University,  
Dubai, UAE.  
Email:

zakaria.maamar@zu.ac.ae

Pallikonda Rajasekaran,  
Associate Professor,  
Kalasalingam University,  
Krishnan koil, India.  
Email:

mpraja@klu.ac.in

**Abstract**—The healthcare setting is multifaceted, comprised of many different components including private, governmental, and regulatory agencies. There is always a necessity of timely and reliable information exchange among these agencies especially on “Infectious Disease” information due to their criticality. The heterogeneity of the systems used by these agencies has led us into designing and developing an interoperable solution to exchange data effectively among several independent yet collaborating health authorities at both state and national levels. This research work articulates the efforts put into achieving an interoperable “Infectious Diseases Reporting System” that incorporates ontology-based semantic rules to align different infectious disease coding standards and to deploy Web services for collecting data from remote sources. This effort is a first step towards achieving a policy-based interoperable Infectious disease monitoring system which can be used across different yet collaborating regulatory agencies.

**Index Terms**—Interoperability, Semantics, Healthcare system.

## I. INTRODUCTION

**I**NFECTIONOUS diseases outbreaks demand a timely and proportional response to mitigate effects on public health. Management of these outbreaks is becoming a growing concern in public health, as it requires extreme actions and coordination between governing authorities at both state and national levels. Tracking and identifying emerging infectious diseases and epidemic outbreaks in particular poses critical challenges on public health researchers and practitioners. Dealing with large numbers of incoming reports and alerts requires an automated system which performs real time analysis on a centralized repository of collected clinical information. As such, intrinsically intensive information exchange should be exercised between healthcare facilities and this repository, which is normally operated by a regulator. Regulators use these collected information to identify, manage, and investigate infectious diseases outbreaks. Such data, when visually and adequately represented, support the education of healthcare providers within participating facilities and improves the outcome of disease outbreaks management.

## A. Challenges

Gathering information from various sources (i.e., healthcare facilities) in real-time is a challenge because of the diverse and heterogeneous nature that labels interfacing healthcare applications. Moreover, the capability of aligning and integrating healthcare data standards varies drastically from one coding system to another. There is a necessity here to design an advanced interoperable system that is empowered with a semantic layer to retrieve and map the information onto a unified standard to support alerting and surveillance. The design of such an interoperable system is complex and challenging due to the following factors:

1. Each healthcare facility has its own software to manage patient’s data e.g., Physician Practice Management System (PPMS) and Electronic Medical Records (EMR). Moreover, even standardized EMR systems might be interfaced differently and sub-systems such as Lab systems are isolated in terms of their used standards (i.e., proprietary standards) and thus raise extra integration challenges.

2. Each facility has its own policies and procedures for reporting patient data.

3. The network infrastructure and real time reporting may affect the data availability.

4. Last but not least, from a regulatory body point of view, since the infectious disease is a growing concern in public health, it is necessary to collaborate with other health authorities to exchange and manage the related information and alerts.

To address the aforementioned factors, a platform is needed to enable interoperable information exchange between independent healthcare systems.

## B. Proposed Solution

This research work presents the design and development of semantic information integration for Infectious Diseases Reporting system using Web services as a core component for information exchange with ontology-based rules. Ontology has been predominantly used to represent well-categorized concepts and to support mappings between models. The central ontology has been designed for the system and expressed in OWL [1][2] so as to semantically assign dis-

eases represented in various coding standards to a single unified entity.

The rest of the paper is organized as follows. Section 2 provides some background information and literature survey. Section 3 presents the architecture and implementation details. Section 4 discusses some design decisions and their impact on the implementation. Section 5 summarizes the research outcomes and outlines future work.

## II. LITERATURE SURVEY

In [3] Iqbal et al. propose an ontology based model for an Electronic Medical Record that targets Chronic Disease Management with focus on providing a coherent information structure to support other acute diseases and co-morbidities. However, this model is patient-centric as it comprises longitudinal information of the patient, but it lacks the capability of collecting data from heterogeneous system.

In [4] Sampalli et al. propose an ontology model for patient profiles. In this model, frequently occurring procedure/diagnosis terms are extracted from patient charts, converted into SNOMED CT (Systematized Nomenclature of Medicine-Clinical Terms) and then categorized using the ontology concepts. The categories include medical, physical, psychosocial, rehabilitation, and nutrition. However, since the rules associated with the ontology are applied on patient charts for recurrent medical terminologies, the degree of accuracy between the doctors' diagnosed term and the converted concept identifiers cannot be easily measured. Such a study can be helpful for statistical analysis but not on patient treatment.

In [5] Ngamnij Arch-int et al. describe a semantic bridge ontology that provides Web services with the same service information but different parameters – according to the proprietary database structures – to integrate or exchange data and obtain common semantic meanings.

Iqbal et al. and Sampalli et al. demonstrate the use of ontology based rules in conceptual categorization and the value-added of ontology to enhance the expressiveness of concept. In our work, we have used “criticality” in “Infectious Disease” as the key indicator of expressiveness based on the diagnosis code. Regarding Arch-int et al. the ontology rules are not used to create a conceptual model for the relationships of the data from the disparate systems.

## III. OUR INFECTIOUS DISEASES REPORTING SYSTEM

### A. Design

We have divided our system into the following modules: Interoperable Web Service, Ontology-based Infectious Disease, Semantic Bridge to communicate with other health authorities, and Business Intelligence and Reporting.

Our infection diseases reporting system is built upon a set of Web services that interface with existing systems to collect infectious diseases. The content of these systems is structured differently with different standards (e.g., ICD-10 [6] or ICD-9 [7]). It is worth noting that some local health-

care facilities send their disease codes as “descriptive text” to Web services. Thus we need an ontology bridge to map counterpart meanings of single concept (e.g., disease name “malaria”) that are defined by different standard codes into a uniform entity.

The regulatory body defines “criticality” of each infectious disease. Each infectious disease has several characteristics properties but we focused only on “Criticality. We utilized OWL to build the relationship between ICD-10 codes and ICD-9 [7] class as “union-of” relationship to streamline a unified infectious disease classification range: Highly Critical, Critical, etc. to establish an “infectious disease centric” ontology model on patient data. The collection of these ontology-based entities reflects our conceptual model. The role of “Ontology-based Infectious Disease” module is to build such a model in order to represent and reason about the “criticality” of infectious diseases.

For a single regulatory body, there is a necessity to communicate its captured infectious diseases incidents to other health authorities at state and national level. Therefore we provide a web service based mechanism to exchange collected incidents data. Each health authority defines its own web service WSDL (Web Services Description Language) description. Since the WSDL parameters for web service methods will be different for each health authority, although the information are same, it is necessary to build ontology based rule to get the semantic meaning of the appropriate parameters. The role of “Semantic Bridge” module is to bridge the internal artifact properties with the WSDL parameter names of external health authority's web services.

The “Business Intelligence and Reporting” module is built to provide the regulatory body a statistical research and education program.

The architecture of our infection diseases reporting system is built upon 5 layers (Fig. 1): Resource, Mediation, Application Layer, Business Intelligence and Reporting, and Semantic Bridge.

The Resource Layer hosts a Centralized Data Repository, which has Patient Profile information that is stored in an Object Relational Database.

The mediation layer hosts three components:

- *Interoperable Web service*

It is implemented in SOAP and facilitates the Infectious Diseases data retrieval that will be triggered by the facilities when a patient encounter of infectious disease occurs. The Web service converts the retrieved patient data into internal artifacts and passes the information through the Ontology Engine.

- *Ontology Engine*

It facilitates building the conceptual model of the “Criticality” of the disease based on the data retrieved in Web services by issuing SOQL query to the ontology rules on “Infectious Disease”. Fig. 2 shows the “Architecture of Ontology”.

gy engine for Semantic Bridge”. The “Cholera” disease is denoted in OWL as follows.

```

InfectiousDisease rdf:ID="Cholera">
  <CriticalityCondition rdf:resource="&xsd:string">
  HighlyCritical
  </CriticalityCondition>
  <DiseaseDescription rdf:resource="&xsd:string">
  Cholera is an infectious Disease
  </DiseaseDescription>
  <owl:unionOf rdf:parseType="Collection">
  <owl:Thing rdf:about="#CholeraCD10" />
  <owl:Thing rdf:about="#CholeraCD9" />
  </owl:unionOf>
    
```

• Alert system

It is implemented as an e-mail notification system that informs the regulatory body authority when a patient has some infectious diseases.

The Application Layer consists of Web interface screens for viewing Patient profile data and also for assigning proper coding to the reported Infectious Disease if it is submitted as a descriptive text without standard coding terminology. If the “descriptive text” contains medical terminology, there is an automated search of the ICD -10 with that medical term. If this search manages to fetch the appropriate code, it will stamp that internal artifact disease coding as “Automated text conversion”. We have built web portal on top of the ICD -10 to search for codes or description, which will help the medical coding experts to assign “descriptive text” proper coding and convert them to conceptual model.

The Business Intelligence and Reporting Layer retrieves the data stored in Centralized Data Repository and transforms them into Dimensional data model and stored in "Warehouse" Object Relational database. We have built "Reports".

The Semantic Bridge Layer allows to address the problem of semantic discrepancies of WSDL parameter names. We have built some common ontology rules on the parameter names between the internal artifact property names and WSDL parameter names. For instance, let us take an internal artifact property called “patient name” that is codified as “PatName” using a WSDL parameter and “PatientName” as another internal artifact property. These two parameters are semantically the same and hence, are described in OWL model as “same-as” relationship. This mapping is illustrated for other parameters in Table I.

TABLE I.  
WSDL PARAMETER NAMES WEB SERVICES ARE MAPPED TO INTERNAL ARTIFACT PROPERTY

Internal artifact property name	WSDL parameter name for the web service 1	WSDL parameter name for the web-service 2
PatientName	Name	PatName
MRNNo	MedicalRecordNumber	MRNNumber
Healthcard	HealthcardNo	HealthcardNumber
DateOfBirth	DOB	BirthDate
Gender	Sex	Gender
DiagnosisCode	ICD10Code	ICD9Code
DiagnosisStatus	Status	Status
PhysicianName	ClinicianName	Providername

The following “Patient Name” property in OWL

```

<Parameter rdf:ID="PatientName">
  <parameterFrom>InternalArtifact</parameterFrom>
  <parameterID>PatientName</parameterID >

  <owl:sameAs rdf:resource="#Name">
  <owl:sameAs rdf:resource="# PatName">

</ Parameter>
<Parameter rdf:ID="Name">
  <parameterFrom>Webservice1</parameterFrom>
  <parameterID>Name</parameterID>

</ Parameter>

<Parameter rdf:ID="PatName">
  <parameterFrom>Webservice2</parameterFrom>
  <parameterID>PatName</parameterID>

</ Parameter>
    
```

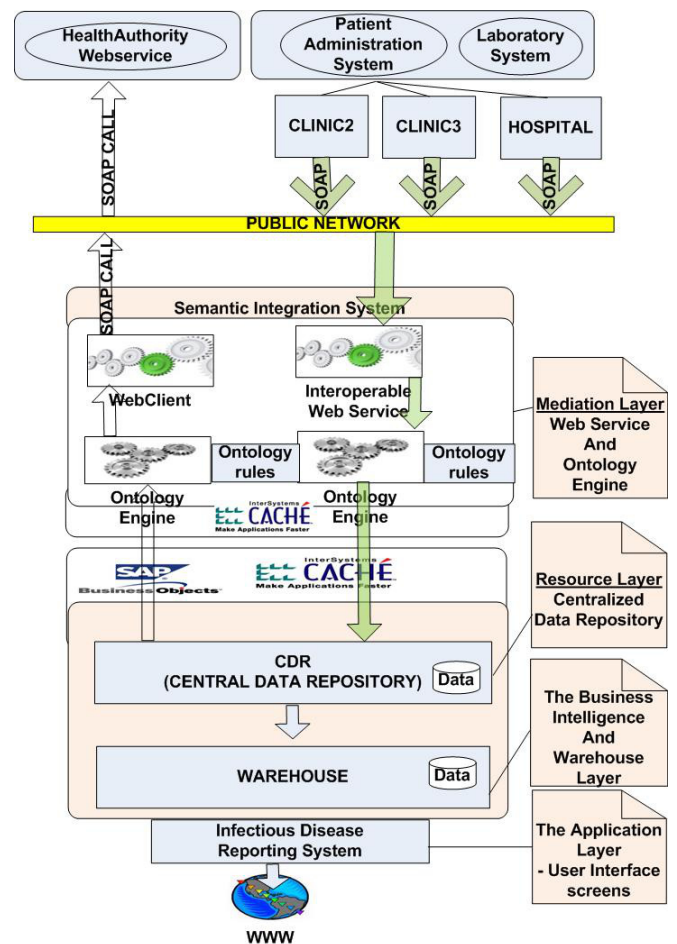


Fig 1 Architecture of Infectious Disease Reporting System

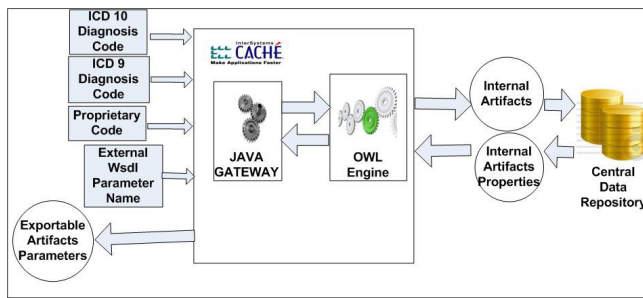


Fig 2 Architecture of Ontology Engine for Semantic Bridge

### B. Dataflow

When a patient is diagnosed with an infectious disease, the clinical system launches a Web service that will send the data to our system. These data will be managed by some ontology rules before they get stored in the Central Data Repository (CDR) as "Patient Profile". Moreover these data will be subject to some mining as follows.

1. The Healthcare facilities send the data of infectious disease to the Web service.
2. The Web service delivers the data to the semantic engine. It is implemented on Intersystem's cache having a "Java Gateway" and "OWL" engine. This engine converts the data into a conceptual model of internal artifacts.
3. The internal artifacts finally are stored in the "Central Data Repository".
4. The data are mined in the "warehouse repository" for statistical reports and charts generation.
5. To report the infectious disease to an external health authority Web service, the ontology relationships are established between the internal artifacts parameters and the external callable WSDL parameters. This will match the semantic meaning of the WSDL parameters and makes web service client as semantically integrated with the external WSDL parameters.

### C. Technologies Used and Delivered Components.

We have built the web service on top of Intersystem's Cache implementation. The infectious disease data collected is transformed into data mining for the statistics research, which is built on Business Objects. The web pages are built on "Cache Server Page". For the transformation of the data to the conceptual model with OWL for both Infectious Diseases and the Semantic wsdl parameters, the jena engine is implemented. Table II shows the technology components and their vendors list. Table III shows the components designed and delivered.

TABLE II.  
SUMMARY OF IMPLEMENTATION TECHNOLOGIES

Technologies	Brand
Database & Language	Intersystem's Cache [8]
Reports & Graphs	Business Objects [9]
OWL	Jena[10]

TABLE III.  
MAJOR COMPONENTS DESIGNED AND DELIVERED

Component	Description of the Component
Web service	This service exposes web method "ReportDisease" and collects for the data submitted from the external request from the Healthcare facility medical systems
User Interface screens	Web page for searching ICD 10 disease codes and description. Used by medical coding experts to assign proper coding
Semantic Bridge	To achieve semantic interoperability, we have used the ontology based rule on the internal artifact property name with the wsdl parameter names of the external web services
E-mail alerting system.	Automated e-mail system was built to notify the regulatory body for the infectious disease reporting.
Reporting system	We have built data warehouse for statistical research with ad-hoc reporting tool on Business Objects
Security	Security Layer on the Web pages with the authentication on Active Directory

## IV. DISCUSSION

The architecture presented in this paper lays down the basis for the development of an Interoperable medical application that targets multi-regulators "Infectious Disease Reporting System". The ontology developed in this study adds another layer of expressiveness whereas Web services secured the reach ability of the existing systems across different platforms. To achieve semantic integration, we have built a set of ontology rules on WSDL parameters with the internal artifacts parameters. The key limitation in this work is the inability to capture medical tests and results, which form together the fundamental support for confirming diagnosis of infectious disease. To achieve this goal, we have to expand the array of data to accommodate medical tests and their results by providing ontology vocabulary on the LOINC codes to convert them into conceptual model.

## V. CONCLUSION AND FUTURE WORK

This research work is driven by the immense needs and posed challenges for integrating different healthcare systems to monitor and manage public health indicators such as infectious diseases. As such, this work can be extended in many aspects such as putting forward a policy-based framework to allow injecting regulators rules. This direction should follow the approach specified by El-Hassan et al. [11] which allows specifying rules for accessing resources

(e.g., patients data) in both normal and emergency situations. Additionally, advances in web services that advocate collaboration between independent agents have to be incorporated. In this perspective, web services research presented by Khosravifar et al. [12] demonstrate a viable candidate for community-based web services that manage service-based interactions between equal parties.

## APPENDIX

### A. Section of WSDL Description of Web service

```
<?xml version='1.0' encoding='UTF-8' ?>
<definitions xmlns:http='http://schemas.xmlsoap.org/wsdl/http/'
xmlns:SOAP-ENC='http://schemas.xmlsoap.org/soap/encoding/'
xmlns:mime='http://schemas.xmlsoap.org/wsdl/mime/' targetNamespace =
'http://HIRAS.ae' xmlns='http://schemas.xmlsoap.org/wsdl/'>
  <types>
    <s:schema elementFormDefault='qualified' targetNamespace =
'http://HIRAS.ae'>
      <s:element name='ReportDisease'>
        <s:complexType>
          <s:sequence>
            <s:element name='FacilityID' type='s:string' minOccurs='0' />
            <s:element name='PatientName' type='s:string' minOccurs='0' />
            <s:element name='MRNNo' type='s:string' minOccurs='0' />
            <s:element name='Healthcard' type='s:string' minOccurs='0' />
          </s:sequence>
        </s:complexType>
      </s:schema>
    </types>
    <message name='ReportDiseaseSoapIn'>
      <part name='parameters' element='s0:ReportDisease' />
    </message>
    <message name='ReportDiseaseSoapOut'>
      <part name='parameters' element='s0:ReportDiseaseResponse' />
    </message>
    <portType name='InfectiouDiseaseReportingSoap'>
      <operation name='ReportDisease'>
        <input message='s0:ReportDiseaseSoapIn' />
        <output message='s0:ReportDiseaseSoapOut' />
      </operation>
    </portType>
    <binding name='InfectiouDiseaseReportingSoap' type='s0:InfectiouDis-
easeReportingSoap' >
      <soap:binding transport='http://schemas.xmlsoap.org/soap/http'
style='document' />
      <operation name='ReportDisease' >
        <soap:operation soapAction='http://HIRAS.ae/CDR.InfectiousDis-
ease.ReportDisease' style='document' />
        <input>
          <soap:body use='literal' />
        </input>
        <output>
          <soap:body use='literal' />
        </output>
      </operation>
    </binding>
    <service name='InfectiouDiseaseReporting' >
      <port name='InfectiouDiseaseReportingSoap' binding='s0:InfectiouD-
iseaseReportingSoap' >
        <soap:address location='http://HIRAS.ae/CDR.InfectiousDis-
ease' />
      </port>
    </service>
  </definitions>
```

### B. OWL representation of Infectious Disease

```
<owl:Class rdf:ID='InfectiousDisease'>
  <rdfs:label xml:lang='en'>InfectiousDisease </rdfs:label>
</owl:Class>
<owl:ObjectProperty rdf:ID='CodeScheme'>
  <rdfs:domain rdf:resource='#InfectiousDisease' />
  <rdfs:range rdf:resource='#CodingScheme' />
</owl: ObjectProperty >
<owl:DatatypeProperty rdf:ID='DiseaseDescription'>
  <rdfs:domain rdf:resource='#InfectiousDisease' />
  <rdfs:range rdf:resource='&xsd:string' />
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID='CriticalityCondition'>
  <rdfs:domain rdf:resource='# InfectiousDisease' />
  <rdfs:range rdf:resource='&xsd:string' />
</owl:DatatypeProperty>
```

## ACKNOWLEDGEMENT

This research work would have not been completed without the enormous support of Dubai Healthcare City's top executives who granted us access to their "Arab Health Award Winning Product, namely, "HIRAS- Healthcare Information Reporting and Analysis System". We particularly thank the CEO, Dr. Ayesha Abdulla and Dr. Abdulkareem Al-Olama. Additionally thanks go to Imad Choucair and Naima Shaikh from the CIO office in TECOM Investment for the continuous encouragement and support.

## REFERENCES

- [1] M. K. Smith, C. Welty, and D. McGuinness. OWL Web Ontology Language Guide. W3C Recommendation, <http://www.w3.org/TR/owl-guide/>, May 14, 2011.
- [2] W3C. Owl web ontology language-reference. LSDIS Lab, University of Georgia, 2004. <http://www.w3.org/TR/owl-ref/>.
- [3] Iqbal, A.M.; Shepherd, M.; Abidi, S.S.R., "An Ontology-Based Electronic Medical Record for Chronic Disease Management" in *Jan. 2011 44th Hawaii International Conference* pp. 4-6.
- [4] Sampalli, T. Shepherd, M. Duffy, J., "A Patient Profile Ontology in the Heterogeneous Domain of Complex and Chronic Health Conditions" in *Jan. 2011 System Sciences (HICSS), 2011 44th Hawaii International Conference* pp. 4-6.
- [5] Arch-int, N.; Arch-int, S "Semantic information integration for electronic patient records using ontology and web services model" in *April. 2011 Information Science and Applications (ICISA), International Conference* pp. 3-5
- [6] International Classification of Diseases. <http://www.cdc.gov/nchs/icd/icd10.htm>, May 14, 2011.
- [7] International Classification of Diseases, Ninth Revision (ICD-9) <http://www.cdc.gov/nchs/icd/icd9.htm>, May 24, 2011.
- [8] Intersystems Cache. <http://www.intersystems.com/cache/>, May 24, 2011.
- [9] Business Objects <http://www.sap.com/solutions/sapbusinessobjects/index.epx>, May 24, 2011
- [10] Jena A Semantic Web Framework for Java. <http://jena.sourceforge.net/>, May 14, 2011.
- [11] El-Hassan, Osama and Fiadeiro, Jos'e Luiz and Heckel, Reiko "Managing socio-technical interactions in healthcare systems" in *2008 Proceedings of the 2007 international conference on Business process management*.
- [12] Khosravifar, B.; Bentahar, J.; Moazin, A.; Maamar, Z.; Thiran, P. "Analyzing Communities vs. Single Agent-Based Web Services: Trust Perspectives", in *July 2010 Services Computing (SCC), 2010 IEEE International Conference*.



# International Workshop on Ubiquitous Home Healthcare

**P**OPULATION aging is a phenomenon affecting many countries around the world. For example in Europe the life expectancy increased from 45 years in the early twentieth century, to 80 years now. Significantly longer life leads to age-related problems and diseases. In parallel, the cost of hospital care is increasing and additionally, a lack of the qualified caregivers is observed. Development of ubiquitous healthcare technologies can improve the quality of life of assisted citizens and can curtail growth in healthcare spending fueled by aging populations, and the prevalence of obesity, diabetes, cancer and chronic heart and lung diseases. In particular, information systems integrated with wearable, mobile devices and sensor networks at home can continuously assist persons while moving out or staying at home. Ubiquitous healthcare systems used as assisted living solutions will not only help to prevent, detect and monitor health conditions of a person but will also support of elderly, sick and disabled people in their independent living.

The goal of the UHH 2011 workshop is to gather researchers and engineers working in the field of ubiquitous healthcare to present and discuss new ideas, methods, and applications of assisted living IT technologies.

## TOPICS

The workshop welcomes all work related to ubiquitous healthcare, but with a focus on the following themes (this list is not exhaustive):

- Ubiquitous healthcare information systems,
- Information processing algorithms for UHH,
- Ubiquitous services for home and mobile applications,
- Human-system interaction in UHH,
- Data mining and knowledge discovery in ubiquitous healthcare,
- Integration of sensors and devices for UHH,
- Security of ubiquitous healthcare systems,
- Ensuring the Availability, Transparency, Seamlessness, Awareness, and Trustworthiness (A.T.S.A.T.) of home and mobile systems,
- Standardization in ubiquitous home healthcare,
- Applications of UHH for elderly, sick, and disabled people,
- Elderly care monitored dosage systems,
- Welfare technology for UHH.

The proposed papers should emphasize at least one of the following aspects:

- Assisted living,
- Home care,
- Self care,
- Mobile care.

## PROGRAM COMMITTEE

**Piotr Augustyniak**, AGH-UST, Institute of Automatics, Multidisciplinary School of Engineering In Biomedicine, Poland

**Adam Bujnowski**, Gdansk University of Technology, Department of Biomedical Engineering, Poland

**Jan Cornelis**, Vrije Universiteit Brussel, Department of Electronics and Informatics, Belgium

**Jens Hauelsen**, Faculty of Computer Science and Automation, Ilmenau University of Technology, Germany

**Zdzisław S. Hippe**, University of Information Technology and Management in Rzeszow, Faculty of Applied Informatics, Poland

**Eddie YK Ng**, Nanyang Tech. University, School of Mechanical & Aero. Engg., Singapore

**Nicolas Pallikarakis**, University of Patras, Biomedical Technology Unit - BITU, Greece

**Ewaryst Tkacz**, Silesian University of Technology, Institute of Electronics, Poland

**Stefan Wagner**, University of Aarhus, Aarhus School of Engineering, Denmark

**Krzysztof Zaremba**, Warsaw University of Technology, Faculty of Electronics and Information Technology, Poland

**Artur Poliński**, Gdansk University of Technology, Department of Biomedical Engineering, Poland

## ORGANIZING COMMITTEE

**Jacek Ruminski** (chairperson), Gdansk University of Technology, Department of Biomedical Engineering, [jwr@computer.org](mailto:jwr@computer.org)

**Jerzy Wtorek**, Gdansk University of Technology, Department of Biomedical Engineering, [jerzy.wtorek@eti.pg.gda.pl](mailto:jerzy.wtorek@eti.pg.gda.pl)

**Mariusz Kaczmarek**, Gdansk University of Technology, Department of Biomedical Engineering, [mariusz.kaczmarek@eti.pg.gda.pl](mailto:mariusz.kaczmarek@eti.pg.gda.pl)





# The role of a mobile device in a home monitoring healthcare system

Marcin Bajorek

Gdansk University of Technology, ul. Gabriela  
Narutowicza 11/12, 80-233 Gdańsk, Poland  
Email: martom@biomed.eti.pg.gda.pl

Jędrzej Nowak

Gdansk University of Technology, ul. Gabriela  
Narutowicza 11/12, 80-233 Gdańsk, Poland  
Email: jedrzej.nowak.tch@gmail.com

**Abstract**—In the present study, a home monitoring healthcare system for elderly and chronic patients has been proposed. The system was developed for three types of users: assisted person, doctor and guardian. It analyzes the collected information (e.g. biomedical signals) and in case of detection of dangerous events informs physician and guardian. A mobile device has a key role in the system. It allows exchange and visualization of data to the users. This paper describes the design and implementation of a tablet in the home monitoring healthcare system, with specially developed data exchange protocol. Additionally special security features to protect data exchange were introduced. Software part of the system was made using modern technologies such as JavaFX for central unit and Android for mobile devices.

## I. INTRODUCTION

RECENTLY healthcare for elderly people has been an important research topic. The increase in life expectancy due to improvements in living standards, and medical treatments, has resulted in an aging population diseases in the last few years [1], [2]. Therefore, the modern health care system aims to enhance the safety and comfort of the patient's life while managing chronic diseases. This creates the need to develop home e-health system, which will integrate various wireless sensors for long-term monitoring and vital signs extraction of the patients. The rapid development of information and telecommunication technology has brought great revolutions in that field [3]. New features of mobile devices (smart-phones, tablets), create new opportunities to use them as devices for the management and presentation of medical data.

There are many promising systems implementing the selected function or the complex monitoring of the patient. One of the first approaches proposed a wearable patient monitoring system, which integrates current personal digital assistant (PDA) technology and wireless local area network (WLAN) technology [4], [5]. A wireless PDA-based monitor is used to continuously acquire the patient's vital signs, including heart rate and SpO<sub>2</sub>. The patient's bio-signals are transmitted in real-time, through the WLAN to a remote central management unit

In another approach a mobile phone was proposed as a client-side part that communicates with the central device by the GSM network [6], [7]. An alert management mechanism has been included in back-end healthcare center to enable various strategies for emergency alerts triggered by automatically recognized situations.

A Bluetooth-enabled in-home patient monitoring system, facilitating early detection of Alzheimer's disease was proposed in [9]. The location and hence the movement of a patient is tracked and reported to a local database, with the use of short-range Bluetooth communications. The collected data is then transmitted via the Internet to a decision engine (on a remote site). "Electronic Doctor's Bag" [8] is another approach to medical system with mobile communication. The main idea of this system is that a nurse instead of a doctor, carries the Electronic Doctor's Bag and visits a patient. Than an equivalent to face-to-face communication between the doctor and the patient is realized remotely.

Data security and optimization of wireless communication between devices of the system [10], [11] are very important aspects of the functioning of the patient in-home monitoring systems.

The purpose of this study is to develop a system of comprehensive and continuous monitoring of the patient at home. The functionality of the various parts of the system was optimized for a user. System that is adapted for continuous measurement of biomedical signals, depending on the patient's disease. System that analyzes the collected information and in the case of detection of dangerous events informs physician and guardian. The purpose of this study is to increase the amount of information that can be acquired from patient at home and exchanging them with doctors.

## II. SYSTEM STRUCTURE

Home monitoring of patients is a wide term and different applications are possible. Some users are highly immobilized others are free to move but suffer dementia. The platform design requires creating a multi-modular system. Proposed by us the Domestic system structure is presented in Fig 1. Our system consists of the following components:

This work was partly supported by European Regional Development Fund concerning the project: UDA-POIG.01.03.01-22-139/09-00 -"Home assistance for elders and disabled – DOMESTIC", Innovative Economy 2007-2013, National Cohesion Strategy.

### A. Users

- An assisted person (AP) – person supported with the system. Communicates with the Domestic system using HCI, various sensors and possibly an Android device.
- A physician - A person in charge of AP health. It should be either doctor or otherwise qualified person. Physician communicates with the system only by tablet.
- A guardian - A person who cares for the patient (eg family member, social service person), communicates with the system using tablet or other Android device.

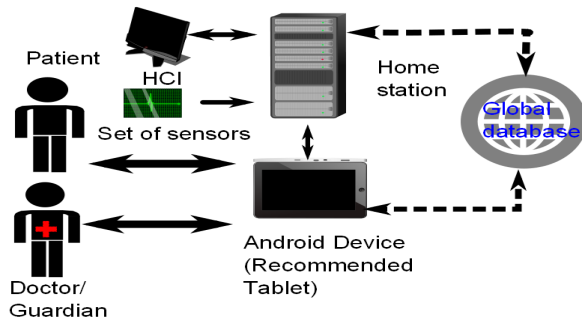


Fig. 1 Domestic system structure (details are given in the text)

### B. HCI

The human-computer interface is one of the most important elements of the platform. The acceptance of the entire system depends highly on the way it can be used by older/immobilized users. Three elementary modes of the HCI are being designed: interface based on touch screen; visually guided interface; audio guided interface.

### C. Home station – central unit

Central computer around which whole system revolves. Home station is a Domestic device consisting of computer with Bluetooth adapter and HCI software. It should manage databases, synchronize with global database, collect and process sensor data, process and dispatch events, integrate data, manage communication, support mental training, etc.

### D. Set of sensors

Set of devices collecting data relevant to patients health state. Different categories of sensors are currently prepared to measure: heart pulse, temperature, body composition parameters, glucose concentration, blood pressure, electric heart activity, and posture activity. Another group of sensors consists of those related to monitor user environment parameters. This is especially important for older people deciding to live independently at home.

### E. Mobile device - tablet

A device with Android operating system, used by users of the system. It allows to communicate with Home station and access to functions such as real-time medical measurements or AP information. Features and details of such device are described in the next paragraph.

## III. MOBILE DEVICE

This work is focused on use of a mobile device for exchange and visualization of data in home monitoring. Portable device used in Domestic system must have Android operating system. This type of device is going to be used by all system users. In the case of the patient and guardian device can be either mobile or tablet, while in the case of a physician use of tablet is recommended (due to the improved readability of medical data presented). Special software was prepared for each user type, which enables, among other things, communication with the central unit and carrying out the appropriate function for the role.

In case of the AP it must fulfill three basic functions: communication with the AP (surveys), monitoring of the AP when not within range of the central unit and notification of some activities. Guardian using a mobile device will be alerted about the dangerous events occurring on the side of the patient and, additionally, when he is in the patient's home will be able to view basic data about the patient's condition (e.g., the last activity of the patient or test results).

Application that was prepared for a doctor is the most expanded. It has been prepared mainly to carry out visits in the patient's home. When the device is within the range of the patient's central station the device will automatically connect to it and update current basic information about the patient. Additionally, there is possibility to browse offline basic medical data, which are stored in the internal database. If the device is equipped with a GSM module, it will be served alerts notifications emitted from the patient central unit as well. During home visits physician has access to personal, electronic healthcare records, which contain the list of actual diseases and prescribed medications, information about recent events and a physician's personal notes about the patient. Additionally, physician can access the archived data or add new data if needed. An example of the patient chart with some test data is presented in Fig. 2. Another feature allows the physician to access the list of measurements which suits the patient's condition. After selecting a particular type of test, information specific to this measurement is presented, both in text and a graphics. The physician has the opportunity to view the entire history of examination, that are automatically downloaded from the central station. There is also a possibility to carry out test directly from the application. Monitoring of patient basic vital signs (ECG, pulse, blood pressure, temperature) which is an example of real time measurement, is shown in Fig. 3.

Interface design for medical applications is the result of consultation with prospective users of the system. It imposed the emphasis on maximizing the clarity of the information presented and intuitive interface. In addition, particular attention was paid to issues of communication and the security of transmitted data between the mobile device and a central unit. These issues are described more specifically in the following paragraphs.

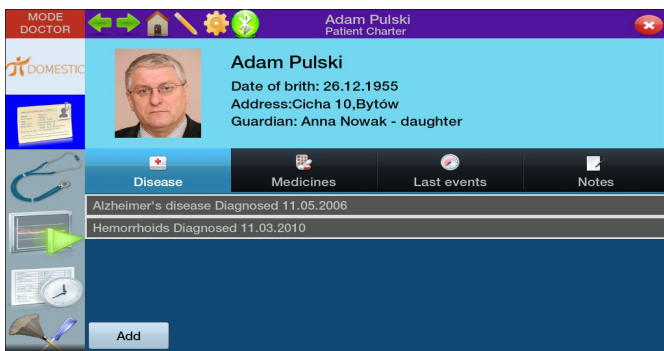


Fig. 2 Patient panel view

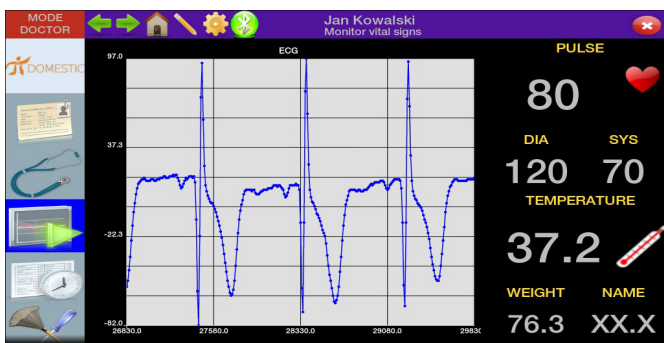


Fig. 3 Real time monitoring of patient vital signs

IV COMMUNICATION

When two of domestic devices get in detection range of each other and at least one of them is a server, communication begins. After connection is established devices start to exchange protocol defined communicates as shown in Fig. 4. At the beginning, message containing information about the sender device type (Initial request), is sent by a server device (e.g. medical tablet). In response, if client device recognized type of server it sends its own type (Initial response), otherwise error message is sent and the connection broken. Server then analyzes information it has been given, to recognize client device. If recognition was negative connection is broken and error message is sent, otherwise communication continues. What happens next depends on what devices are connected. In case of Home Station - Medical Tablet connection conversation will go as follows. Server device requests list of assisted persons associated with connected device (Request list of AP). Client responds by sending *ids* of users stored in its database (Send list of AP). Medical tablet presents received data to the physician and wait for him to take action. During this time, if the entire system is not set to be synchronized externally (e.g. through the internet), devices update each others databases (DB synchronization). Further steps are results of user actions. For each action server device sends a request (Request AP data) to which client responds accordingly (Send AP data).

V SECURITY

Bluetooth offers some security mechanisms to prevent unauthorized access to transmitted data and device functions. Two most important of them are pairing and data encryption.

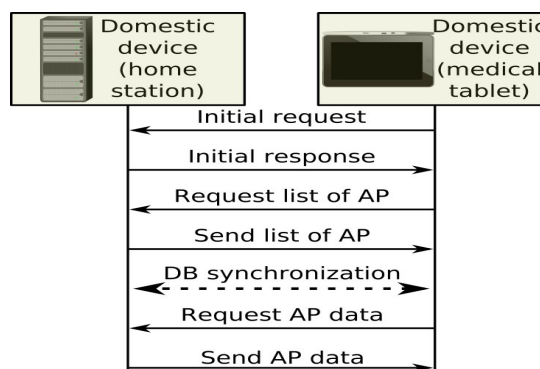


Fig. 4 Data exchange protocol (details are given in the text)

This security measures are not perfect however. To further increase safety of medical data, additional mechanisms were implemented: registering devices; additional data encryption; specially designed system structure. Domestic applications respond only to devices that has been specifically registered to work with them. Registration of device's Bluetooth address is done through applications user interface and not remotely to reduce possibility of registering unauthorized device as much as possible. To prevent data leakage in case of hijacking of Bluetooth connection, domestic applications encrypt all communicates before sending them through Bluetooth adapter (in addition to Bluetooths encryption). Domestic system structure is designed to enhance all security mechanisms mentioned above. Data storing devices such as home stations act as clients for purpose of establishing Bluetooth connections so it is impossible to scan them for services. Part of the data transmitted between domestic devices is in form of unique IDs. To interpret it a device must have access to the Domestic database, otherwise it would only get meaningless strings of characters.

VI DISCUSSION

This paper has presented a comprehensive system of home monitoring of elderly and chronic patients. Previously these types of systems were designed primarily for one group of users. Our system assumes the existence of three types of active members: an assisted person, a guardian and a physician. In the design phase of the system main purpose was to achieve maximum benefit for its individual members. In the case of th AP the goal was to ensure the maximum possible convenience of the system that will, at the same time, provide the highest possible level of control over his condition. For a doctor, it is necessary to provide a solution that will allow the correct interpretation of medical data. We used a tablet with Android system and our software, which is prepared to acquire, process and manage medical data and communicate with other devices, using readable and easy to use graphical user interface. In case of a guardian it was important to obtain basic information about patient condition and alert messages via a GSM network

It is usually necessary to monitor and record multiple physiological parameters of patient with chronic disease as the base data to analyze and track patient's condition and to

provide medical treatment. Currently, the majority of the proposed systems of this type is characterized by large size or limited working area, which reduces the desire to use it and quality of everyday life of the patient. We offer a complete system with modular structure, which allows to choose a set of wireless sensors individually for each patient depending on his condition. It provides integration of multiple physiological parameters extracted from all devices. The main component of the system is the central unit, which communicates directly with other devices, systems and processes and stores all data related to patient. The system has three main strengths: good expandability, highly flexible architecture, simple to design hardware and software.

The software of central unit was developed in Java/JavaFX environment. It offers rich graphical user interface creation combined with all the advantages of Java language. This allows the creation of visually pleasing, cross platform applications with minimal effort. On top of that JavaFX is meant to be able to run on many mobile devices, in browsers and possibly even on TV sets. A trait that might prove useful during development of further Domestic devices.

For Domestic mobile devices Android OS was chosen. This is one of the most popular systems (next to the Apple OS's) used in mobile devices, characterized by the following advantages. Open source platform based on Java, with multi-tasking and wide hardware support. A large variety of available devices, including wide range of tablets with the version of Android 3.0 which is suited to this type of devices. Personal mobile devices with Android system are powerful and flexible in use. It reduces both the time and cost needed for system development.

Both Wi-Fi and Bluetooth provide enough range and transfer rate to fulfill Domestic goals. Advantages Bluetooth has over Wi-Fi are security and ease of use. Bluetooth has two level password protection and getting access to one point of Bluetooth network does not grant access to any other part of it, as opposed to Wi-Fi networks. It is also easier, both software and hardware wise, to establish Bluetooth connection. Another possible choice is ZigBee, however it is inconvenient to send large amounts of data over ZigBee, besides hardly any popular mobile device has built-in ZigBee module, which make it practically useless in out system.

## VII CONCLUSION

The concept of full care service is to prevent interference with patients daily life and still be able to provide long-term

health monitoring services. Achieving this goal is attempted by: flexibility and modularity of the system that makes its presence less noticeable; easy-to-use, intuitive interfaces that are enhancing every day usage of the system; deployment of devices that are well suited for their functions, open and accessible which reduces both development and installation time; usage of modern technologies that allow less devices do more. All of that makes Domestic system as close to provide full care services as close as possible.

## REFERENCES

- [1] B. Rechel, Y. Doyle, E. Grundy, M. McKee, "How can health systems respond to population ageing?" *World Health Organization, Regional Office for Europe*, Copenhagen, 2009.
- [2] WHO, "WHO European Health for All Database", *WHO Regional Office for Europe*, Copenhagen, 2009.
- [3] I. Korhonen, J. Parkka, and M. Van Gils, "Health Monitoring in the Home of the Future," *IEEE Eng. Med. Bio.*, vol. 22, no. 3, May-June 2003, pp. 66-73
- [4] Y.-H. Lin, I.-C. Jan, P. C.-I. Ko, Y.-Y. Chen, J.-M. Wong, and G.-J. Jan, "A wireless PDA-based physiological monitoring system for patient transport," *IEEE Trans. Inf. Technol. Biomed.*, vol. 8, no. 4, December 2004, pp. 439-447.
- [5] U. Anliker, J. A. Ward, P. Lukowicz, G. Tröster, F. Dolveck, M. Baer, F. Keita, E. B. Schenker, F. Catarsi, L. Coluccini, A. Belardinelli, D. Shklarski, M. Alon, E. Hirt, R. Schmid, and M. Vuskovic, "AMON: a wearable multiparameter medical monitoring and alert system," *IEEE Trans. Inf. Technol. Biomed.*, vol. 8, no. 4, December 2004, pp. 415-427.
- [6] R.-G. Lee, K.-C. Chen, C.-C. Hsiao, C.-L. Tseng, "A Mobile Care System With Alert Mechanism", *IEEE Trans. Inf. Technol. Biomed.*, vol. 11, no. 5, September 2007, pp. 507-517.
- [7] Y. Kogure, H. Matsuoka, Y. Kinouchi, M. Akutagawa, "The Development of a Remote Patient Monitoring System using Java-enabled MobilePhones", *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference Shanghai, China*, September 1-4, 2005.
- [8] M. Yoshizawa, T. Yambe, S. Konno, Y. Saijo, N. Sugita, T. K. Sugai, M. Abe, T. Sonobe, Y. Katahira, S. Nitta, "A Mobile Communications System for Home-Visit Medical Services: The Electronic Doctor's Bag", *Proceedings of 32nd Annual International Conference of the IEEE EMBS Buenos Aires*, pp. 5496-5499, Argentina, September 2010, pp. 5496 - 5499.
- [9] H.-T. Cheng, W. Zhuang, "Bluetooth-Enabled In-Home Patient Monitoring System: Early Detection Of Alzheimer's Disease", *IEEE Wireless Communications*, February 2010, pp. 64-69.
- [10] L. Jatoba, U. Grossmann, J. Ottenbacher, W. Stork, K. Muller-Glaser, "Development of a Self-Constructing Neuro-Fuzzy Inference System for Online Classification of Physical Movements", in *9th International Conference on e-Health Networking, Application and Services*, 2007, pp. 332-335.
- [11] Tadj Chakib, Hina Manolo Dulva, Ramdane-Cherif Amar, Ngantchaha Ghislain "The LATIS Pervasive Patient Subsystem: Towards a Pervasive Healthcare System", in *ISCIT '06. International Symposium on Communications and Information Technologies*, 2006, pp. 851-856



## MuSA: a multisensor wearable device for AAL

Valentina Bianchi  
Centro TAU, Università di Parma  
Email: valentina.bianchi@unipr.it

Ferdinando Grossi  
Centro TAU, Università di Parma  
Email: ferdinando.grossi@unipr.it

Ilaria De Munari  
Centro TAU, Università di Parma  
Email: ilaria.demunari@unipr.it

Paolo Ciampolini  
Centro TAU, Università di Parma  
Email: paolo.ciampolini@unipr.it

**Abstract**—In this paper, a novel multi-sensor wearable device, called MuSA, is introduced. MuSA aims at integrating in the CARDEA ambient-assisted-living framework: on the one hand, MuSA provides CARDEA with useful ambient-intelligence features, such as localization and identification; on the other hand, it may borrow from the environmental control system many infrastructural and communication components, resulting in a less expensive implementation. MuSA exploits on-board sensors and signal processing units for fall detection, heartbeat and breathing rates detection. At this level too, sharing of part of the circuitry enables power and cost savings. Ubiquitous computing paradigm is followed, carrying out all of the signal processing and decision processes at the wearable node: this makes communication toward supervision levels much less demanding and independent on the actual physical features of the sensors themselves. Test have been carried out, confirming that the low-cost approach which has been followed still allows for adequate quality of responses. Field test is starting, to evaluate psychological and ergonomic aspect as well.

### INTRODUCTION

POPULATION ageing is putting to severe proof current health- and social-care models: the relative number of people experiencing frailty and disability conditions due to old age is increasing, so that conventional caregiving schemes are becoming hardly sustainable. In particular, institutionalization is frequently exploited to provide care to lone elderly. This practice, however, implies high costs (besides potentially threatening quality of life), and cannot be easily scaled to the increasing number of older people needing assistance: hence, home care and home assisted living strategies are an essential component of present and future care policies. Tools based on information and communication technologies may play an enabling role, allowing for the implementation of many assistive functions in the home environment, aimed at autonomy and independent life. To this purpose, basic needs to be accounted for are related to ambient safety and security, as well as personal and health monitoring. In this paper, an integrated approach to such issues is described. The CARDEA system [1] encompasses within the same framework many functions which are customarily carried out by independent entities: CARDEA relies on standard IP

communication techniques (even at the field level) and is inherently based on distributed intelligence techniques. On-board processing is extensively exploited by ambient sensors and wearable sensors, aiming at reducing communication overheads, increasing reliability and reducing costs. An open and flexible architecture is implemented, allowing for reconfigurability and interoperability. Remote access and control of every device is enabled through web-based tools.

In the following section, the CARDEA framework is introduced, emphasizing the adoption of distributed-intelligence ambient control modules. Then, MuSA wearable wireless sensor platform is described. Conclusions and ongoing work are eventually discussed in last section.

### CARDEA

CARDEA[1] is a powerful and versatile Ambient Assisted Living (AAL) system, developed at University of Parma.



Fig. 1 CARDEA system view

The system is inherently based on standard IP communication, and has a hierarchical structure, easily scalable from the single apartment to large residential complexes. An intelligent module, called FEIM (Field Ethernet Interface Module) has been designed and

implemented to cope with a wide variety of sensor devices: FEIM module allows for connecting low-cost sensors, even if not conceived for network connectivity.

FEIM is based on an embedded microcomputer, and can be programmed to control analog and digital multiple field devices (up to 25 per module). Local processing power is also exploited to implement low-level decision strategies (light or appliance control, for instance) or safety-critical tasks (alarm messaging) making them independent on actual availability of network connection. FEIMs communicate among them on a peer-to-peer basis, requiring no external supervision, allowing for establishing operating rules involving different device clusters.

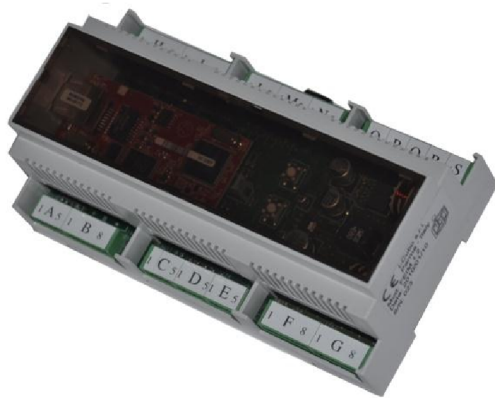


Fig. 2 FEIM module

Also, dynamic reconfiguration is exploited to implement fault-tolerant policies and graceful degradation strategies, by accounting for redundant configurations and module hot-swapping. At a higher hierarchical level, supervisor processes, running on any networked machine, are exploited for the implementation of more complex operating rules and to deal with user's interfaces and external network gateways. CARDEA includes a number of user interfaces, easily accessible and suitable for elderly people and people with disabilities [2]. It also features a web-extension module, which enables full control and monitoring functions from any remote location, by means of dynamic web applications [3].

A wireless sensor network (WSN) contribute to the system architecture as well, dealing with mobile devices. Here, we shall refer to a WSN based on the IEEE-082.15.4 (ZigBee) standard protocol, exploited for the implementation of wearable sensors described in the following.

#### CARDEA-MUSA

Wearable devices are exploited to monitor personal activities and vital signs. They are conceived to provide continuous monitoring, so they need to be small and lightweight, and minimally intrusive. Also, battery power should last as long as possible, this calling for accurate management of the

power budget. Finally, being such devices exploited for security and health purposes, high reliability, as well as ease of use and reconfigurability [4], are mandatory. Of course, inexpensiveness is needed to allow for large-scale deployment.

CARDEA MuSA (MULTi Sensor Assistant) is a wearable multisensor platform, specifically designed with assistive purposes, compliant with ZigBee/IEEE802.15.4 standard protocol. MuSA is designed to be worn at belt or at chest: it is quite small (78x48x20 mm), and lightweight (about 70 grams, Li-Ion battery included). Different functions can be implemented on the same platform: basic configuration of MuSA includes a call button, automatic fall detection and an indoor localization function. The latter function can be exploited in large residential complexes, allowing CARDEA to make caregivers aware of the actual position of a person needing assistance, or to detect wandering behaviors of cognitive-impaired people [5]. CARDEA MuSA can be extended with further functions, hosted by the same hardware platform: a basic ECG system is implemented, used to evaluate heart rate. A breath rate detector is included as well, based on chest expansion measurement. All of the signal acquisition and processing is carried out by MuSA on-board circuitry: detection of abnormal behaviors or deviation of vital signs from their "normal" range is carried out by MuSA. Radio communication is hence kept at a bare minimum (alarm messages and network management), saving battery energy.

MuSA board is based on CC2531 system-on-chip [6] by Texas Instruments. Two basic building blocks can be identified: a IEEE 802.15.4 radio transceiver, and a microcontroller taking care of ZigBee stack management. The same microcontroller is exploited for digital signal processing. The board also include sensors and analog front-end circuitry needed to acquire vital signs.



Fig. 3 MuSA wearable device

MuSA is fully integrated in the CARDEA framework: a network of ZigBee fixed-position nodes is deployed into the environment, and managed by CARDEA. Such nodes exchange information with MuSA mobile devices, and make them available to CARDEA, either by exploiting a FEIM interface channel (i.e., similarly to environmental sensors) or directly at the supervision level, communicating with supervising processes through a TCP/IP socket. Then, all of

the communication features of CARDEA (web-based remote management, phone or SMS messaging, etc.) are straightforwardly available to MuSA, with no need of replicating such features in a stand-alone MuSA base-station.

#### A. Fall-Detection

The fear of falling is a major issue, threatening elderly self-confidence and independence. Falls are one of the first causes of death or serious injury in older adults [7]: the use of automatic fall detection systems could both speed up the assistance of the people in need and give a security feeling to the person using it. Perspectively, behavioral analysis of people's motion (gait quality, for instance, based on the same accelerometric patterns exploited for fall detection) can be exploited for possibly preventing fall conditions.

There are several ways to fall: the fall movement can be quite different, depending heavily on the actual situation. In [8] a fall classification attempt is presented; the author identifies three different kinds of most common falls for an older adult: fall during sleep, from the seated position and from standing up to lying on the floor. Whereas the first two can be somehow monitored by means of bed- or chair-occupancy sensors managed by CARDEA, the last one (also being the most frequent kind) calls for smarter automatic detectors [9]. Fall detectors may exploit artificial vision algorithm, environmental sensors and wearable sensors. Wearable sensors provides a fair trade-off among cost, performance and intrusiveness: the adoption of a wearable device, moreover, also provides CARDEA with identity information, enabling management of personalized settings and behaviors.

At the heart of fall detector, the low-power LIS331DLH [10] triaxial MEMS accelerometer manufactured by ST Microelectronics is used for algorithm implementation.

Basic fall detection algorithms exploit threshold comparison [11]: since falls are often associated to acceleration peaks, current acceleration (the Euclidean norm of the acceleration vector, actually) is checked against a given threshold, which depends on personal physical features (height, weight, ...). However, tuning such a threshold is critical, with respect to false-positive and false-negative conditions: many daily-living activities may result indeed in accelerations comparable with those involved in a fall (stumbling, sitting down or standing up, bending down, ...). Hence, a more reliable detection strategy is needed: to this purpose, MuSA correlates acceleration pattern with postural information. Body orientation can be easily inferred by exploiting further sensors such as gyros or protractors, introducing however additional costs, size and power constraints. We therefore exploited digital signal processing to extract relative orientation information from the same stream of accelerometric data. The MEMS accelerometer, in fact, is subject to the gravity acceleration  $G$ , which can be regarded as a "static" component of the sensed acceleration. Static acceleration can be extracted from the sensor output stream by proper filtering, thus providing a reference that

can be used to calculate the angle shift between subsequent estimations [20].

A differential approach is followed, which makes the algorithm independent on the actual sensor-wearing fashion. Whenever an acceleration peak exceeding the threshold is observed, a comparison between the orientation before and after the peak occurrence is carried out.

The algorithm is illustrated in Fig. 4: acceleration components ( $A_x, A_y, A_z$ ) are acquired from the MEMS every 16 ms. Then, the acceleration norm is computed, and acceleration components are stored for subsequent processing. When the norm exceeds the given threshold, comparison of static accelerations starts, looking for the tilt angle computed just before and after the acceleration peak.

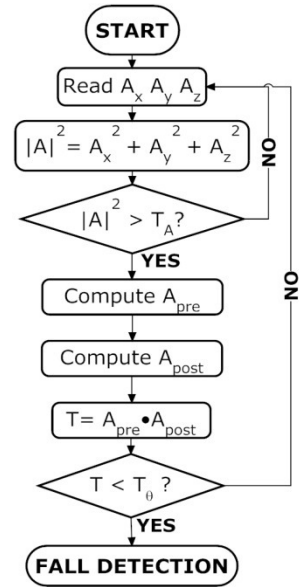


Fig. 4 Fall detection algorithm flowchart

$A_{pre}$  is static acceleration (i.e., steady state) component, averaged on a 1 sec time interval prior to the acceleration trigger, while  $A_{post}$  refers to the same average component, computed after the trigger. Averaging allows for noise reduction, and the dot product between pre- and post-peak average accelerations can be computed:

$$T = A_{pre} \cdot A_{post} = \sum_i A_{pre_i} A_{post_i} \quad (1)$$

Since the intensity of  $A$  in both cases should be equal to  $G$ , the tilt angle  $\theta$  can be readily worked out:

$$T = |A_{pre}| |A_{post}| \cos \theta = G^2 \cos \theta \quad (2)$$

and compared with a suitable threshold, in order to infer an actual fall.

So doing, data coming from a single accelerometer provide both acceleration and (relative) tilt-angle estimate.

The algorithm reliability has been verified through some tests, following the methodology introduced in [12]. We tested MuSA on a set of twelve volunteers (20-30 years old, 51-78 kg weight, 163-192 cm height), who were asked to

simulate different kind of falls from the standing position (backwards to sitting position, backwards to lying position, forward to knees, laterally to right side, laterally to the left side). Then, the volunteers were asked to perform daily living activities, suitable for being mistaken for falls (recovering standing position from previously described falls, sit down on a chair, stand up from a chair, sit down on a stool, stand up from a stool). Every test was repeated five times by each volunteer, summing up to 840 records in the database.

By considering the number of True Positives (TPs) and the number of False Negatives (FNs) we may evaluate the sensor sensitivity, i.e., the ability of recognizing actual falls:

$$\text{Sensitivity} = \frac{TPs}{TPs+FNs} \quad (3)$$

For the given test set, a 99 % sensitivity figure was achieved. Similarly, taking True Negatives (TNs) and False Positives (FPs), we may evaluate sensor effectiveness in discriminating misleading events:

$$\text{Specificity} = \frac{TNs}{TNs+FPs} \quad (4)$$

A 97.8 % specificity figure was estimated on the given set.

The algorithm, although relying on limited computational power available, is hence quite accurate; data processing and interpretation is carried out by MuSA on-board processor, which hence can be seen by CARDEA as a simple binary sensor, signaling falls by means of a Boolean variable.

A low-power operating mode has also been implemented, exploiting LIS331DLH features to reduce MEMS sampling frequencies. In fact, acceleration threshold and orientation can be monitored by the MEMS device itself, relieving the microprocessor from continuously checking the data stream. The CPU is awoken by the accelerometer (through an interrupt line) whenever an over-threshold acceleration is detected. In normal conditions, CPU activity is hence limited to coarse sampling of accelerometer registers (acceleration and orientation). This allows for better exploitation of sleep modes, at the expense of a less detailed recording of motion data: by tuning time intervals, we were able to attain a 50% reduction in the processor power consumption, with a negligible degradation in sensitivity and specificity.

### B. Heartbeat Detection

CARDEA MuSA is also capable of estimating the heartbeat rate. This can be exploited to promptly notify abnormal heart rhythms (i.e., arrhythmias, tachycardia or bradycardia), or, when combined with motion data, to provide a more accurate picture of the energy expenditure. A simple electrocardiogram (ECG) section is exploited to this purpose. Diagnostic systems usually rely on a variable number of body electrodes, placed at specific configuration patterns (leads) [13]: of course, ECG definition and accuracy increases with the number of electrodes and leads acquired,

providing more physiological insight. MuSA, however, aims at continuous monitoring and is not conceived as a diagnostic tool: then, in order to limit sensor intrusiveness and to cope with circuit size and power consumption, we adopted a simple, two-electrode scheme [14], which enables, depending on the actual body placement of electrodes, exploitation of three fundamental Einthoven leads. On-board circuitry include an analog front-end, consisting of an instrumentation amplifier and a low-pass analog filter (106 Hz cutoff frequency). Digital processing is then carried out by the CC2531 CPU: just after first A/D conversion, digital filtering of mains (50 Hz) frequency noise is carried out. Then the input signal is numerically derived, in order to emphasize Q-wave peaks, the frequency of which is subsequently identified by means of a threshold comparator.

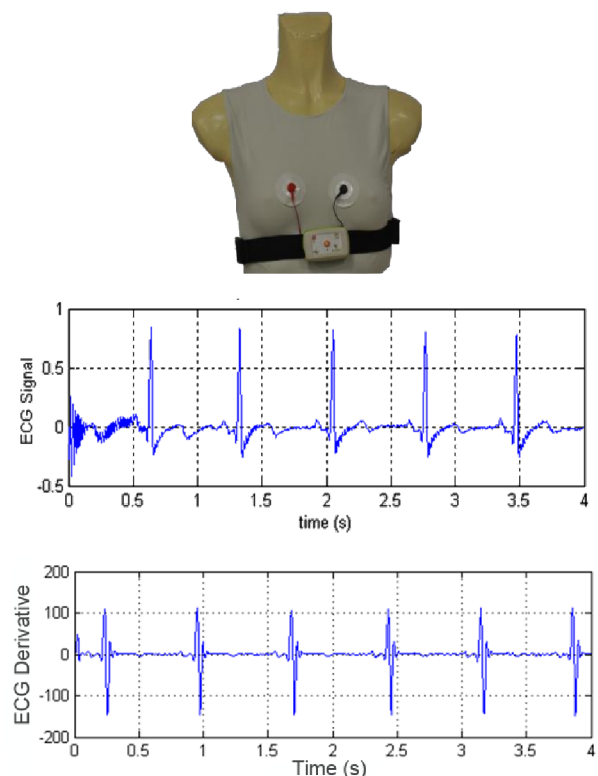


Fig. 5 MuSA embedded ECG subsystem

In order to deal with individual diversity, the algorithm thresholds are automatically calibrated, on the basis of the signal acquired in the early monitoring stages.

TABLE I.  
HEARTBEAT RATE ESTIMATION

	Average relative error
Einthoven I lead	4,1955%
Einthoven II lead	2,8221%
Einthoven III lead	2,0008%



The estimated heartbeat rate is then checked against (user-defined) normal range boundaries: should such boundaries be exceeded, an alarm is issued to CARDEA, which, in turn, activates messaging strategies aimed at relatives, medical doctors, neighbors, caregivers, etc., according to the current profile. Since the estimate on moving subject can be quite noisy, the alarm is issued on average estimation. Such an approach is compatible with available computing resources and accurate enough for the given purpose: by computing heartbeat frequency on-board, no ECG tracing needs to be transmitted on the radio-link, thus greatly reducing radio power consumption.

Tests have been conducted on a set of volunteers, under different activity conditions, and exploiting different Einthoven leads: estimated frequencies were then compared with those extracted from a reference instrument. Results are summarized in Table I above, and demonstrate the achievement of fairly reliable detection in all cases.

### C. Breathing Rate Detection

Estimation of breathing rate exploits a piezoelectric sensor (Measurement Specialties LDT0-028K) inserted into the elastic chest strap. The sensor detects variations in the strap strain due to inhaling and exhaling movements, and produce a charge variation at the piezo-polymeric film. The signal is hence acquired through a charge preamplifier, having a narrow, low-frequency bandwidth (0.007 Hz - 16 Hz), so to filter out components due to movements different from breathing.

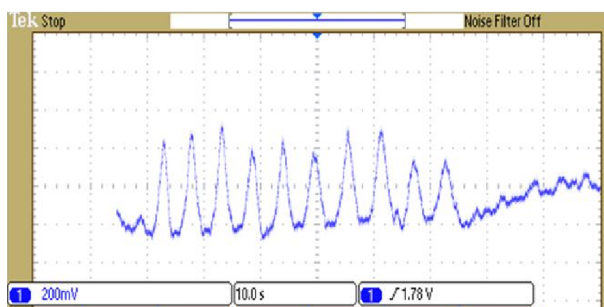


Fig. 6 MuSA embedded breathing-rate estimation subsystem

Similarly to the ECG section, after a further analog amplification stage, the signal (shown in Fig. 6) is acquired by the TI-CC2531 chip, which looks for amplitude peaks. In this case too, the time-domain sensor signal (an example of which is shown in Fig. 4) does not need to be transmitted to CARDEA: MuSA compares breathing frequencies worked out with the assigned “normal” range, and issues anomaly warnings when needed. By exploiting CARDEA cooperation, careful partition of tasks between the wearable device and the environmental infrastructure was made possible, allowing for effective management of the available power and computational budgets. At the base station side, most functions (alarm dispatching and notifications) can be borrowed from environmental control modules, thus resulting in an inexpensive and versatile device. Moreover,

data coming from MuSA can be correlated with environmental information sources, to increase, through data fusion, the overall reliability of the monitoring system. Finally, sharing the same CARDEA information space, makes MuSA outcomes readily available on the web, thanks to the CARDEA<sub>web</sub> extension.

### CONCLUSIONS AND ONGOING WORK

The concept of multi-sensor, multi-functional wearable platform is being currently expanded, accounting for further functionalities: in particular, MuSA prototypes are being developed including body temperature and microphonic sensors. Body temperature is acquired through a low-cost NTC thermistor, embedded in the chest strap as well. Analog signal treatment is carried out through an instrumentation amplifier (TI-INA330, [6]), whereas digital processing is carried out, as usual, by the CC2531 chip. First tests show that such a low-cost approach still allows for accuracy in the 0.1 °C, which is adequate for long-term monitoring purposes.

Including a microphone will also allow MuSA user to communicate verbally with remote caregivers, when seeking for assistance or in case of fall. Integration with CARDEA avoids the need of accounting for a bi-directional voice-channel: incoming voice message can be managed by the environmental control system, thus not requiring to embed speakers or audio amplifiers into MuSA. A tiny MEMS microphone (Analog Devices ADMP401- 4-5) easily fits the MuSA board. Sampling and digital encoding of the audio stream is carried out by the digital processor, which subsequently send it over the radio link.

CARDEA-MuSA is currently undertaking a field-test campaign, being deployed at some assisted-living facilities already running long-term CARDEA trials [3]. Besides technical performance and reliability, assessed by lab test, this will allow to check its ergonomic and psychological impact on elderly users and on their caregivers, allowing for optimizing the service and to evaluate potential benefit of the adoption of wearable multifunctional devices in actual care policies.

### REFERENCES

- [1] F. Grossi, V. Bianchi, G. Matrella, I. De Munari, P. Ciampolini, “An Assistive Home Automation and Monitoring System”, *ICCE 2008 Digest of Technical Papers*, (2008), pp 341-342.
- [2] V. Bianchi, F. Grossi, I. De Munari, P. Ciampolini, “Multi-modal interaction in AAL systems”, *AAATE2011 Conference Proc.*, IOS Press, Assistive Technology Series, (2011), .
- [3] A. Losardo, F. Grossi, G. Matrella, V. Bianchi, I. De Munari, P. Ciampolini, “Web-enabled Home Assistive Tools”, *AAATE2011 Conference Proc.*, IOS Press, Assistive Technology Series, (2011), .
- [4] T. Martin, E. Jovanov, and D. Raskovic, “Issues in Wearable Computing for Medical Monitoring Applications: A Case Study of a Wearable ECG Monitoring Device”, in *Proc. International Symposium on Wearable Computers*, (2000), pp. 43–50.
- [5] V. Faucounau, M. Riguet, G. Orvoen, A. Lacombe, V. Rialle, J. Extra, A.-S. Rigaud, “Electronic tracking system and wandering in Alzheimer's disease: A case study”, *Annals of Physical and Rehabilitation Medicine*, vol. 52, iss. 7-8, Sep-Oct 2009, pp. 579-587.
- [6] Texas Instruments web site: [www.ti.com](http://www.ti.com)

- [7] M. E. Tinetti, M. Speechley, and S.F. Ginter, "Risk factors for falls among elderly persons living in the community". *The New England journal of medicine*, vol 319, iss. 26, Dec 1988, pp. 1701–1707.
- [8] Xinguo Yu, "Approaches and principles of fall detection for elderly and patient", in *Proc. IEEE International Conference on e-Health Networking, Application and Service*, (2008), pp 42–47.
- [9] S. R. Lord, J.A. Ward, P. Williams, and K.J. Anstey. "An epidemiological study of falls in older community-dwelling women: the randwick falls and fractures study". *Aust J Public Health*, vol. 17 iss 3, 1993, pp. 240–245.
- [10] ST Microelectronics web site: [www.st.com](http://www.st.com)
- [11] J. Chen, K. Kwong, D. Chang, J. Luk, and R. Bajcsy.—Wearable Sensors for Reliable Fall Detection, in *proc. 27th IEEE Engineering*
- [12] *Medicine and Biology Society Annual International Conference*, (2005), pp. 3551–3554.
- [13] A. K. Bourke, J. V. O'S'Brien, and G. M. Lyons, "Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm". *The New England journal of medicine*, vol 26, 2007, pp. 194–199.
- [14] John E. Madias, "The 13th multiuse ECG lead: Shouldn't we use it more often, and on the same hard copy or computer screen, as the other 12 leads?" , *Journal of Electrocardiology*, Vol. 37, Iss. 4, Oct. 2004, pp. 285-287.
- [15] Thakor, Nitish V., Webster, John G., "Ground-Free ECG Recording with Two Electrodes", *IEEE Transactions on Biomedical Engineering*, Vol BME-27 Issue:12, Dec 1988, pag. 699 – 704.

## An intelligent bathroom

Adam Bujnowski, Arkadiusz Palinski, Jerzy Wtorek

Gdansk University of Technology, Faculty of Electronics, Telecommunications and Informatics,  
ul. Narutowicza 11/12 80-233 Gdansk

Email: bujnows@biomed.eti.pg.gda.pl, arekpalinski@gmail.com, jaolel@biomed.eti.pg.gda.pl,

**Abstract**—Monitoring system for detection of person and his/her activity in the bathroom is described in the paper. It also detects and monitors person taking bath. The system consists of sensors measuring humidity, air and water temperature, spilled water, and state of bathtub. The bathtub state detector (BSD) allows controlling water level and its temperature. It also monitors of person’s activity when being in bathtub. It is an essential part of the system. The BSD distinguishes between four cases: 1. the bathtub filled only with water, 2. the bathtub filled with water and occupied by person, 3. the bathtub filled with water and occupied by active (moving) person, and 4. empty bathtub. Presence and activity of the person is recognized by means of multifrequency impedance measurements.

**Keywords:** Bath monitoring, intelligent sensors, bio-impedance

### I. INTRODUCTION

HYGIENE is an important factor that may affects a health status of the human. Regular usage of the bath plays significant role in maintaining good health state and in preserving conditions of life. However, utilization of the bathtub may involve dangerous situations especially for persons living alone. Thus, it is important to monitor a process of the bathroom utilization in the case of the elders.

Tasks realized by such a system may be divided in two categories. First, is to provide a comfortable conditions, e.g. temperature of air and water used while another one is to recognize dangerous situations, including risk of sudden death, especially when temperature of the bath water is too high [6]. Thus, in the study we concentrated on two basic problems. One is a control system of the bathroom environment such as temperature and humidity of the air, water flames on the floor and the water temperature and its level when the bathtub was filling and/or filled. Another problem was to develop a monitoring system for person having a bath. It could be achieved by measuring signals generated by a human body, e.g. electrocardiogram (ECG) or generating and measuring signals which parameters would be affected by the human body. Moreover, the developed system had to be relatively cheap and had to preserve privacy demands. Additionally, the device had to be “invisible” (did not demand any control or adjust) for the person being monitored. Electrical impedance measurements has fulfilled above demands. In order to improve quality of the data a four-electrode technique was examined both theoretically and experimentally.

### II. METHODS

#### A. The system structure

A global structure of the system is based on a star topology. Thus, it is a sensor-actuator network. A central element of the network, called the central system, is a kind of computer equipped with communication interfaces. It receives all the information from the sensors and so called auxiliary systems. It can interact either with the supervised person, or installed actuators (valves, switches etc.) according to the information extracted from the data collected. The central system communicates with the external world by means of alerts broadcasted to relatives, caregivers, supervisors or physicians using different media (e.g. GSM, Internet, etc.). A simplified structure of the system is shown in Fig. 1.

The internal structure, decision rules and performance of the system must be adaptable to different demands. That is why it has a modular structure allowing its reconfiguration. The term “dedicated systems” in Fig. 1 stands for devices performing complete signal processing and taking decision. Thus, it contains an autonomic computation/processing unit. It is also equipped with own set of sensors and actuators.

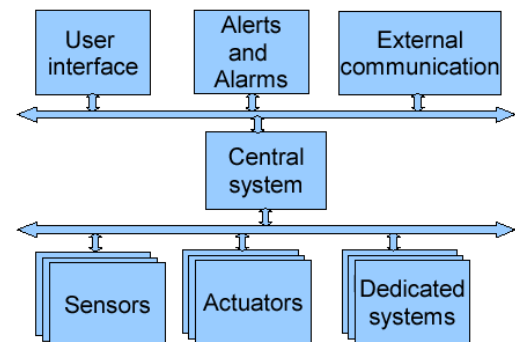


Fig 1: A simplified structure of the monitoring system

The dedicated systems communicate with the central one. These systems govern all resources and also collect data from the sensors directly connected to each and control the actuators.

#### B. Dedicated systems

The dedicated systems controlling events in the bathroom (Fig. 2) are described in the following paragraphs of section. One of them is devoted to control air conditions in the bathroom. Another one is dedicated to detect a presence of the water on a floor. Next one allows controlling of temperature

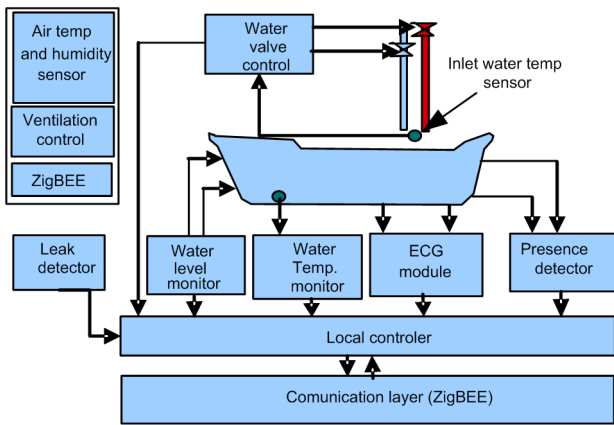


Fig. 2. A block diagram of the bathroom dedicated systems (devices)

and level of the water in a bathtub. And the last one presented in the paper enables detection of human body presence in the bathtub. Together, all of them, together with a central system, form the system for monitoring events and conditions of the bathroom when it is being utilized.

#### i. Air conditions device

The bathroom is a space with specific air conditions. Usually, it is the warmest chamber in a house or in a flat. Moreover, because of using hot and evaporating water, a humidity of the air is higher than in other chambers.

High humidity and temperature of the air, aside being uncomfortable for elders, may create perfect conditions for growing and developing of moulds which in turn, also may be dangerous for the health. Thus this space requires special care in order to maintain the appropriate conditions. Mechanical ventilation is a typical method to keep a proper humidity level. However, it must be controlled as excessive ventilation may lead to significant decrease of temperature in the bathroom. Thus, temperature controller and mechanical ventilation system to maintain low level of the humidity has been designed. It is built around the microcontroller PIC24F equipped with ZigBee module, temperature and humidity sensors (Fig. 3). Basing on measured data appropriate procedure controls the fan performance. It also allows reporting value of humidity and temperature of the air to the central system via ZigBee. Basing on their values adequate efficiency of the ventilation system is adjusted.

In order to achieve compatibility of the ventilation controller with the central system and other devices it has been equipped with ZigBee and the communication protocol [6]. At the moment, the bathroom air-conditions are controlled by means of mechanical ventilation. However, it can be extended to a more sophisticated system by including other actuators (e.g. heating device) or sensors.

#### ii. Water level and temperature monitor

Water filling control module consists of electrically controlled valves for cool and warm water with water temperature measurement and two-level detector (Fig. 4). The valves are controlled by means of the microcontroller (PIC24F16) equipped with sensors. It enables analogue

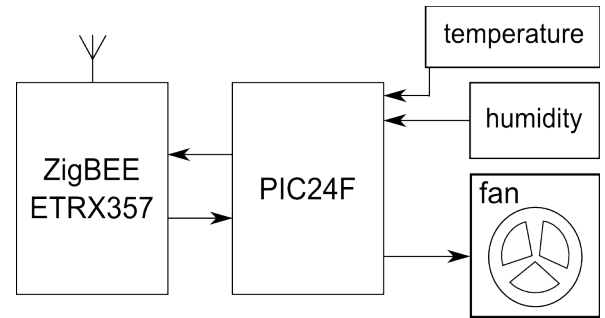


Fig 3: A block diagram of bathroom conditions controller

temperature measurement and water level detection. Water level is determined by measuring electrical impedance between electrodes. A reference electrode is placed close to the sink-hole and two other electrodes are located at two places indicating a low level and a high level of the water. Output data are transferred to microcontroller and by means of ZigBee module are available for the local controller or the central system.

Similar, the temperature of the water in the bathtub can be measured during process of taking bath. If the temperature will be too low an alert suggesting end of the bath can be raised.

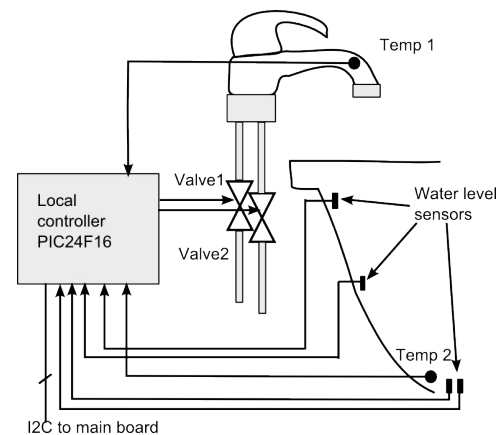


Fig. 4. A block diagram of system controlling the process of bathtub filling with water by measuring its level and temperature

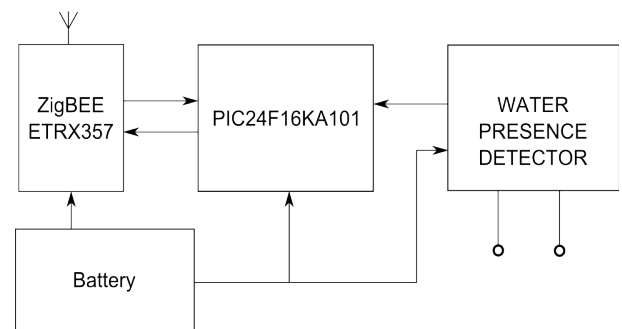


Fig. 5. A block diagram of the leak detector

It is also possible to stop the filling process (using actuators controlled by this system). As the microcontroller is connected to the ZigBee communication module the central system may also control (supervise) temperature of the water in the bathtub. Moreover, it controls the process of



filling the bathtub with water basing on information obtained from the system detecting presence of person in the bathtub. It prevents the water from overflowing the bathtub.

### iii. Water leak detector

Next device already examined is a water leak-detector (Fig. 5). It is a portable and battery-operated device and it detects water presence on the floor. As it is a portable device it may be also utilized in other chambers exposed to water, e.g. kitchen. It consists of a microcontroller (PIC24F16KA101), water presence detector, ZigBee communication module and battery power supply. We have utilized low power microcontroller using advanced sleep modes to maintain long live time of the battery. Thus, we can use single lithium battery (Minmax ER14505M) for a very long period (at least one year).

### iv. Monitoring system of a human body presence in the bathtub

Human body consists of electrically dispersive materials. I.e. its electrical properties, e.g. complex permittivity, depend on frequency of electrical field. The dispersion of electrical permittivity occurs in a few frequency ranges named respectively  $\alpha$ ,  $\beta$ , and  $\gamma$  dispersion. The  $\alpha$ -dispersion appears at relatively low frequency range. On the other hand, water exhibits invariable properties in this frequency range. This suggests a simple method of human presence detection based on electrical impedance measurements. However, there were some problems to be solved before utilizing this technique. Determination of an optimal electrodes position, providing adequate sensitivity of the method independently on body position and localization in the bathtub, was one of them. A Finite Element Method was used for performing simulations. It allowed selecting adequate shape, size and localization of electrodes in the bathtub. An example of 2D model is presented in Fig. 6. As a result a four-electrode technique was selected for experiments. Thus, also the developed measuring system utilized a four electrode technique (Fig. 7). It was built around the AD5933 integrated circuit devoted to measuring impedance.

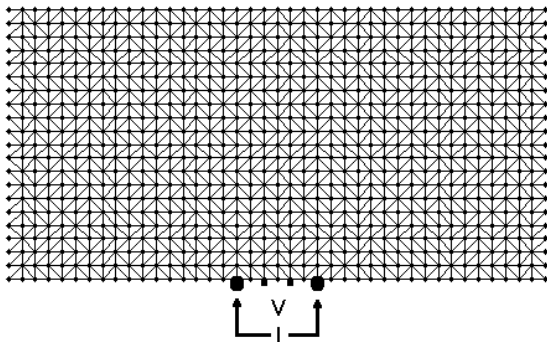


Fig. 6. Sensitivity plot for the selected geometry

However, originally this integrated circuit utilized a two electrode technique. The two-electrode technique is very susceptible to electrode impedance arising from polarization phenomenon. Thus it was necessary to equip the AD5933 with additional circuits (converter V/I, instrumentation amplifier and high-pass filters see Fig. 7) what allowing its utilization in four-electrode technique. The circuit performing

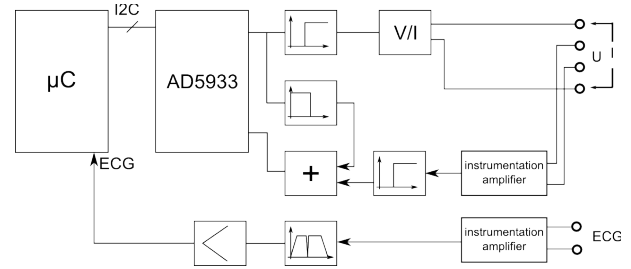


Fig 7. Block diagram of patient detection module enhanced with ECG measurement unit

measurements of electrical impedance was enhanced with unit devoted to electrocardiographic signal measurements. Since the person taking the bath is immersed in the water almost all the time it is possible to measure ECG signals of the person. In typical application signal is processed and only HR parameter is reported to the central system. However, in case of arrhythmia or other situations recognized as dangerous, collected ECG waveforms can be transferred for further analysis. We decided to attach the measuring electrodes at the both sides of the bathtub while the reference one was put at the bottom of the bathtub. They were combined with a bath-tub carpet. It simplified system service and allowed removing of the carpet for cleaning or exchange. In such way almost any bathtub may be adopted for such measurements. At first stage of the study a separate set of electrodes was used for both types of measurements, i.e. electrocardiographic and impedance. However, it was assumed that the final solution would utilize the same set of electrodes for measuring simultaneously both signals. Up to now, we have achieved results from separate electrode matrices. To perform experimental tests we have used a FriendlyARM mini2440 embedded board.

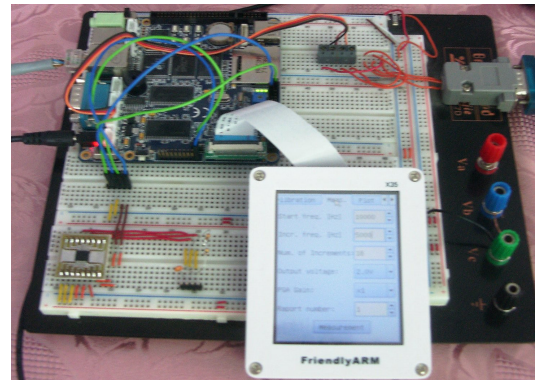


Fig 8. Prototype of the impedance measuring circuit with the host embedded computer

It consisted of main board with CPU S3C2440A, ARM920T (Samsung) with 64MB of 32-bit SRAM, Solid State 256MB NAND Flash hard disk. Additionally board was equipped with 640x480px graphic display and resistive touch-screen. The board was working under GNU/Linux operating system. Therefore using of IIC bus was native. We utilized a custom board with the AD5933 circuit (Analog Devices) and the developed analogue front-end (See Fig. 8). An application for communication to the AD5933 impedance

analyzer has been created. Additionally a graphic front-end using the QT libraries has been created in order to communicate with the application and to present the results.

A reliability of ECG measurement would be affected by many factors. At first we needed to know if the person is in the water. As it was already mentioned, detection of the person presence in the bathtub was done by the electrical impedance measurements. The impedance spectroscopy measurements were performed for a set of selected frequencies (200 Hz – 100 kHz).

### III. RESULTS

#### A. Theoretical model

The Finite Element model of the bath and electrodes was simulated using Matlab package. At first we have calculated spatial sensitivity distribution. Values of the sensitivity shown in Fig. 9 are both positive and negative. Thus, it is necessary to answer if, and how it will affect the

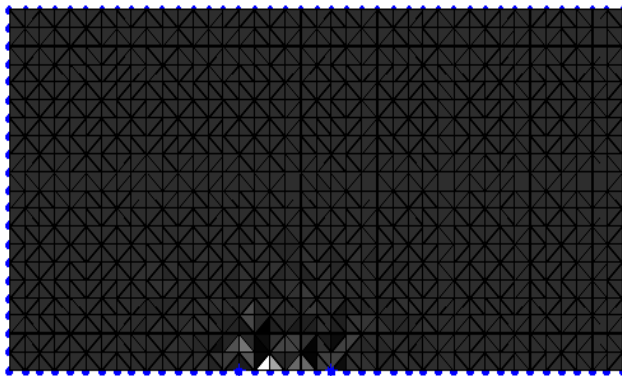


Fig. 9. Sensitivity plot for the selected geometry

measurements. A different localization of human body in respect to position of electrode matrix has been simulated. A dispersive property of human body has been assumed. In general, the model of the dispersive object immersed in non-dispersive media (tap water) has been examined assuming different localization of the object. In such case the dependence of impedance on frequency takes form of semicircle (Fig. 10). Position of the center and the radius of the semicircle depend on distance between the body and surface of the electrode matrix.

The bigger is the distance the smaller radius of impedance plot and in fact the lower value of impedance. This may be explained by the fact that increasing the distance between the body and the matrix of electrodes reduces amount of measurement current flowing through the body. Thus, the body impedance is paralleled by impedance of water. The higher volume of water localized between the body and electrodes the smaller value of impedance. It appears that the impedance of bathtub filled only with water for low frequency, was lower than that of the body.

Increasing volume of the body in comparison to volume of the water leads to wider dispersion (Fig. 11). In this experiment the distance between the electrodes and the body remained constant. The volume of the body was being increased by expanding it laterally and toward upper limits of the model.

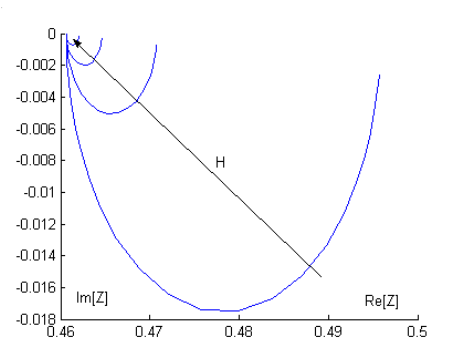


Fig. 10. Influence of the height of the dispersive object above electrode level on spectroscopic measurement (result of simulations)

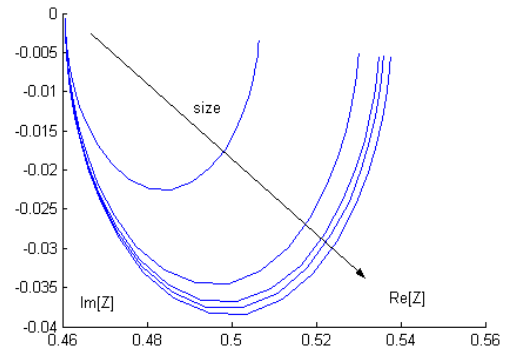


Fig. 11. Influence of the size of the object on the measured impedance values

It was achieved by substitution of water by body tissue in the model.

#### B. Phantom measurements

We have also performed experiments with a body immersed in the bathtub. Measurements have been performed using a developed model of the measurement system. Before attempting measurements in the bathtub a phantom studies were performed. We used a plastic tank filled with ordinary tap water. At the bottom of the tank a set of Ag electrodes was placed. After calibration we have immersed different objects (hand, leg, etc.) into tank and performed measurements with frequency range 200 Hz to 100 kHz.

Data have been collected using specially developed software (Fig. 12). Results of such measurements are presented in Figs. 13 and 14. However, when immersing the hand or leg in the tank led to quite big current leaks to surroundings at high frequency. So, we decided to reduce the range of current frequency to 80 kHz. It also resulted from low accuracy of measurement system for higher range of frequency. Nevertheless, a character of the measured impedance plot is similar to that obtained from simulation study.

Another presentation of impedance data is shown in Fig. 14. The curves were obtained for the following conditions: f1 – small part of biological tissue around electrodes, w1, w2 – bathtub filled only with water, b1, b2 – the human body directly on electrode matrix, b3 – the human body at distance (a few centimeters) from the electrode matrix.

Finally, measurement of ECG was performed with person sitting in the bathtub filled with a tap water. A three

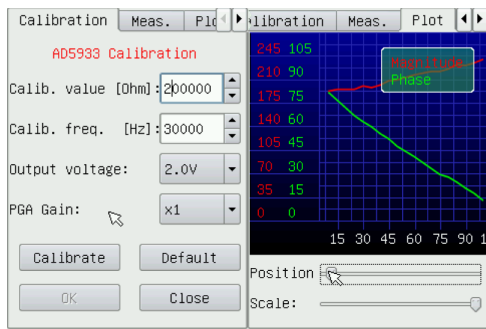


Fig. 12. An example of screen-shot from the embedded application

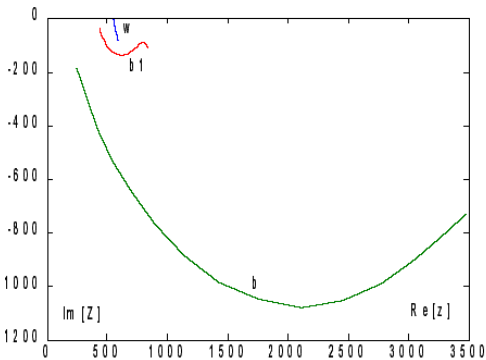


Fig 13. Real measurements of the measured impedance over frequency range from 200 Hz to 80 kHz: b – body directly on electrodes, b1 – body about 1cm above electrodes, w – water only.

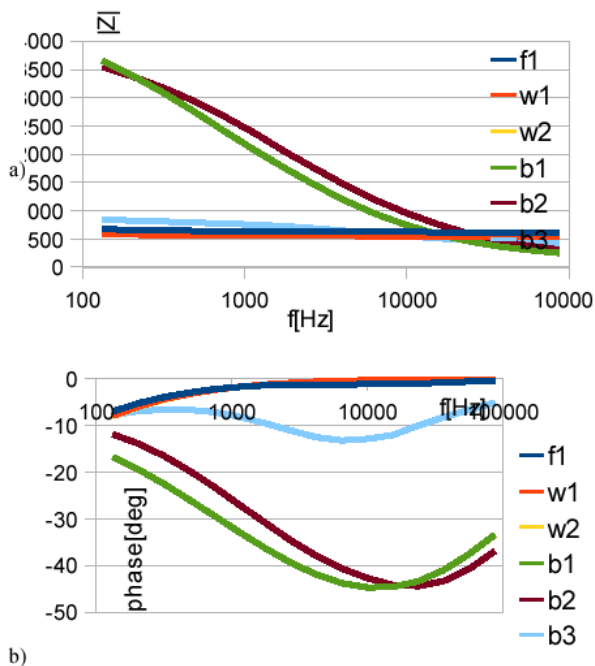


Fig 14. Dependence of impedance modulus and phase on frequency (details in the text)

electrode technique was employed. The result of measurement is presented in Figs. 15 and 16. Impedance was

measured only for one frequency of current equal to 10 kHz.

It appeared that person sitting on electrodes used for measuring impedance drastically increased its value

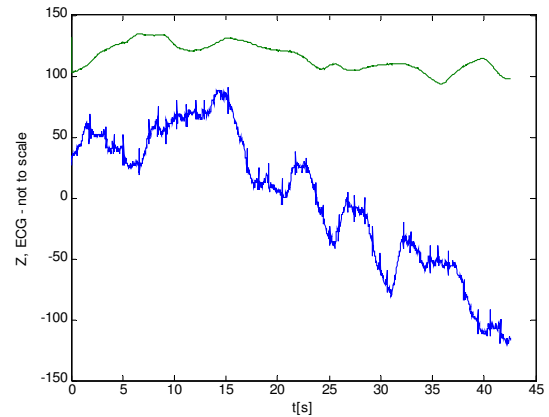


Fig. 15. Impedance measured at 10 kHz and ECG using separate matrices of electrodes

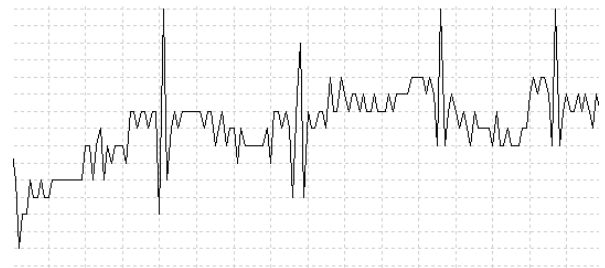


Fig. 16. Example of the ECG recording

in comparison to that obtained from theoretical studies by means of FEM method. However, the measured value was smaller, as expected, than that obtained in phantom studies. It was mainly due to much bigger object (bathtub vs. tank). As it is seen in Fig. 15 the recorded ECG contains a large component of very low frequency. Also, power frequency interference is very high. However, it is possible to monitor heart rate as QRS waveform is easily distinguished from other components of the signal.

#### IV. DISCUSSION

Home automation and so called intelligent buildings are regarded as an expensive toy for rich society. This is caused by cost of such installations. However, some parts of such installation becoming installed, besides the price – for example alarm systems. Similarly when considering protection and safety of elders some installations can be regarded as affordable. We are presenting such example of intelligent supervision of the bath.

Quality of the signals obtained during experiments suggests that further improvement has to be done yet. Nevertheless, it is shown that it is possible to detect the presence of the person in the bathtub and moreover it is possible to evaluate heart rate of that person. Thus, dangerous situations may be recognized, e.g. arrhythmia, additional excitations, etc.

This information is sent to caregiver and appropriate decision has to be taken.

However, to make the system a more reliable (to reduce a number of possible false alarms) another form (additional) of monitoring person's state when taking a bath is also considered.

#### V. CONCLUSIONS

We have showed that it is possible to detect presence of the person in the bathtub and to measure heart rate. However, quality of the recorded signals still has to be improved. Additionally it is possible to estimate person's activity by means of the impedance spectroscopy.

When the impedance measurements are stable it is possible to record ECG signal properly. Usage of readily available embedded platform reduces cost of development and reduces time of development. Additionally, popular operating system with well documented libraries allows easy development of modern applications.

#### ACKNOWLEDGMENT

This work has been partially supported by European Regional Development Fund concerning the project: UDA-

POIG.01.03.01-22-139/09-00 -"Home assistance for elders and disabled – DOMESTIC", Innovative Economy 2007-2013, National Cohesion Strategy.

#### REFERENCES

- [1] <http://www.telegesis.com> [June 2011]
- [2] T. Chibaa, M. Yamauchia, N. Nishidaa, T. Kanekob, K. Yoshizakib and N. Yoshiok, *Risk factors of sudden death in the Japanese hot bath in the senior population*. Forensic Science International 149 (2005) 151–158
- [3] K. Nakajima, K. Sekine, K. Yamazaki, A. Tampo, Y. Omote, H. Fukunaga, Y. Yagi, K. Ishizu, M. Nakajima, K. Tobe, M. Kobayashi and K. Sasaki, *Detection of respiratory waveforms using non-contact electrodes during bathing*, Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE, Aug. 31 2010-Sept. 4 2010, 911 - 914
- [4] T. Tamura, A. Kawarada, M. Nambu, A. Tsukada, K. Sasaki and K. Yamakoshi, *E-Healthcare at an Experimental Welfare Techno House in Japan*, The Open Medical Informatics Journal, 2007, 1, 1-7
- [5] A. Mori, K. Kusano, S. Nagase, K. Nakamura, H. Morita and T. Ohe, *Clinical Impact of the Hot-Bath Test in Patients with Brugada Syndrome*, Circulation. 2008;118:S\_674.
- [6] Gesselowitz D. An Application of electrocardiographic lead theory to impedance plethysmography, IEEE Trans. Biomed. Eng., vol 18, pp. 38-41, 1997
- [7] J. Ferreira, F. Seoane, A. Ansele and R. Bragos, *AD5933-based spectrometer for electrical bioimpedance applications*, J.Phys., Conference series, vol. 244, No. 1
- [8] <http://www.trolltech.com> [June 2011]



## Low-coherence method of hematocrit measurement

Małgorzata  
Jędrzejewska-Szczerska  
Gdansk University of Technology  
Narutowicza 11/12  
80-233 Gdańsk, Poland  
Email: mjedrzej@eti.pg.gda.pl

Marcin Gnyba  
Gdansk University of Technology  
Narutowicza 11/12  
80-233 Gdańsk, Poland  
Email:mgnyba@eti.pg.gda.pl

Michał Kruczkowski  
Gdansk University of Technology  
Narutowicza 11/12  
80-233 Gdańsk, Poland  
Email:michal\_kruczkowski@o2.pl

**Abstract**—During the last thirty years low-coherence measurement methods have gained popularity because of their unique advantages. Low-coherence interferometry, low-coherence reflectometry and low-coherence optical tomography offer resolution and dynamic range of measurement at the range of classical optical techniques. Moreover, they enable measurements of the absolute value of the optical path differences, which is still an unsolved problem in high-coherence interferometry. Furthermore, the use of the spectral signal processing makes this method immune for any change of the optical system transmission.

In this article the low-coherence method of a hematocrit measurement has been presented. Elaborated measurement method has many advantages: relatively simple configuration, potentially low cost and high resolution. Investigation of this method confirms its ability to determine the hematocrit value with appropriate measurement accuracy. Furthermore, results of experimental works have shown that the application of the fiberoptic low-coherent interferometry can become an effective base of method of the *in-vivo* hematocrit measurement in the future.

### I. INTRODUCTION

BLOOD analysis is frequently performed for medical diagnosis. One of the important analytes is the blood hematocrit (HCT), defined as the ratio of packed red blood cells volume to whole blood volume. The normal ranges of the hematocrit are 39-50% for male and 35-45% for female respectively. The high level of the hematocrit indicates risk factors for heart and cerebral infarction due to hemoconcentration. Dehydration or cerebral can be also origins of the HCT high level. Therefore, the hematocrit value as well as blood pressure should be controlled during the daily life as the indices of various physiological conditions in order to reduce the cardiovascular disease risk factor. It should be noted that the continuous monitoring of the HCT is also needed to perform appropriate dialysis and blood infusion.

The blood hematocrit is routinely determined in the clinic by analysis of blood samples. There are several methods of measurement of the HCT and hemoglobin. Unfortunately, almost all of them require either blood sampling or catheterization. Repeatable blood sampling during continuous monitoring of the hematocrit value is associated with an increasing

risk of infection (e.g. HIV or hepatitis). Moreover, such way of examination and loss of even small volume of blood required for the HCT measurement can be harmful to the patients (especially for neonates, small children and old people). Therefore, there is a need of non-invasive method of the hematocrit measurement, because recently used methods are based on the *in-vitro* measurement.

There is a great interest in optical measurement that would permit simultaneous analysis of multiple components (analytes) in whole blood without the need for conventional sample processing, such as centrifuging and adding reagents. There are few optical methods of the hematocrit measurement. Schmitt et al. [1] uses the dual-wavelength near IR-photoplethysmography. Intensity sensitivity detector working with dual wavelength has been described by Oshima et al. [2]. Xu et al. applied optical coherence tomography for investigating the HCT value [3]. Enejder et al. [4] used Raman spectroscopy and partial least squares (PLS) data analysis for simultaneous measurement of concentrations of multiple analytes in whole blood, including the hematocrit and hemoglobin. There are also works in which authors tried to find correlation between oxygen saturation of whole blood and hematocrit value [5, 6]. However, all of them investigated the HCT value *in-vitro* by using sample of blood. Recently, only one method of non-invasive hematocrit measurement by the use of spectral domain low coherence interferometry has been demonstrated by Iftimia et al. [7]. Unfortunately, until now it is possible to make such a measurement only insight the eye, which is uncomfortable for the patient and difficult because of the eye movement. The light beam interrogating the eye must be stabilized on a fixed location on a specific retinal vessel to collect reproducible depth-reflectivity profiles in the blood. This is possible only by employing the eye motion stabilization technique. The eye motion stabilization can generally be accomplished invasively with suction cups or retrobulbar injections, inaccurately with fixation, more precisely at slower speeds with a passive image processing approach, or at high speeds with active tracking. Fixation requires patient cooperation and is difficult in patients with poor vision. Therefore, at the present state of art *in-vivo* measurement of the HCT is more complex and expensive than laboratory analysis of collected blood samples.

The purpose of our study is to develop an accurate and stable low-coherence optical hematocrit measurement

This work was supported by the European Regional Development Fund in frame of the project: UDA-POIG.01.03.01-22-139/09-02 -“Home assistance for elders and disabled – DOMESTIC”, Innovative Economy 2007-2013, National Cohesion Strategy.

method. The preliminary stage of research, which is subject of this paper, includes analysis and selection of sufficient configuration of the low-coherence system for the HCT determination, setting up the low-coherence measurement system and *in-vitro* confirmation whether proposed method has sufficient accuracy to be the base of system which, if manufactured successfully, could have impact to development of the systems for *in-vivo* monitoring of the hematocrit.

## II. LOW-COHERENCE MEASUREMENT METHOD

The low-coherence interferometric measurement system consists of a broadband source, a sensing interferometer and an optical processor (see Fig. 1).

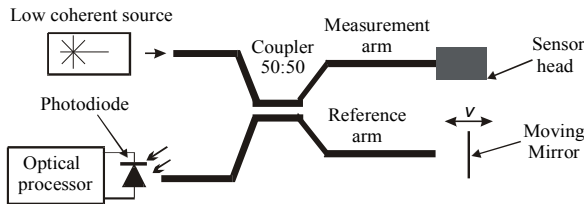


Fig. 1. Basic low-coherence interferometry setup.

The light from the broadband source is transmitted to the sensing interferometer by the coupler and a fiber optic link. At the sensing interferometer the amplitude of light is divided into two components and an optical path difference (OPD), which depends on the instantaneous value of the measurand (mainly its dimension or refractive index), is introduced between them. The sensing interferometer is designed in such a way that a defined relationship exists between the optical path difference and the measurand. The signal from the sensing interferometer is transmitted back by the fiber to the optical processor. The optical processor consists of a second optical system, the output of which is a function of OPD generated at the sensing interferometer. The sensing interferometer is located inside the measurand field whilst the optical processor is placed in a controlled environment far from the field. The optical processor is either a second interferometer (when the phase processing of the measured signal is used) or a spectrometer (when the spectral processing of the measured signal is used). The measurement system with the phase processing of the measured signal has very high measurement sensitivity and resolution, much higher than that of the system with spectral processing. However, the system with spectral processing of the measured signal has two important advantages. It does not need movable mechanical elements for precise adjustment as well as it is immune to any change of the optical system transmission. This is possible because in such a system the information about the measurand is encoded in the spectra of the measurement signal. Optical intensity at the output of such an interferometer can be expressed as [10]:

$$I_{out} = \langle EE^* \rangle \quad (1)$$

where:  $E = E_1 + E_2$ ,  $E_1$  and  $E_2$  – amplitudes of the electric vector of the light wave reflected from the first and the

second reflective surfaces inside the sensing interferometer respectively; brackets  $\langle \rangle$  denote time averages, asterisk \* denote the complex conjugation.

When the spectral signal processing is used, the recorded signal can be expressed as [11]:

$$I_{out}(\nu) = S(\nu) \left[ 1 + V_0 \cos(\Delta\phi(\nu)) \right] \quad (2)$$

where:  $S(\nu)$  – spectral distribution of the light source;  $V_0$  – visibility of the measured signal,  $\Delta\phi(\nu)$  – the phase difference between interfering beams.

The phase difference between interfering beams can be calculated from a following equation [8]:

$$\phi(\nu) = \frac{2\pi \nu \delta}{c} \quad (3)$$

where:  $\delta$  – optical path difference,  $c$  – velocity of the light in vacuum.

If the light source exhibits a Gaussian spectrum, the normalized spectra pattern is predicted to be a cosine function modified by the Gaussian visibility profile. In the spectral domain signal processing the modulation frequency of the measurement signal gives information about the measurand (Equation (2)), as shown in Fig. 2.

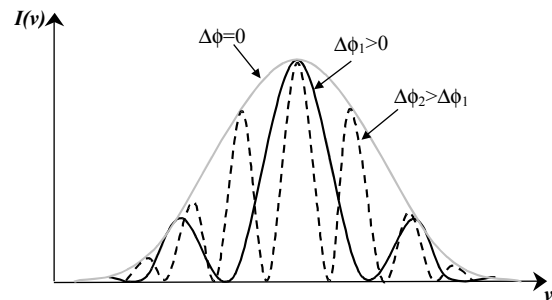


Fig. 2. Calculated measured signal from interferometer with spectral signal processing for different phase differences between interfering beams. ( $\Delta\phi$  – phase difference).

It can be noted that for  $\Delta\phi = 0$  there is no spectral modulation. If the phase difference between the interfering beams varies from zero, the function takes the form of the cosine curve. The spacing of adjacent transmission peaks is proportional to the inverse of the optical path difference ( $1/\delta$ ) [12].

In this processing, it is necessary to use special measurement equipment and advanced mathematical processing of the measurement signal. However, as it was mentioned earlier, low-coherence measurement system based on the spectral signal processing has significant advantages – there is no need of precise mechanical scanning and no need of use of movable adjusting components. Moreover, the system using the spectral signal processing and Fabry-Perot sensing interferometer is not sensitive for any change of a transmission of the optical system. This is possible because in the system information about the measurand is encoded in the spectra of the measured signal. Therefore such a setup is the most convenient for the low-coherence hematocrit measurement.

### III. EXPERIMENTAL

#### A. Measurement setup

The scheme of the developed low-coherence fiber-optic set-up for hematocrit measurement is shown in Fig.3. The optical spectrum analyzer (Ando AQ6319 or Anritsu MS9740A) was implemented as the optical processor. The superluminescent diode with Gaussian spectral intensity distribution was used as a low-coherence source. The optimal choice of optical source parameters is a very important problem in low-coherence measurement system because they affect the metrological parameters of such a system. Therefore, not only the central wavelength ( $\lambda$ ) and optical spectral bandwidth full width at half maximum ( $\Delta\lambda$ ), but the shape of spectral characteristic as well, should be selected very carefully. In presented research work the selection of optical parameters of the source was made to increase visibility of the measured signal. During laboratory tests conducted for a few sources having different parameters, the best results were obtained for superluminescent diode Superlum Broadlighter S1300-G-I-20 having following optical parameters:  $\lambda_0 = 1290$  nm,  $\Delta\lambda = 50$  nm.

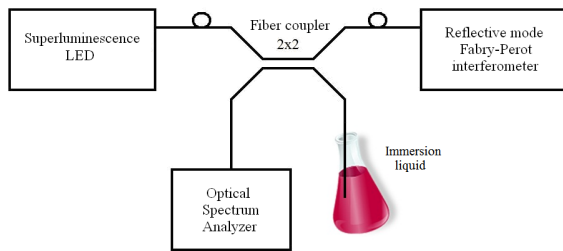


Fig. 3. The experimental setup.

As a sensing interferometer authors implemented a low-finesse fiber-optic Fabry-Perot interferometer, shown in Fig. 4. The theoretical investigation had shown that the use of a single mode optical fiber would be the most convenient. Hence, the Fabry-Perot interferometer was built using the standard single mode telecommunication optical fibers, a fiber coupler, the measurand field and a mirror. The interferometer consists of optic-fiber with uncoated end, which has a reflectance of 0.04. The second reflectance surface is made by the silver mirror with reflectance of 0.99. This configuration provided high visibility value of measured signal.

When the measurand field is filled by the investigated liquid, the reflective surfaces of the Fabry-Perot are made by two boundaries: fiber/liquid and liquid/mirror respectively. In such a setup each change of refractive index of the investigated liquid results in change of the optical path difference of interfering beams and the phase difference of those beams due to Equations 2 and 3.

#### B. Object of investigation

In order to find out whether proposed method has sufficient accuracy to be base of system for *in-vivo* monitoring of the hematocrit in the future, serie of *in-vitro* measurements

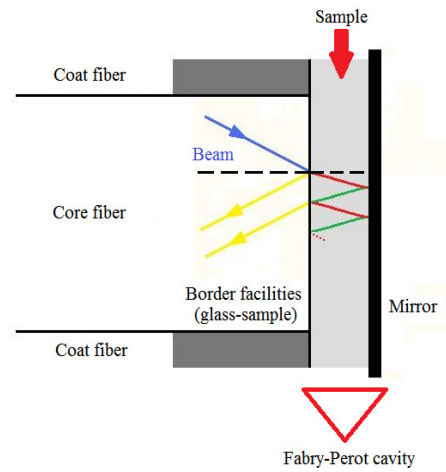


Fig. 4. The experimental setup.

was carried out. During experimental work authors used the whole human blood for tests. Set of 2 ml blood samples with various hematocrit levels were provided by the Gdansk Blood Donor Centre. Such approach has significant advantage, because we were able to use wide representative group of volunteers. It should be noted that samples were get from rather healthy volunteers and therefore our measurement range of the hematocrit measurement was limited to the value of 30 to 50%. However, this range was wide enough to find out the relationship between the HCT and measured optical signal and if resolution of the measurement is sufficient, as well. At this stage we were not measuring blood samples having very extreme HCT values which refers to very sick patients. This will be the object of the future research work. Detailed information the HCT distribution in investigated samples is given in Table 1.

HCT range	Number of samples
32,2 ÷ 33,8	12
34,0 ÷ 35,8	9
36,1 ÷ 37,9	12
38,0 ÷ 39,8	10
40,0 ÷ 41,5	13
42,1 ÷ 43,9	14
44,1 ÷ 45,8	19
46,2 ÷ 47,6	6
48,0 ÷ 49,0	4

Moreover, the hematocrit level of each blood sample was obtained by clinical diagnostics with accuracy  $\pm 0.1$  at the Gdansk Blood Donor Centre as reference measurements, as well. Our investigation, as well as clinically research, showed that obtained HCT level of blood probes were stabled for 72 hours. During all part of our measurement

process the temperature of blood probes were restricted controlled.

#### IV. RESULTS AND DISCUSSION

With the use of elaborated low-coherence system with Fabry-Perot interferometer, the hematocrit value of numerous blood sample was measured. Experimental investigation gave a series of recorded spectra. In Fig. 5 measured reference signal from sensor without any liquid is presented. Measured signals of the blood sample with  $HCT = 35.2\%$  and with  $HCT = 49.2\%$  are shown in Fig. 6a and 6b respectively.

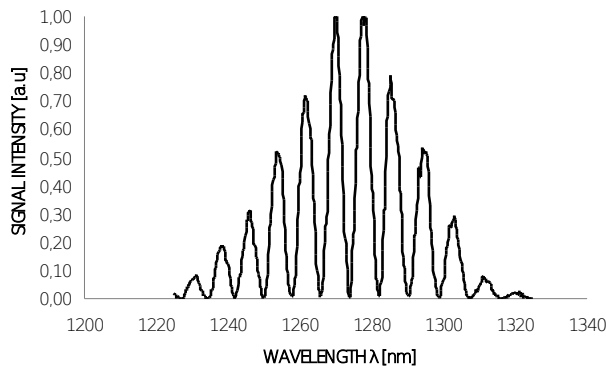
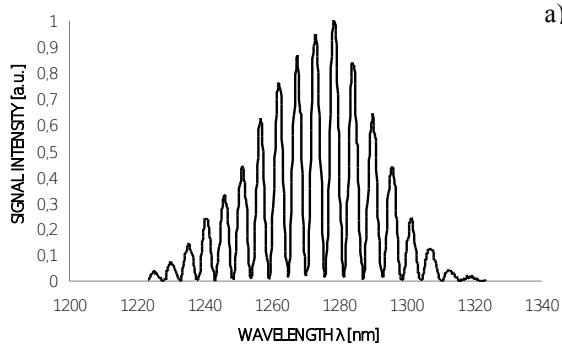
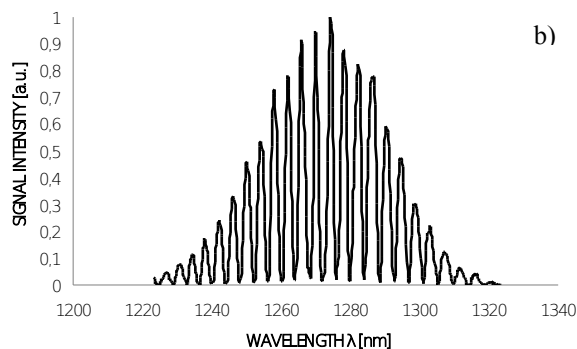


Fig. 5. The reference signal.



a)



b)

Fig. 6. Measured signal of the blood sample with a)  $HCT=35.2\%$ , b)  $HCT=49.2\%$ .

It should be noted (Fig.5, Fig.6) that by the use a dedicated sensing interferometer, designed in our laboratory, it was possible to get visibility of the measured signal  $V=0.98$ , which is really hard to achieve in really optoelectronic low-coherence system.

As there are shown in Fig. 6a and 6b, the change of the hematocrit value changes the refractive index of the blood sample and optical path difference between interfering beams as well. It occurs in phase changes (Equation 2 and 3) and can be detected by the analysis of the measured signal modulation or by counting number of fringes in the measured signal. The change in the number of fringes in the measured signal due to HCT level is shown in Fig. 7. Additionally, the deviation of the number fringes during HCT level measurement is presented in Fig. 8

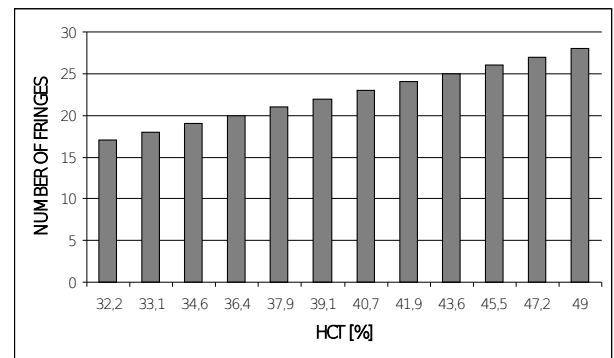


Fig. 7. Number of fringes in the spectra of measured signal vs. HCT level.

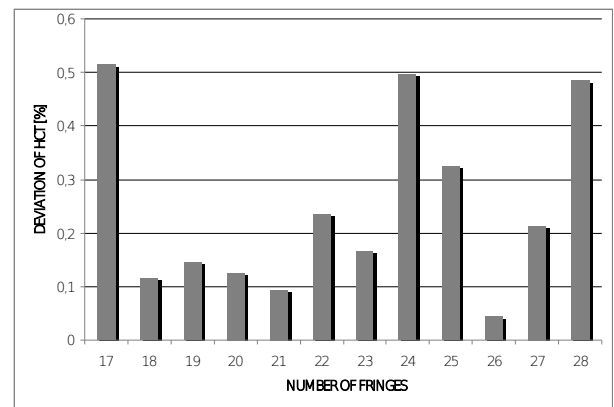


Fig.8. Number of fringes in the spectra of measured signal vs. HCT level.

Mathematical processing of the measured signals provided information about measurand from the spectrum of signal is shown in Fig. 9.

The result of experimental works shows, that can be seen in Fig. 7, that low-coherence method of hematocrit measurement with Fabry-Perot interferometer configuration has proper accuracy to measure the hematocrit value. The method was elaborated in the HCT range extending from 30% to 50%. The output signal was analysed by measurement the change of fringe numbers of the spectra pattern. The change of the number of fringes of investigated spectral pattern equal to 8 was achieved in investigated range, thus obtained sensitivity of the hematocrit measurement can be

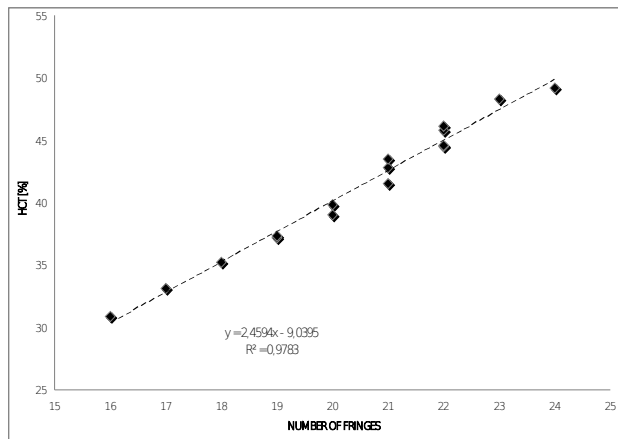


Fig. 9. Experimental results - change of number of signal spectra fringes vs. hematocrit value: dots – measured value, dash line – regression line. ( $R^2$  - determination coefficient of statistical model of HCT vs. number of fringes relationship)

estimated as 0.4 [a.u./%]. Time required for single measurement was in range of 0.8-1.2 s. Experimental configuration had very high the value of visibility of the measured signal (usually 0.98), what always leads to decrease of the value of signal-to-noise ratio of the measured signal. Furthermore, it had very high determination coefficient of statistical model of HCT vs. number of fringes relationship ( $R^2=0.978$ ) as well.

## V. CONCLUSIONS

In this paper the low-coherence method of the hematocrit measurement with spectral signal processing has been presented. Elaborated measurement system which used this method has numerous advantages: relatively simple configuration, potentially low cost, high resolution, dielectric construction. Furthermore, it had optical sensor head (Fabry-Perot interferometer) of small size and very low thermal inertia. What is more important, by utilizing spectral signal processing those sensors exhibit immunity of any changes of optical transmission in measurement system.

The results of experimental works showed that implemented experimental set-up provided good quality of the measured optical signals by offering great value of visibility of the measured signal, exhibits immunity for changes of the optical signal polarization and got simple configuration.

The investigation of this method confirms its ability for the hematocrit control with appropriate measurement accuracy. The presented preliminary results can be the base for building sensor ready for practical applications and in the opinion of authors it will be possible to build the low-coherence optical *in-vivo* hematocrit sensor in the future. In the further stage of the research, developed system will be applied for the *in-vivo* measurements carried out through the skin. The best place on the human body for measurement as well as method of preparation patient skin surface will be determined. As during the *in-vivo* measurements results would be probably influenced not only by refractive index of the blood but also by change of Fabry-Perot cavity dimensions caused by a human pulse, we predict that phase-sensitive detection synchronised with the human pulse should be applied.

## REFERENCES

- [1] J. Schmitt, Z. Guan-Xiong, J. Miller, „Measurement of blood hematocrit by dual-wavelength near-IR photoplethysmography”, *Proc. of SPIE*, vol. 1641, pp.150-161, 1992.
- [2] S. Oshima, Y. Sanakai, „Optical measurement of blood hematocrit on medical tubing with dual wavelength and detector model”, in *Proc. 31st Annu. Conf. IEEE EMBS*, Minneapolis, 2009, pp. 5891-5896.
- [3] X. Xu, Z. Chen, “Evaluation of hematocrit measurement using spectral domain optical coherence tomography”, in *Proc. Conf. 2008 International Conference on BioMedical Engineering and Informatics*, 2008, Sanya, pp. 615-618.
- [4] A. M. K. Enejder, T.-W. Koo, J. Oh, M. Hunter, S. Sasic, M. S. Feld, “Blood analysis by Raman spectroscopy”, *Optics Letters*, Vol. 27, No. 22, 2002, pp. 2004-2006.
- [5] M. Nogawa, S. Tanaka, K. Yamakoshi, “Development of an optical arterial hematocrit measurement method: pulse hematometry”, in *Proc. 27th Annual Conference Shanghai*, pp. 2634-2636, 2005.
- [6] S. Takatani et al., “A miniature hybrid reflection type optical sensor for measurement of hemoglobin content and oxygen saturation of whole blood”, *IEEE Trans. Biomedical Engineering*, vol.35, pp. 187-198, March 1988.
- [7] Ifitimia N. et al., “Toward noninvasive measurement of blood hematocrit using spectral domain low coherence interferometry and retinal tracking”, *Optics Express* vol.14, pp. 3377-3388, April 2006.
- [8] K. Grattan, B. Meggit, *Optical Fiber Sensor Technology*, Boston: Kluwer Academic Publisher, 2000.
- [9] M. Jędrzejewska-Szczerska, B. Kosmowski, R. Hypszer, “Shaping of coherence function of sources used in low-coherent measurement techniques”, *Journal de Physique IV*, vol. 137, pp. 103-106, 2006.
- [10] F. Yu [ed], *Fiber Optic Sensors*, New York: Marcel Dekker, 2002.
- [11] S. Egorov, A. Mamaev, I. Likhachiev, “High reliable, self calibrated signal processing method for interferometric fiber-optic sensors”, *Proc. SPIE*, vol.2594, pp.193-197, 1996.
- [12] M. Jędrzejewska-Szczerska et al., “Fiber-optic temperature sensor using low-coherence interferometry”, *The European Physical Journal Special Topics*, vol.154, pp.107-111, 2008.





# Multimodal platform for continuous monitoring of elderly and disabled

Mariusz Kaczmarek  
Gdansk University of Technology  
Narutowicza 11/12, 80-233  
Gdansk, Poland  
email: mariusz  
@biomed.eti.pg.gda.pl

Jacek Ruminski  
Gdansk University of Technology  
Narutowicza 11/12, 80-233  
Gdansk, Poland  
email: jwr@biomed.eti.pg.gda.pl

Adam Bujnowski  
Gdansk University of Technology  
Narutowicza 11/12, 80-233  
Gdansk, Poland  
email:  
bujnows@biomed.eti.pg.gda.pl.

**Abstract**—Health monitoring at home could be an important element of care and support environment for older people. Diversity of diseases and different needs of users require universal design of a home platform. We present our work on a sensor-based multimodal platform that is trained to recognize the activities elderly person on their home. Two specific problems were investigated: configuration and functionality of central workstation as a module for data acquisition and analysis and second problem is devoted to user’s home environment monitoring.

## I. INTRODUCTION

HEALTH personalization and support for older and immobilized people is actually very important target of many national and international initiatives (e.g. Framework Program 7, Hong Kong “Care for the Elderly 2007 - Active Mind”). Different research areas are connected with those initiatives including “Wearable Sensors (WS)” [1], “Body Area Sensors (BAS)” [2], “Wireless Sensor Networks (WSN)” [3] and telemedicine methods [4]. As a result of this research different sensors and integrated solutions were proposed, usually dedicated for a particular goal. Designing a home platform for support of older or immobilized people different categories of existing and possible components should be considered from particular sensors to central computer stations.

In case of systems with integrated sensors many solutions were proposed like Crossbow IRIS [5], Sun SPOT [6], eWatch [7], Smart-Its [8] or other [9]. Many motes are currently under constructions, however they are usually equipped with embedded sensors (e.g. temperature, light, and location), expending slots (e.g. sandwich model) and communication modules (Bluetooth or based on IEEE 802.15). Dedicated, medical extensions are often proposed like results of CodeBlue project [10], MobiHealth project [11] or UbiMon (Ubiquitous Monitoring Environment for Wearable and Implantable Sensors) project [12]. Typical so-

lutions used for such extensions (or standalone systems) are universal medical diagnostic devices, including ECG, pulse oxymeter, blood pressure/pulse monitors, etc.

Communication interfaces allow data acquisition (especially at home) from motes to a one central station (or a middleware). The central station may be used to process data to assess user state based on many parameters and inform a user relatives or healthcare professional (a nurse, general practitioner) about the patient condition. The central station is often required to limit data processing at sensor node and to build an integrated view on the patient (including ontology based context models [13]).

Recent advances in sensor technology, cellular networks and information technology allow to improve the well-being of the elderly by assisting them in their everyday activities, monitoring their health status and environment conditions. There is more and more projects concerning the “smart” or “intelligent” homes with prototyping information-sensor systems recognizing habitant activities and abnormalities of them [14][15][16][17]. It is possible to develop systems that recognize an individual’s everyday activities and automatically detect changes in the behavioral patterns of people at home that indicate declining health [18].

Human-computer interface (HCI) for older citizens or immobilized people/patients is also very important aspect of the home-based system. Specially designed user-interfaces and interaction devices are often required.

The main goal of this paper is to present a design of the multimodal, integrated platform for communication, training and health and environment monitoring at home. Communication includes technical and functional methods of a user communication with his/her environment as well as processing of alerts from a home/user sensor network. The training is mainly related to promote mental activity of a group of patients in danger (e.g. patients with dementia). Home monitoring is devoted to collect patient-related data and home environment data (e.g. fire detection). The very important aspect of the presented platform, is designing of a central computer station to collect data, process events/alerts, supply a proper human-computer interface, etc. In section II

This work was partly supported by European Regional Development Fund concerning the project: UDA-POIG.01.03.01-22-139/09-00 -“Home assistance for elders and disabled – DOMESTIC”, Innovative Economy 2007-2013, National Cohesion Strategy.

and III the design of the proposed system is presented. Next results of first module implementations are shown, including human-computer interface for immobilized patient and a module for home environment monitoring.

## II. PLATFORM STRUCTURE

The aim of our work is to create a smart system that will be adaptable to individuals, will be able to recognize their activities and will help their well being by raising alarms when a potential departure from routine or desirable behavior is detected (for example, the individual did not eat lunch or appears to not take their medication at prescribed times).

Home monitoring of people/patients is a wide term and different applications are possible. Some users are highly immobilized (only basic head/eyes communications) others are free to move but suffer dementia. The platform design requires creating a multi-modular system. The most important features of the platform are human-computer interface (or human-platform interface), middleware (e.g. used for data processing) and sensor/communication nodes.

### A. Central Workstation

The role of central workstation combines the middleware functionality and HCI (described later). The following functionalities are requested from the central workstation:

- database management,
- sensor data collection and processing,
- events dispatching and processing (e.g. “take a pill” and “no activity alarm”),
- data integration, classification, rules induction and other activity related to create overall person/patient model including his/her environment,
- communication management (e.g. which interface should be used to send a particular message),
- access control and support for other security mechanisms,
- support for mental training (actually discussed with psychologists/neurologists),
- Human Computer Interface.

TV-set connected to personal computer can be used as central workstation – Fig. 1. Of course it is possible to use standard computer monitor or touchable monitors.

The human-computer interface is one of the most important elements of the platform. The user acceptance of the entire system depends highly on the method how the system can be used by older/immobilized users. Three elementary modes of the HCI were designed:

- interface based on touch screen,
- visually guided interface,
- audio guided interface.

All interfaces require a new design of the graphical user interface. This includes to prepare a platform front end (with selection of services divided into three groups: emergency,

medical, and personal activity), GUI of each service and virtual devices (e.g. keyboard, remote control).

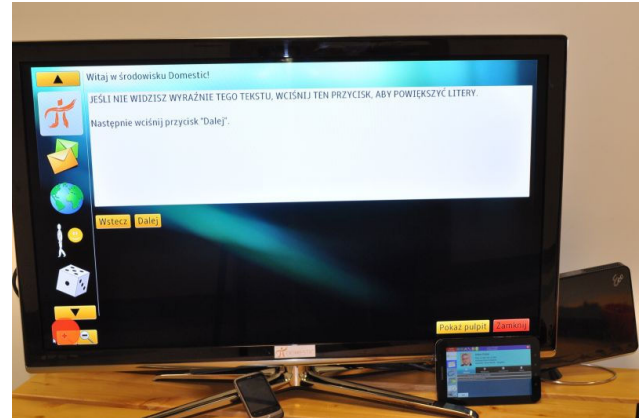


Fig. 1 TV set as a Central Workstation – example of Graphic User Interface also portable tablet with proper software is presented

Visually guided interface is especially designed for this category of patients, which are unable to move. Even in this group there are different subcategories that should be taken into account (patient can use only eyes; patient can move his/her head; patient can move head and a hand, etc.).

We assumed that the platform should also use existing internet services, however, redesigned for the older/immobilized people.

The General Purpose Input Output (GPIO) interface was used to connect different sensors for user’s home environment monitoring. Currently, a thermistor-based temperature measurement sensor and an optoelectronic-based movement detector, water leakage and current load detectors were implemented. Humidity, fire and gas detectors are under construction.

The Siemens TC65T module was used as a main controller for home environment monitoring multi-sensor. The module combines GPIO and GSM communication interface. Using Java 2 Micro Edition a set of Midlets was prepared to process GPIO events and communicate with the environment. The module is used as one of the external communication interfaces. Communication between sensors and central station is realized using ZigBEE standard.

The central workstation was build using Java 2 Enterprise Edition, Apache Web server and MySQL database management system. Events from the sensor modules are stored in MySQL tables and can be processed and visualized using dynamically constructed web page. The external access to information services is limited by access lists and is secured by SSL. Additionally, a set of J2ME Midlets was prepared for mobile phones to remotely manage the selected sensor module. Midlets allow to set alarm limits (e.g. smallest and highest acceptable temperatures) which are used to decide about alert generation to a privileged user (e.g. relative, guardian). The module accepts also SMS requests (from the configurable phone numbers) for sending current status data (request-response model).



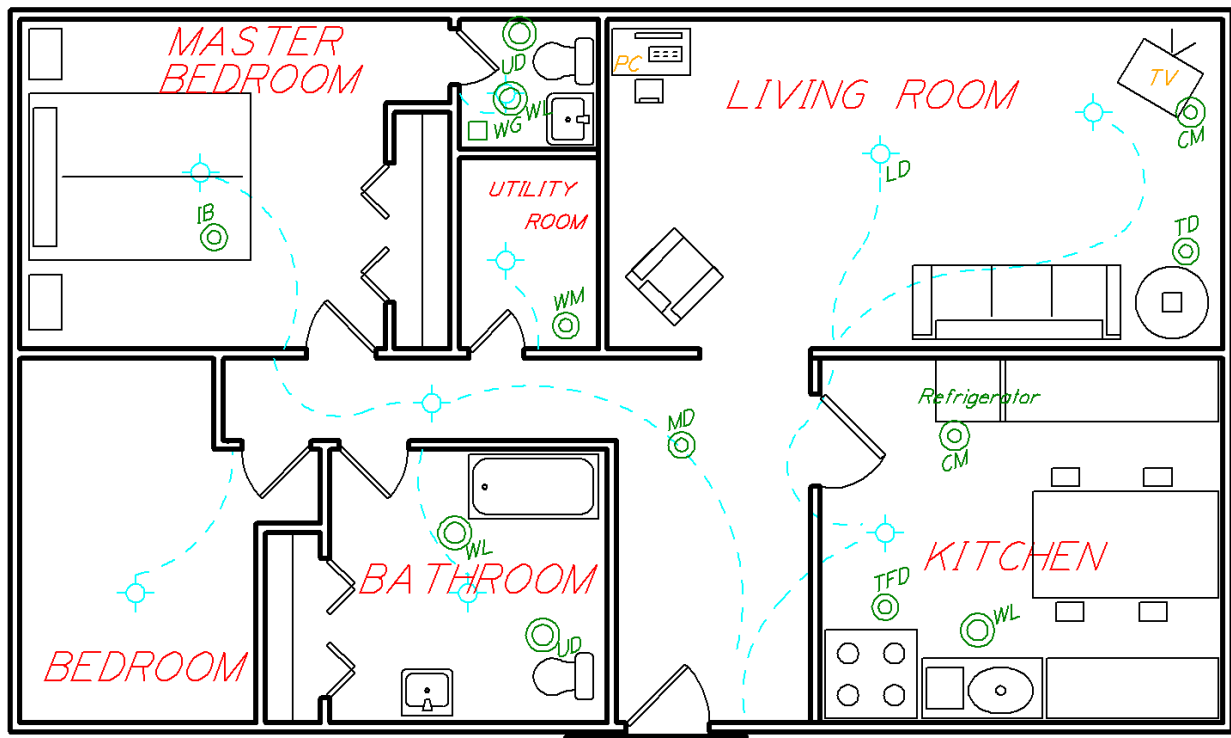


Fig. 2 Floor plan with sensor placement

### B. Sensors

To allow digital devices to be treated as ‘everyday stuff’ we need to open up access to the supporting infrastructure that connects devices and provide users with a simple model that allows them to manage the introduction and arrangement of new interactive devices.

Multimodal platform will use different sensors (or multi sensors) to collect information about person/patient state and his/her environment. Different categories of sensors are currently prepared to measure: heart pulse, temperature, body composition parameters (weight, fat content, etc.), glucose concentration, blood pressure, electric heart activity (ECG), and posture activity (accelerometer). Additionally special multi-sensors are considered to design for a particular group of patients (disease-oriented). Placement of sensors is shown on Fig.2. Appropriate database structure is designed to collect information from each sensor. The one-to-many relationship is used between mother table (directory of sensors) and sensor tables. Each sensor has its own data table (tables). Particular data event is marked using timestamp so it is easy to analyze a set of measurements from many sensors during given period of time. The platform can be easily scaled with a new sensor (multisensory) using plug-in methodology (common interface, XML configuration file, a new data table).

Another group of sensors consists of those related to monitor user environment parameters. This is especially important for older people deciding to live alone without

permanent, personal help. Taking into account the privacy of those persons, different sensors can be required to observe fire and gas dangers, humidity/temperature conditions at home activity of the person, emergency calls (e.g. symptoms of heart attack, ischemia, etc.).

### III. SENSORS CONFIGURATION

An exemplary flat plan is shown on Fig.2. Living spaces is consisting of two bedrooms, two bathrooms, kitchen, living room and laundry/utility room. In addition, there will be a shared basement with a home entertainment area with centralized computing services. Different wireless sensors are placed in the habitation area – indicated green on the Fig.2. There are:

- CM – current load meter and switch,
- IB – intelligent bed, with medical diagnostic devices,
- LD – light detector,
- MD – movement detector, sensors that can detect movement over whole flat (indicated blue line),
- TD – room temperature detector,
- TFD – temperature and fire detector,
- UD – urine detector,
- WG – weight meter,
- WL – water leak detector,
- WM – water meter.

Entertainment/main station devices used for acquire and analyze sensors data and for personal rehabilitation and activate purposes:

- TV set – which can be also a central station of the multimodal platform controlled by special designed remote control,

- PC – personal computer – and also central station of the multimodal platform.

System will include human position tracking through ultrasonic sensors or RF technology.

Activities in a smart environment include physical activities as well as interactions (with objects) can be monitored. For example, activities may include walking, resting on a couch and using the coffee machine, refrigerator, etc. An important rule is that these activities are not instantaneous, but have distinct start and end times. In addition, well-defined time relations exist between the events constituting an activity. These temporal relations are important in the determination of the monitored user's indoor activities and can be used for knowledge and pattern discovery in day-to-day activities.

#### A. Current consumption measurement

Electrical current consumption monitoring can be achieved by means of monitoring of the AC current drawn by the load. This can be achieved assuming that the amplitude of the power network voltage is constant during measurement period. This simplifies significantly measuring circuit. AC current is monitored by means of the current transformer. Current induced in the transformer is converted into voltage, amplified and rectified. This signal is converted to constant value by means of filtration. Voltage that is proportional to amplitude of the current can be directly measured but instead it is integrated and value of the signal is measured with specified and constant acquisition period. This allows to detect even very short current peaks. Results of measurement is stored in internal memory of the microcontroller and transmitted to central system by means of the ZigBEE network. However, we have designed and developed two different power meters. One that is working on a simple principle of AC current monitoring and second more precise – that utilizes readily available integrated circuit (ADE7753 from Analog Devices) for power consumption measurement. The block diagram of first circuit is shown on Fig. 3.

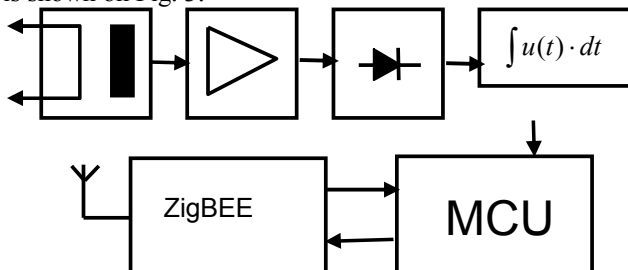


Fig. 3 Energy meter 1 block diagram

The circuit is detecting AC current by means of the current transformer. Signal is then amplified, rectified and filtered by means of low-pass filter. DC value of voltage is then equivalent to the momentary current drawn by the load – and assuming constant amplitude of the AC voltage – to the power delivered to the load. Momentary values are then integrated to avoid losing of the short power peaks that might occur. After read-out, the integrator is zeroed and is scoring the energy until the next read-out. Cyclic read-outs are collected and transmitted to the host system by means of the wireless connection using the ZigBEE standard.

Another project utilizes the integrated power meter. The idea of operation is very similar, but now we can measure energy more accurately. The simplified block diagram of the circuit is shown on Fig. 4a) where the photograph of the prototype is shown on Fig. 4b).

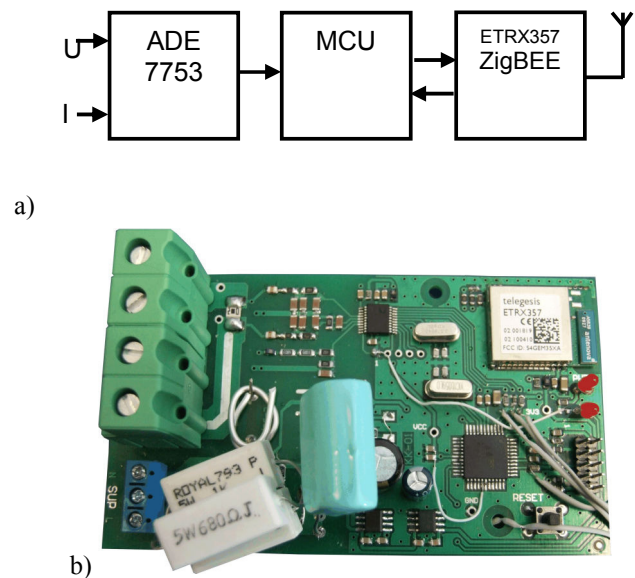


Fig. 4 Energy meter 2 a) block diagram; b) the prototype

#### B. Intelligent wall-socket

We also have developed the electronic circuit for electric wall socket - Fig. 5. The main task of the circuit is detection of the electrical current flowing to the load and possibility of remote power off of the socket e.g. in the case that supervised person leave the place leaving e.g. iron switched on. The power consumption is measured also by means of the current monitoring. The power delivery is controlled by means of triac with zero-crossing switch on ability. The communication with the system is also realized by means of ZigBEE

#### C. Intelligent wall-switch

Intelligent wall-switch (Fig. 6) circuit was developed to support classical wall-switch by option of the light measurement and detection of the switch state. Information about current state of the switch and ambient light intensity

is available by means of the ZigBEE communication standard.

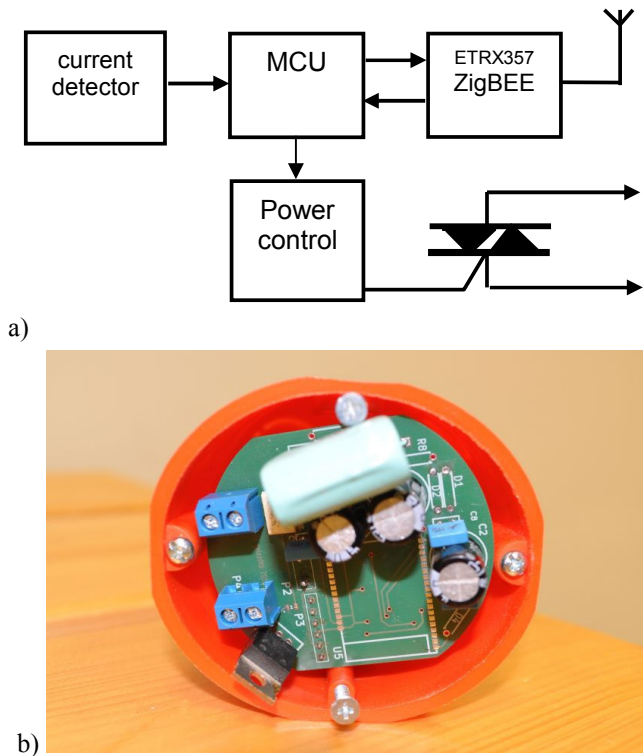


Fig. 5 Wall socket module a) block diagram; b) the prototype – CM (current meter)

Additionally it is possible to detect, whether the ambient light comes from (AC power supply or daylight).

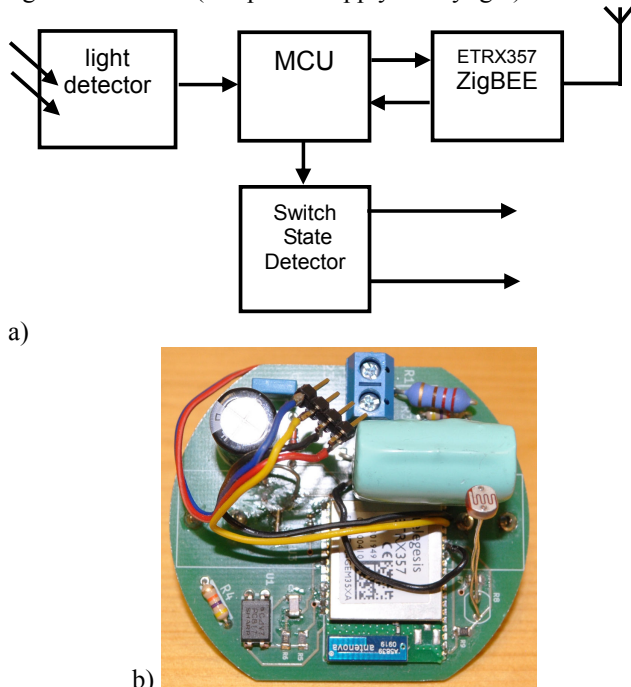


Fig. 6 An intelligent wall switch a) block diagram; b) the prototype

#### D. Water consumption measurement

We have developed two different water measurement meters working based on two different principles. First circuit is scoring pulses from mechanical water consumption meter. It requires no modification from the casual water meter as most of modern devices have rotating indicator. Thus, we are scoring the revolutions of the meter internal rotor by means of reflection of the IR light. Block diagram of the module is shown on Fig. 7a) where the working prototype is shown on Fig 7b).

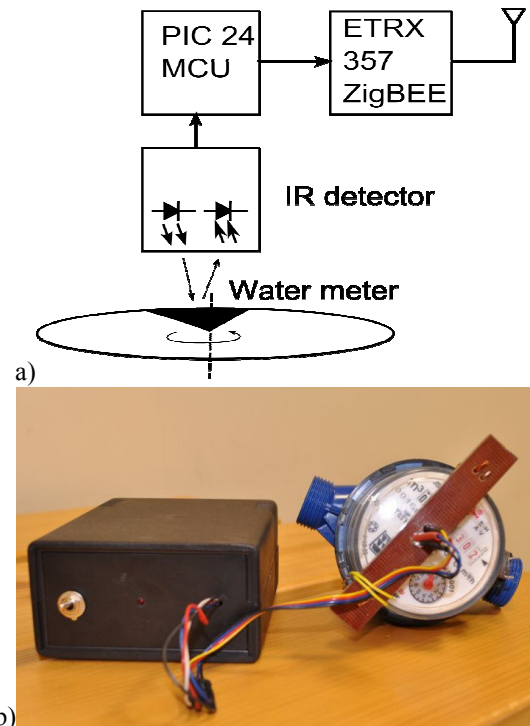


Fig. 7 Automatic detection of water leaking – WD, a) block diagram; b) the prototype

Unfortunately we are not always allowed to gain access to the water flow meter, for many reasons, but happily we are not interested in accurate water measurement, rather water flow information. Thus we developed also a water flow sensor, that can report binary information about state of the water in the pipe – is it flowing or not. Water flowing in the pipe makes the noise that can be picked up by means of the microphone. The amplified acoustic signal is then filtered and processed by means of the microcontroller. The microcontroller (Atmega8 by Atmel) is also communicating with the central system by means of the ZigBEE standard. The block diagram of the detector is shown on Fig.8a where on Fig. 8b) the working prototype can be seen.

#### E. Water leak detector

The water leak detector is a module that detects water leakage on the floor. Device is build on the basis of the ZigBEE module, microcontroller and resistance meter that is detecting unit.

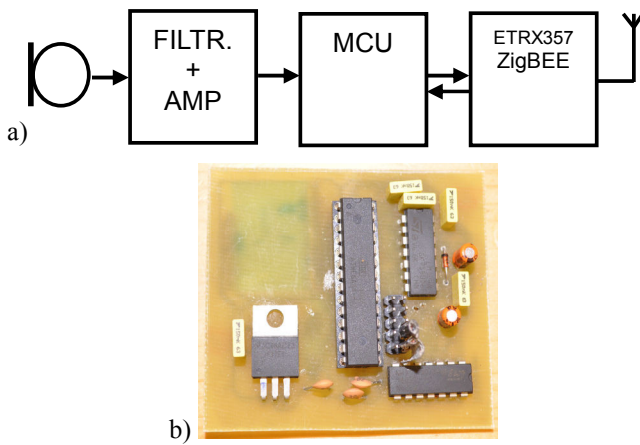


Fig. 8 Water flow sonic detector diagram and prototype

Device is battery powered and thus it can be placed anywhere, where is a risk of water leakage. The block diagram of the device is shown on Fig. 9a) where on Fig. 9b) the prototype is shown.

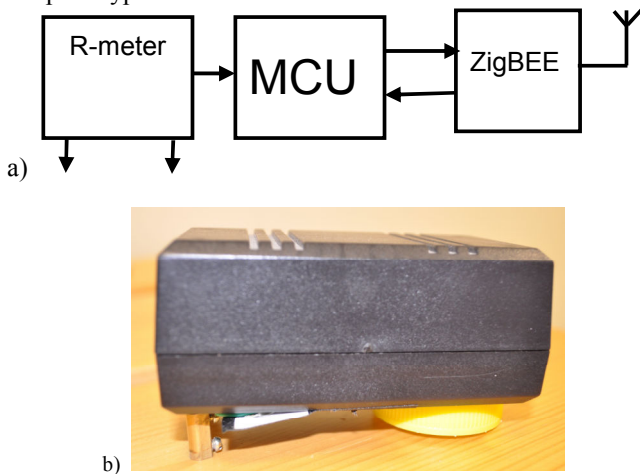


Fig. 9 Water leakage detector a) the block diagram; b) the prototype

#### F. Sound detector module

Sound detection module is a device that can be placed anywhere in the room. Its task is to detect and classify all surrounding sounds and perform its classification. The classification is performed within device by means of the embedded digital signal microcontroller (dsPIC30f6014 from Microchip). The device can distinguish between groan, fall, door closing and several others. Recognized sound is

classified and reported to the central system, which will take adequate action. The block diagram of the device is shown on Fig. 10a) where the photo of the prototype can be found on Fig. 10b).

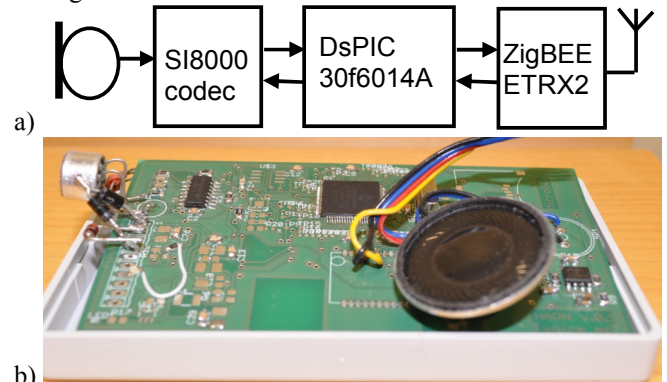


Fig. 10 Ambient sound detection module a) block diagram; b) working prototype

#### IV. ALERTS

Alerts are messages generated by the person aided, sensor network nodes, medical devices or the central station (as a result of processing of measurement data) to inform the people supported (internal alerts) and individuals (relatives, guardians, doctors – external alerts) of the occurrence of a particular situation. Alerts are divided into internal and external alerts. Table 1 presents the mutual relationship between the types of alerts and typical actors of the support system. Particular situations have been classified into the following priorities:

0 (emergency) - critical situation concerning the danger to life or of injury / incident (including a call for rescue of persons assisted), etc. This level is an "emergency" that requires call the appropriate emergency services.

1 (alarm) - alarm situation concerning the health status of the monitored person or his environment (including the request for assistance from the assisted person). Requires immediate action from a relative / guardian / physician. It does not require an automatic emergency call.

2 (warning) - the situation of concern regarding activities assisted person (broken diet, problems with taking medications, applying procedures, rehabilitation, etc.). This requires rapid intervention relative / guardian.

3 (system error) - the situation of concern associated with

TABLE I.  
THE MUTUAL RELATIONSHIP BETWEEN THE TYPES OF ALERTS AND TYPICAL ACTORS OF THE SUPPORT SYSTEM

Alert priority/alert recipient	Emergency teams	GP/ Personal doctor	Nurse/social service	Relatives/guardians	Neighbors	Assisted person
0 (emergency)	X	X	X	X	X	X
1 (alarm)		X	X	X	X	X
2 (warning)			X	X		X
3 (system error)				X		X
4 (serious problem)			X	X	X	X
5 (problem)						X



the technical problems with the support system - required the intervention of a relative / guardian and, possibly, maintenance of the system

4 (serious problem) - the occurrence of a failure in the home environment, or request for help / information generated by the person aided - for a person aided and relative / guardian

5 (problem) - the occurrence of a failure in the home environment (or a simple remainder) - for a person assisted.

Alert system was implemented in Java and Java FX. Concurrent processing (java.io.concurrent package, e.g. PriorityBlockingQueue) has been used to map N-sources to M-alerts and K-recipients. Each source (e.g. water consumption sensor, ECG sensor, etc.) is related to an observable producer, which delivers XM-based message to the central station. The central station processes particular or combined messages to identify the priority of the situation and send the alert to the consumers. Consumers distribute alerts to particular recipients. It is assumed that at least four communication channels can be used:

- visual and audio messages for the assisted person;
- audio or/and visual messages for neighbors (environment of the assisted person home),
- messages transmitted by GSM module (SMS, audio messages, data alerts),
- messages transmitted by the secondary data channel (e.g. analog phone modem, internet connection, etc.).

When the situation occurs the message is received by central station (internal alert) and is processed to prepare appropriate response. One possible response is audio/visual information to the assisted person (locally) or external recipient (processed message in the dedicated software). In Fig.11 the examples of windows are presented for different priorities of the alerts. All alerts are recorded in the database as well as responses.

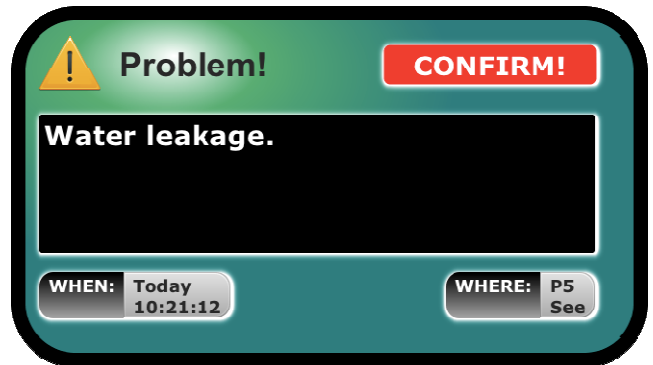
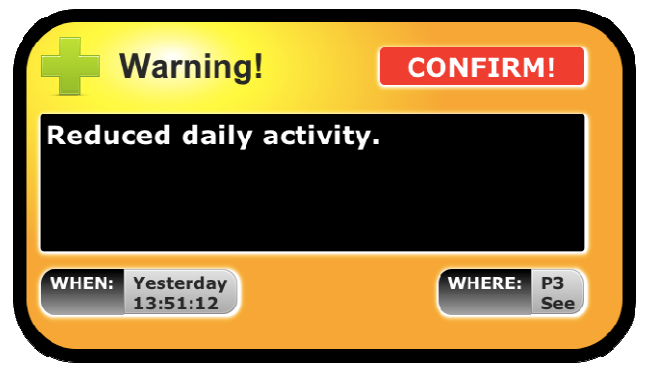


Fig. 11 Examples of windows for different priorities of the alerts

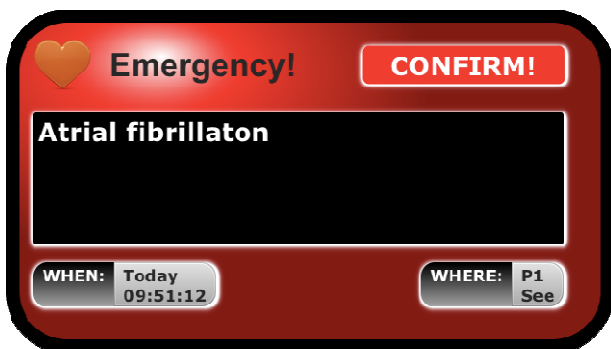
A user can get precise localization of the alert source (button “WHERE”) and should confirm the message (the information is stored together with the recipient role). When the “CONFIRM” button is pressed the user can get additionally information (which depends on the alert message), for example in case of “water leakage” the contact data to the appropriate service can appear.

V.CONCLUSION

The design of multimodal platform for communication, training and patient monitoring at home was presented.

In our work we want to solve three aspects. First, we want to support social connections between elder parents and their adult children or colleagues and friends preventing against digital and social exclusion. These persistent connections will convey activity in the respective homes as well as trends over time. Second, we hope to support "everyday cognition" by augmenting those aspects of memory that decline with age and planning capabilities of elder residents. Third, we also plan to sense and identify potential crisis situations so that appropriate outside services can be contacted as needed.

The presented multimodal platform infrastructure is an excellent chance to obtain general information about a user while at home, and a wearable computer can gather data wherever the user may go outside. The home can contain a large amount of computation and infrastructure for sensing at a distance, while a wearable has the advantage of immediate and intimate contact with the user. The data gathered on the wearable might then be filtered and released to the environmental infrastructure as appropriate. On the other



hand, the wearable may draw on the house's data resources to cache important information for the mobile user when away from the house. Thus, an automated wireless collaboration between the platforms seems appropriate, with the user placing limits on the type and level of information transferred between his personal and environmental infrastructure. We will develop such infrastructure interactions and explore some of the technical and social benefits.

The very important aspect (especially for older people on retirement) of the system is the cost. We assumed to build implementations of all platform elements using no expensive solutions. The already implemented modules are using free software and relatively inexpensive hardware elements.

One possibility is to optimize the GUI of applications and design special virtual keyboards (with limited set of key). Different options of HCI are required for different group of patients. The new audio commands processing subsystem is required (modified version of the general purpose system present in Microsoft Vista/7). Finally the combination of visually and audio guided interfaces could be more comfortable for many patients.

Dedicated multi sensors for special group of patients are still under designing. The final goal is the possibility to collect a platform from building blocks according to the given patient needs. That is why the proper system design is so important.

Another open problem is respecting people/patients privacy and creates a highly secure system in a house and in external communications. Some previous works were published [19][20] but this is still an open subject. The system is still under development and up to now we don't have a evaluation results of usage the system by older people. In laboratory condition system works reliably but still with limited functionality. We will plan to evaluate prototyping system in situ in the nursing home and in the home of older person.

#### ACKNOWLEDGMENT

Author thanks the other members of the project team: prof. J. Wtorek, A. Meresta, M. Madej, A. Polinski, A. Palinski and D. Meller, L. Wolnik and PhD students: M. Bajorek, T. Kocejko, M. Lewandowska, M. Mazur-Milecka, M. Moderhak.

#### REFERENCES

- [1] P. Binkley, Predicting the Potential of Wearable Technology, IEEE Eng. in *Medicine and Biology Magazine*, vol. 22, No. 3, pp. 23-24, 2003.
- [2] E. Jovanov, A. Milenkovic, Ch.Otto C, P. Groen, A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation, *Journal of NeuroEngineering and Rehabilitation* 2005, 2:6, pp. 1-10, 2005.

- [3] Proceedings of the 1st International Workshop on Wireless Sensor Networks for Health Care, WSNHC 2007.
- [4] Y. Jasemian, Elderly comfort and compliance to modern telemedicine system at home, *Proceedings of Second International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth 2008*. Jan. 30 2008-Feb. 1 2008, pp.60 – 63, 2008.
- [5] Crossbow, Crossbow IRIS, [http://www.xbow.com/Products/Product\\_pdf\\_files/Wireless\\_pdf/IRIS\\_Datasheet.pdf](http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/IRIS_Datasheet.pdf), 2008.
- [6] Sun, SPOT - Small Programmable Object Technology, <http://www.sunspotworld.com/docs/>, 2008.
- [7] U. Maurer, A. Rowe, A. Smailagic, D.P. Siewiorek, eWatch: a wearable sensor and notification platform, *Proc. of the Int. Workshop on Wearable and Implantable Body Sensor Networks*, 4 pages, 2006.
- [8] L. E. Holmquist, F. Mattern, B. Schiele, P. Alahuhta, M. Beigl and H. W. Gellersen. Smart-Its Friends: A Technique for Users to Easily Establish Connections between Smart Artefacts, *Proc. of UBICOMP 2001*, Atlanta, GA, USA, Sept. 2001.
- [9] Wikipedia, Sensor node, [http://en.wikipedia.org/wiki/Sensor\\_node](http://en.wikipedia.org/wiki/Sensor_node), 2009.
- [10] Shnayder V., Chen B., Lorincz K., Fulford-Jones T., Welsh M., Sensor Networks for Medical Care, *Harvard University Technical Report TR-08-05*, April 2005.
- [11] van Halteren, A. T. and Bults, R. G. A. and Widya, I. A. and Jones, V.M. and Konstantas, D., Mobihealth-Wireless body area networks for healthcare, Wearable eHealth Systems for Personalised Health Management, *Studies in Health Technology and Informatics* Vol. 108, pp. 121 - 126,2004.
- [12] Ng J. W. P., Lo B. P. L., Wells O., Sloman M., Toumazou Ch., Peters N., Darzi A., Yang G. Z., Ubiquitous Monitoring Environment for Wearable and Implantable Sensors (UbiMon),*Proc. of the International Conference on Ubiquitous Computing (UbiComp)*, 2 pages, Sep 2004.
- [13] Dong-Oh Kang, Hyung-Jik Lee, Eun-Jung Ko, Kyuchang Kang and Jeunwoo Lee, A Wearable Context Aware System for Ubiquitous Healthcare, *Proceedings of the 28th IEEE EMBS Annual International Conference*, New York City, USA, Aug 30-Sept 3, 2006.pp. 5192-5195.
- [14] Papamathaiakis G., Polyzos G.C., Xylomenos G., Monitoring and Modeling Simple Everyday Activities of the Elderly at Home, *Proceedings of the CCNC*, 2010, 5 pages
- [15] Rodden T., Crabtree A., Hemmings T., Koleva B., Humble J., Åkesson J., Pär Hansson K. P., Configuring the Ubiquitous Home, *Proceedings of the 6th International Conference on the Design of Cooperative Systems*, 2004, 16 pages
- [16] Gaver I. W., Sengers Ph., Kerridge T., Kaye J., Bowers J., Enhancing Ubiquitous Computing with User Interpretation: Field Testing the Home Health Horoscope, *CHI Proceedings 2007*, April 28–May 3, 2007, San Jose, California, USA, pp. 537-546
- [17] Kidd C. D., Orr R., Abowd G. D., Atkeson Ch. G., Essa I. A., MacIntyre B., Mynatt E., Starner T. E., Newstetter W., The Aware Home: A Living Laboratory for Ubiquitous Computing Research, 9 pages
- [18] Popescu M., Florea E., Linking Clinical Events in Elderly to In-home Monitoring Sensor Data: A Brief Review and a Pilot Study on Predicting Pulse Pressure, *Journal of Computing Science and Engineering*, Vol. 2, No. 1, March 2008, Pages 180-199
- [19] H S Ng, M L Sim and C M Tan, Security issues of wireless sensor networks in healthcare applications, *BT Technology Journal*, Vol. 24, No. 2 April 2006, pp. 138-144, 2006.
- [20] Y. Jasemian, Security and privacy in a wireless remote medical system for home healthcare purpose, *Proceedings of Pervasive Health Conference and Workshops*, 2006, Nov. 29 2006-Dec. 1 2006, pp. 1 – 7, 2006.

## Design of a wearable sensor network for home monitoring system

Eliasz Kańtoch  
AGH University of Science and  
Technology, 30 Mickiewicza Ave.  
30-059 Kraków, Poland  
Email: kantoch@agh.edu.pl

Joanna Jaworek  
AGH University of Science and  
Technology, 30 Mickiewicza Ave.  
30-059 Kraków, Poland  
Email: jaworek@agh.edu.pl

Piotr Augustyniak  
AGH University of Science and  
Technology, 30 Mickiewicza Ave.  
30-059 Kraków, Poland  
Email: august@agh.edu.pl

**Abstract**— In this paper we describe a wearable ubiquitous healthcare monitoring system that integrates electrocardiogram (ECG device) and an accelerometer sensor with a mobile device in a Bluetooth-based body surface network (BSN). Our research focused on the right connection of the hardware units, combination of the detection of QRS complexes, calculation of heart rate (HR) and the detection of human falls. The main aim of this research was the early detection of abnormal situations (high/low HR, a fall) and the heart rate variability analysis. The human falls are very risky events that occur not only in the elderly people's daily living but also by epileptics and asthmatics. For these people independent living is strictly forbidden. A wearable sensor based monitoring system can inhibit serious injuries and allow those people live an independent life. Additionally, an SMS messaging module was integrated with the monitoring system. After detecting a non-standard situation a short notice is sent. The implementation of the QRS complex detection algorithm, that was based on the Tompkins formula, was tested on the records from the MIT-BIH database.

### I. INTRODUCTION

RECENT advances in biomedical engineering, wireless network and computer technologies have enabled the possibility of remote patient monitoring. Based on these technologies, it is possible to improve patient care, chronic disease management, and promote lifelong health and well-being for the ageing population.

The aim of this paper is to provide an overview of the authors experience in constructing and implementing wearable sensor network for home monitoring. Our project and research are based on a wearable ubiquitous healthcare monitoring system that integrates 12-leads ECG signal transmitter (ASPEKT 500), an accelerometer sensor and a mobile device in a Bluetooth-based body surface network (BSN).

### II. HARDWARE UNITS

ASPEKT 500 is a digital unit designated for wireless ECG signal transmission to PC or mobile device (fig. 1). It is manufactured by Aspel [8]. The transmitter allows a free patients movement up to 10 m from receiver. Small dimensions and weight make the examination more comfortable for a patient.



Fig. 1 ASPEKT 500 - Wireless ECG signal transmitter [8]

ASPEKT 500 - Wireless ECG signal transmitter is equipped with ten electrode cable. In order to receive 12-leads (Einthoven, Goldberger, Wilson) we need to connect electrodes as shown in the fig. 2.

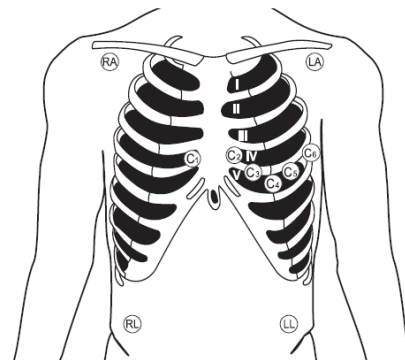


Fig. 2 Electrodes location on the patient [8]

ECG data is sampled at 500 Hz frequency. Battery operated time is about 12 hours.

### III. METHODS

The most important aim in the monitoring system is the detection of the QRS complexes in real-time. This gives us the opportunity to calculate the heart rate and observe the heart rate variability. The data processing is crucial to extract the correct part of the signal, the QRS complex. The ECG waveform contains also P, T, sometimes U waves and a lot of noise (60 Hz power line noise, EMG, motion artifacts) [3].

One of the most popular and often cited QRS detection algorithms that works in the time domain is the Pan and Tompkins algorithm that was proposed in 1985 [1, 2, 3]. The QRS detection algorithm is based on analysis of the slope, amplitude and width of the QRS complex which refers to the depolarization of the right and left ventricles. Figure 3 shows a block diagram of the Pan-Tompkins algorithm which will be hereafter described.

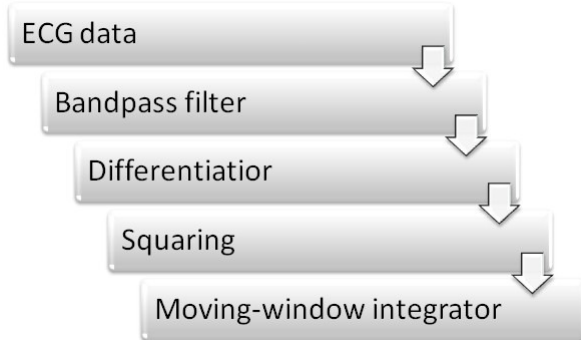


Fig. 3 Diagram of the Pan-Tompkins algorithm [1, 2].

The bandpass filter contains of a lowpass filter and a highpass filter. The lowpass filter has a cutoff frequency of 17 Hz and the highpass filter has a cutoff frequency of 5 Hz. After filtering the signal is differentiated. The derivative operator suppresses the low-frequency elements (P and T waves) and picks the high frequencies out (QRS complex). This operation gives us information about the QRS complex slope. The next step is squaring operation which makes all values positive. This enables us to analyze each channel. The squaring operation also emphasis the higher frequencies and makes the QRS complex interpretation easier. After the squaring operation multiple peaks are observed within the QRS complex. To eliminate this the moving-window integration filter is being performed. To detect QRS complexes an adaptive thresholding is applied. New peak is marked when a local maximum is find during before defined period. After the QRS detection algorithm the heart rate is calculated.

One of the project goals is to detect a fall. Fall detection could be achieved by analyzing the accelerometer data. The most common method of detecting a fall is calculating the absolute sum of ACC signal in different direction as shown in fig. 4.

$$FALL = \sqrt{ACC_X^2 + ACC_Y^2 + ACC_Z^2}$$

Fig. 4 FALL parameter

Thresholding FALL value is used to alert a fall.

#### IV. SYSTEM ARCHITECTURE

The monitoring system prototype has been build and analyzing software has been implemented in order to check if it is possible to monitor human activity remotely. The wireless ECG recorder transmits data to mobile device via wireless network based on Bluetooth technology. Data is analyzed by

implemented software and forwarded using general packet radio service to Internet database, where medical data is accessible to a doctor. Fig. 5 presents the system design.

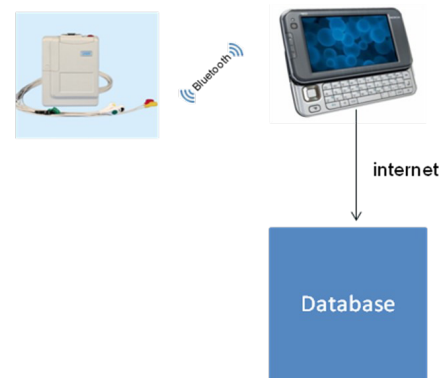


Fig. 5 Monitoring system design.

Implemented software is organized into 4 modules: ECG signal analysis, Fall detection, Heart Rate Variability and Database (fig. 6).

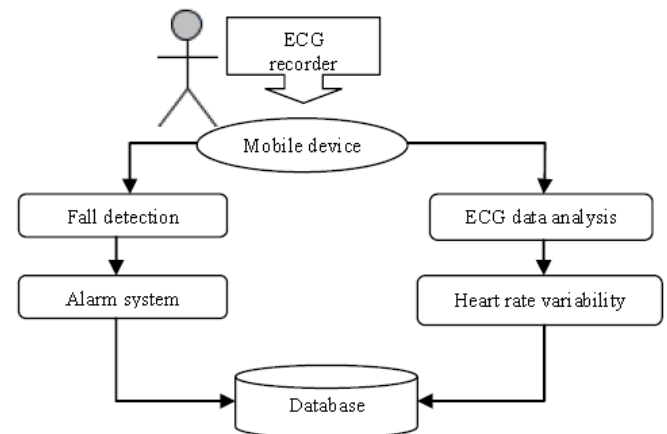


Fig. 6 Monitoring system architecture.

The most important module of the monitoring system is the ECG data analysis module which analysis the incoming signal data from the ECG recorder. The ECG data analysis module executes the Pan and Tompkins algorithm that was described in section III. The output of this module is transmitted to the Heart rate variability module. This part of the system calculates the HRV and sends the results to the Database via GPRS.

Fall detection module is responsible for analyzing data from accelerometer and calculating FALL factor described in section III. If the fall is detected the alarm module is switched on. The alarm module is a combination of two methods. The first one is a signal bell which can be switched off during 10 seconds. If isn't accomplished a text message is send to a trusted friend.

#### V. TESTS AND RESULTS

One of the most important stages in the implementation of the monitoring system is its verification. Tested has been the



QRS detection algorithm using ECG signals from the MIT-BIH arrhythmia database [5] and fall detection algorithm. Fig. 7 presents four cases of QRS detection possibilities.

QRS detection	Absence of QRS complex	QRS complex occurs
Detected	FP (false positive)	TP(true positive)
Undetected	TN (true negative)	FN (false negative)

Fig. 7 QRS complex detection possibilities.

The effectiveness of the application was performed by calculating the four cases mentioned in fig. 7 for every file from the database. The MIT-BIH database contains signals with different abnormalities (very noisy signal, high P wave, high T wave, arrhythmias and noise that is similar to QRS complex).

The most common used equation to demonstrate the effectiveness of the QRS detection algorithm is the sensitivity which equals TP times all QRS complexes (FN+TP). Fig. 8 presents results for selected files.

Signal nr.	Total QRS complex	TP	Sensitivity
100	2273	2273	100 [%]
101	1865	1863	99,8 [%]
105	2572	2541	98,6 [%]
113	1795	1711	95,2 [%]
117	1535	1498	97,4 [%]

Fig.8 Results for the MIT-BIH database files.

The proposed method of detecting falls did not give satisfactory results in performed tests. It often provided with false positive detections especially during making rapid movements or stops. Further research is needed in order to elaborate more accurate algorithm.

## VI. CONCLUSION AND FUTURE WORKS

The achieved results are satisfactory for our monitoring purposes. However, more tests are needed on a representative group to assess system performance. In the future works we plan to develop more advanced algorithms of detecting non-standard situations and improve fall detection algorithm.

The proposed healthcare monitoring system can help to monitor health conditions (heart rate, heart rate variability) and support elderly, sick and disabled people in their independent living.

## ACKNOWLEDGMENT

This work was supported by AGH University of Science and Technology in Cracow as a research project no. 11.11.120.612.

## REFERENCES

1. Tompkins W.J., Pan J., A real-time QRS detection algorithm, Biomedical Signal Analysis, IEEE Press, Vol. BME-32, NO. 3, 1985
2. QRS detection algorithm at <http://enel.ucalgary.ca/People/Ranga/enel563/Lab8.pdf>
3. Augustyniak P., Tadeusiewicz R. (2009) Ubiquitous cardiology. Hershey, New York
4. Rangayyan R. M. (2010), Pan-Tompkins algorithm to detect QRS complex in ECG signal, Biomedical Signal Analysis, IEEE Press at <http://www.doestoc.com/docs/22491202/Pan-Tompkins-algorithm-algorithm-to-detect-QRS-complex-in-ECG>
5. ECG signal records at <http://www.physionet.org/physiobank/database/mitdb>
6. Huang D., Lin P., Fei D., Chen X., Bai O. (2009), Decoding human motor activity from EEG single trials for a discrete two-dimensional cursor control, J. Neural Eng. 6, 046005 (12pp),
7. IEC 60601-2-51. (2003), Medical electrical equipment: Particular requirements for the safety, including essential performance, of ambulatory electrocardiographic systems, First edition 2003-02, International Electrotechnical Commission, Geneva,
8. Aspel ASPEKT 500 technical documentation: <http://www.aspel.com.pl/index.php?lang=pl&pg=372>



# Measuring Pulse Rate with a Webcam – a Non-contact Method for Evaluating Cardiac Activity

Magdalena Lewandowska

Gdansk University of Technology  
 ul. Narutowicza 11/12,  
 80-233 Gdansk, Poland  
 Email:  
 maglew@biomed.eti.pg.gda.pl

Jacek Rumiński

Gdansk University of Technology  
 ul. Narutowicza 11/12,  
 80-233 Gdansk, Poland  
 Email: jwr@biomed.eti.pg.gda.pl

Tomasz Kocejko

Jędrzej Nowak  
 Gdansk University of Technology  
 ul. Narutowicza 11/12,  
 80-233 Gdansk, Poland  
 Email: kocejko@gmail.com  
 jedrzej.nowak.tch@gmail.com

**Abstract**—In this paper the simple and robust method of measuring the pulse rate is presented. Elaborated algorithm allows for efficient pulse rate registration directly from face image captured from webcam. The desired signal was obtained by proper channel selection and principal component analysis. A developed non-contact method of heart rate monitoring is shown in the paper. The proposed technique may have a great value in monitoring person at home after adequate enhancements are introduced.

## I. INTRODUCTION

HOME health care is nowadays growing and changing discipline. The remote monitoring of vital signs includes not only the high accuracy diagnostic devices but also simple ones and accessible for everyone. One of the most frequent examinations performed in health care monitoring is cardiac pulse measurement. There are many different methods of contact measurement of a heart rate among which the golden standard is an electrocardiography (ECG). However, recording electric potential generated by the heart requires appropriate application of the electrodes what may be too complicated and annoying in home conditions. Other methods of measuring cardiac pulse involve thermal imaging [1], Doppler phenomenon both optical [2] and ultrasonic [3] or piezoelectric measurements [4]. Photoplethysmography (PPG) is another method that is being used in detecting pulse rate [5]. It utilizes changes of the optical properties of a selected skin area involved by pulsating blood contents. The typical implementation of PPG uses dedicated light sources, e. g. near-infrared light. Changes of the light intensity reflected from the skin correspond to a volume of tissue blood perfusion. Moreover, it has been proved that pulse measurement from human face is also possible using daylight as the illumination source [6]. *Poh et al.* has developed a robust method for computation of the heart rate from digital color video recordings of the human face [7]. The method is based

This work was partly supported by European Regional Development Fund concerning the project: UDAP0IG. 01.03.01-22-139/09-00 -“Home assistance for elders and disabled – DOMESTIC”, Innovative Economy 2007-2013, National Cohesion Strategy.

on blind source separation of the color channels into independent components.

A principal component analysis is used in our procedure. It is applied to video channels and in effect it reduces computational complexity in comparison to independent component analysis. We also show that it is possible to determine a pulse rate based on small rectangular region of the face image and only on two color channels. It is important when considering computational efficiency of the home health care monitoring system and its operation in real time.

## II. METHODS

### A. Experimental setup and measurement procedure

The experimental setup consisted of web and thermographic cameras, and synchronized recorder of ECG (Fig. 1).

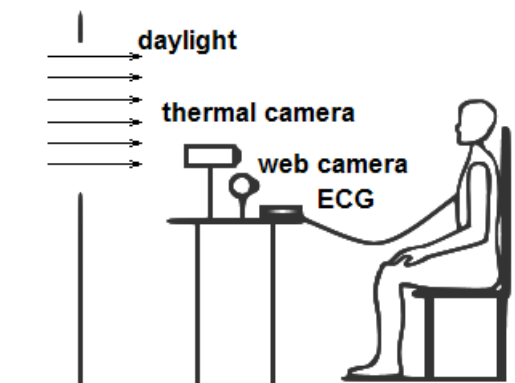


Fig. 1 Experimental setup

The measurements were performed indoors and the only light source was sunlight. The 30 seconds long video sequences were recorded by means of a Logitech Webcam 9000 Pro. The resolution of the videos was 640x480 pixels and the frame rate was 20 fps. Sequences of images were saved in AVI format without compression. While the video

was being recorded the ECG signal was collected using the AsCARD electrocardiograph (AsCARD MrGrey v.201, Aspel). The sampling rate of ECG signal was 400 Hz.

Together 10 white volunteers, 2 women and 8 men, of different age (20 - 64 years), were examined. During the measurements they were sitting still and distant 1 m in front of the camera. Experiments were performed in room naturally lighted at midday (Fig. 1).

### B. ROI's selection

The analysis was performed for two different ROI's (regions of interest) sizes. First, the rectangle containing the face region was selected at the first frame of video recording (Fig. 2a). Coordinates of the selected face region remained the same for the whole sequence of images. The second ROI was a rectangular-shaped part of the forehead area. It was defined basing on pupils' coordinates and distance calculated between them (Fig. 2b). It was expected that it would be possible to find a part of forehead to be visible "uniformly". To prove this assumption thermographic images were taken for all examined persons. Thermograms were recorded by the FLIR ThermoCam SC3000 thermal camera having temperature resolution equal 0.1°K.

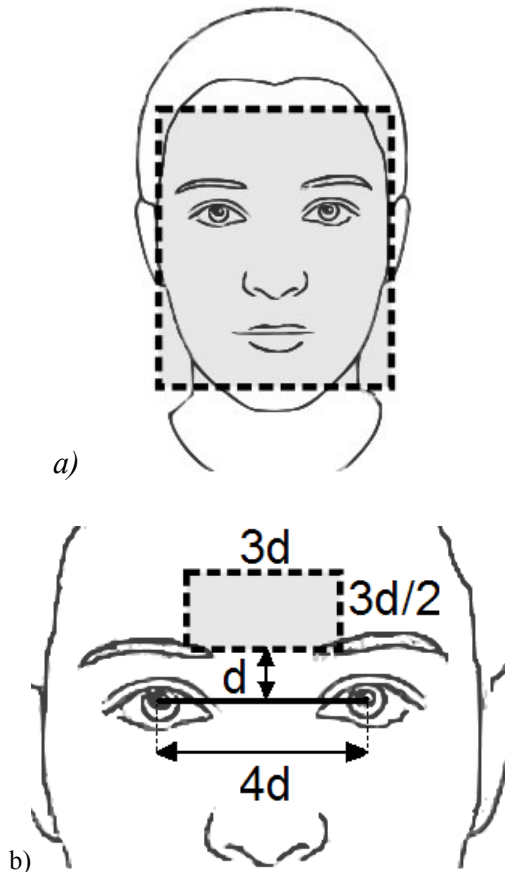


Fig.2 Definition of ROI's for two approaches utilized in the paper, a) the whole face as a ROI, b) selected part of the forehead as a ROI

### C. Image decomposition

The ROI's were decomposed into three RGB channels. Analyses were performed for a different channels combination: RGB, RG, GB, RB.

### D. Methods of analysis

The Independent Component Analysis (ICA) is a statistical and computational technique used to separate independent signals from a set of observations that consist of linear mixtures of the underlying sources [8]. The ICA model assumes that the observed signals  $\mathbf{y}(t)$  are linear mixtures of the unknown sources  $\mathbf{x}(t)$ :

$$\mathbf{y}(t) = \mathbf{A}\mathbf{x}(t) \quad (1)$$

where the mixing matrix  $\mathbf{A}$  is also unknown. To estimate both  $\mathbf{A}$  and  $\mathbf{x}(t)$  we assume that the components of vector  $\mathbf{x}$  are statistically independent and nongaussian. After estimating  $\mathbf{A}$  matrix its inverse  $\mathbf{W}$  (demixing matrix) can be computed. Then the independent components can be obtained:

$$\mathbf{x}(t) = \mathbf{W}\mathbf{y}(t) \quad (2)$$

To evaluate the demixing matrix many algorithms have been proposed. In the present study the FastICA algorithm was used [9].

The Principal Component Analysis, (PCA), is sometimes called the Karhunen-Loeve Transformation, which is a technique commonly used for data reduction in statistical pattern recognition and signal processing. The PCA is a transformation that identifies patterns in data, and expresses the data in such way that it highlights the similarities and differences. That makes PCA a powerful tool for analyzing data [10].

The basic idea in PCA is to find the components  $s_1, s_2, \dots, s_N$  so that they explain the maximum amount of variance possible by  $N$  linearly transformed components. The principal components are then given by  $s_i = w_i^T \cdot x$ . The computation of the  $w_i$  can be accomplished by using the covariance matrix  $E\{xx^T\} = C$ . The vectors  $w_i$  are the eigenvectors of  $C$  that corresponds to the  $N$  largest eigenvalues of  $C$ .

These components should be ordered in such way that the first component,  $s_1$ , points in the direction where the inputs have the highest variance. The second component is orthogonal to the first and points in the direction of highest variance when the first projection has been subtracted, and so forth.

Suppose we have two zero mean random vectors,  $\mathbf{X}$  and  $\mathbf{Y}$ , that gives  $E[\mathbf{X}] = 0$  and  $E[\mathbf{Y}] = 0$ . Let  $\mathbf{u}$  denote a unit vector, onto which the  $\mathbf{X}$  is to be projected. This projection is defined by the inner product of the vectors  $\mathbf{X}$  and  $\mathbf{U}$ , as shown by

$$\mathbf{Y} = \mathbf{U}^T \mathbf{X} \quad (2)$$

where  $\mathbf{U}$  is an orthonormal matrix. The principal components are columns of  $\mathbf{U}$  and they are found by seeking the directions of maximum data variance, under the orthogonality constraint. Columns of  $\mathbf{U}$  are eigenvectors of the covariance matrix ordered with decreasing variance [11].

### E. Algorithm

For every ROI's channel, pixels values were added separately for each frame. The signals obtained this way were filtered using a FIR bandpass filter (0.5–3.7 Hz, 32-

point Hamming window, designed with MATLAB FDATool). Next, independent and principal component analyses were performed. The ICA was performed using FastICA algorithm implemented in MATLAB. Principal components were obtained with the use of MATLAB *processpca* function.

### III. RESULTS

The images of examined volunteers were processed so as to find regions of interest to further analysis (Fig. 3).

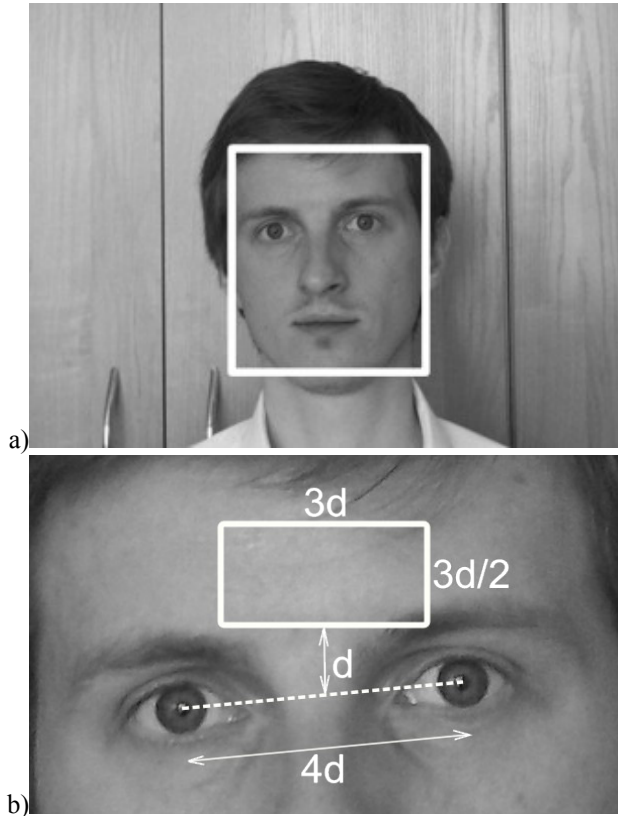


Fig. 3 Selection of the analyzed region: a) whole face ROI, b) forehead ROI

The uniformity of ROI localized on the forehead was analyzed using thermal images. The mean pixel value and the variance  $\sigma^2$  were calculated for face and forehead ROI. Examples of obtained images with calculated  $\bar{x}$  and  $\sigma^2$  are shown in Fig. 4. The images support the assumption that it is possible to select almost uniform part of face to be analyzed.

Video sequences were processed in order to obtain a desired time - dependent signals. A sum of pixels values was calculated for each channel throughout the ROI (Fig. 5). Obtained signals were bandpass filtered (0.5–3.7 Hz).

Independent component analysis and principal component analysis were conducted for different sets of data (Fig. 6 - 7). Firstly, analyses were performed for the ROI containing whole face area, and then a rectangular shape ROI selected on the forehead center was analyzed. Signals from three channels R, G and B were an input to ICA and PCA.

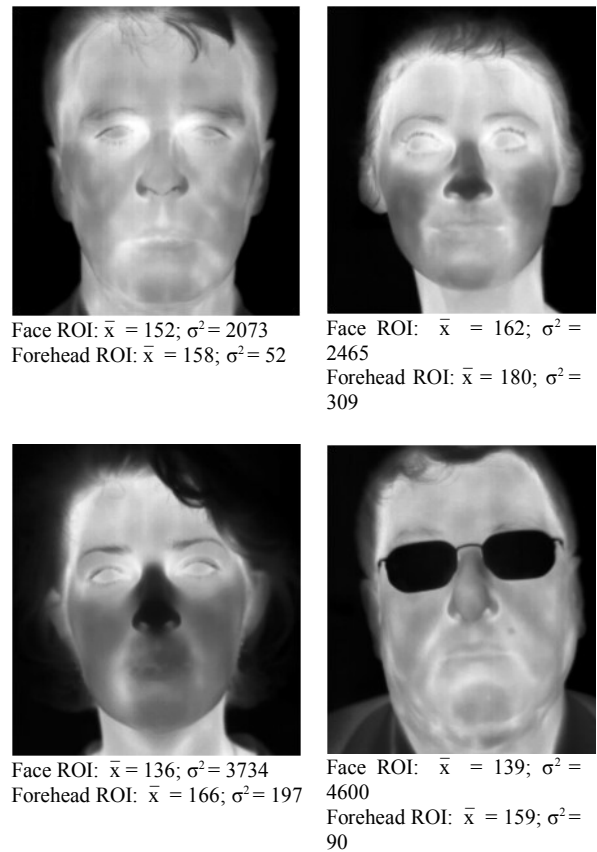


Fig. 4 Temperature distribution of the face for selected persons examined in the study

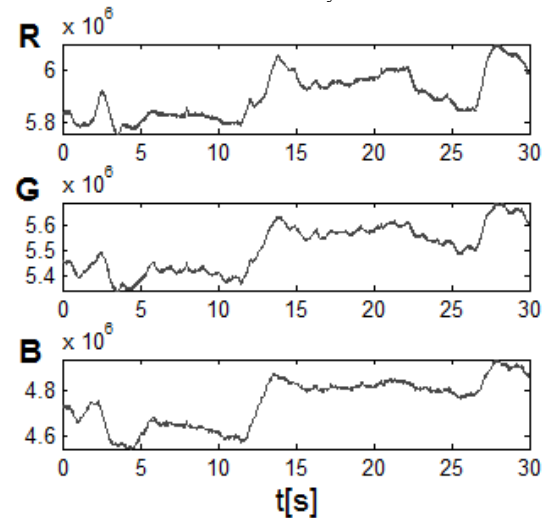


Fig. 5 Sum of pixels values of video sequence for whole face ROI; R, G and B stands for RGB channels

The result of PCA was compared with that obtained by ICA (Fig. 8). Time of calculation for ICA performed on the face ROI was equal to 223 ms, while for forehead ROI 94 ms. The time of calculation when using PCA for the same data set was respectively 1.4 ms and 1.2 ms.

To check whether obtained “pulse” signal changes are related to the heart rate the results obtained from PCA were compared with an ECG signal that was measured during video recordings (Fig. 9).

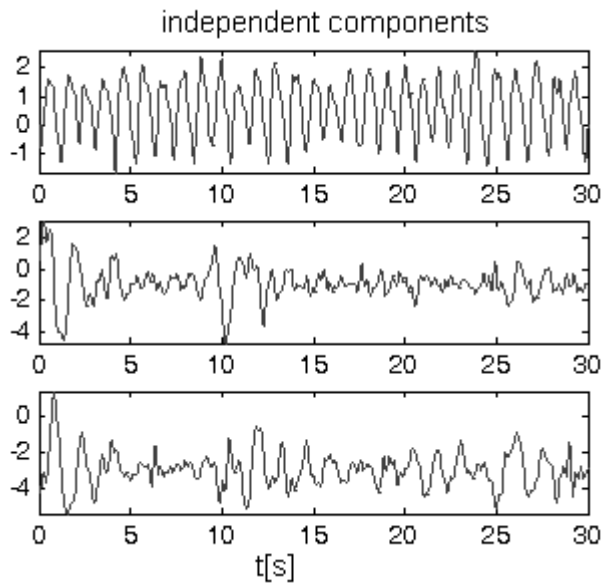


Fig. 6 Bandpass filtered result of independent component analysis for three RGB channels, ROI – the whole face

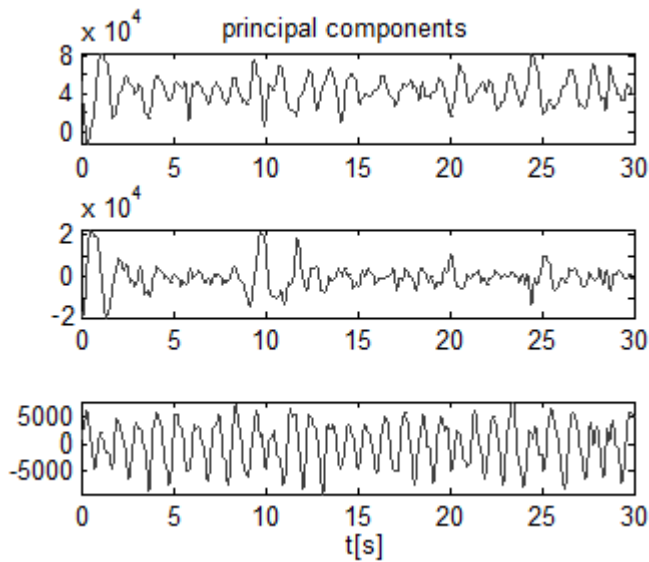


Fig. 7 Bandpass filtered result of principal component analysis for three RGB channels, whole face ROI

In Table I mean heart rates of four selected patients are presented. Results obtained from webcam measurements are compared to that obtained from ECG (R-R interval). Mean heart rate was calculated using two methods: one based on the interval between positive slope zero-crossings of the 2<sup>nd</sup> or 3<sup>rd</sup> PC and the second using maximum of the power spectral density function. Results obtained with the use of the second method are the same for face and forehead ROI in all cases.

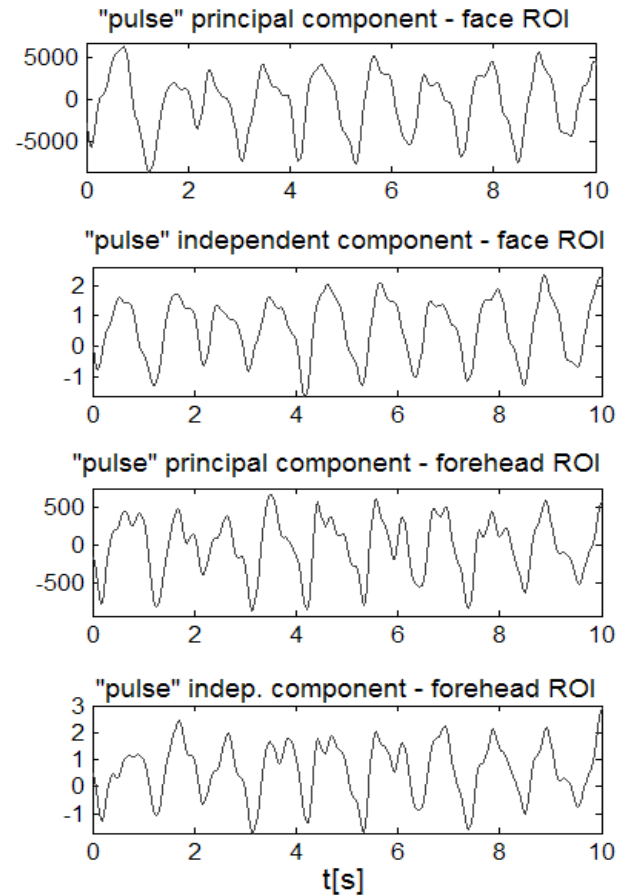


Fig. 8 Comparison of results obtained for ICA and PCA for the same data set

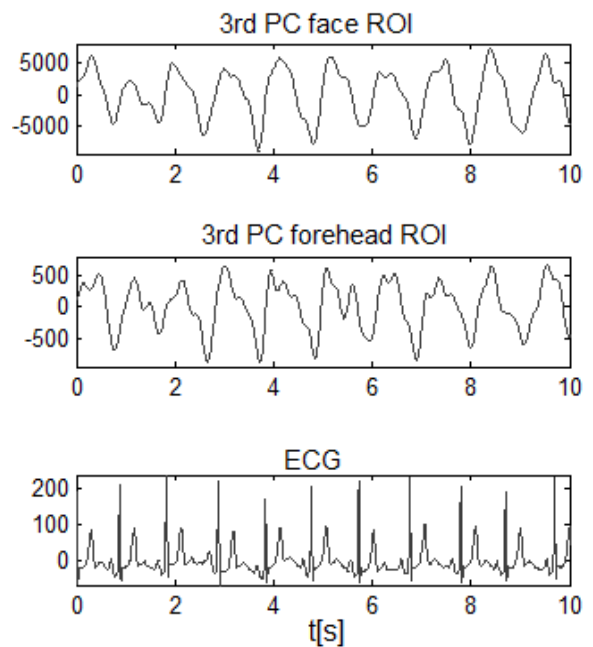


Fig. 9 Third principal component for two different ROI's: face and forehead, compared to ECG signal; it can be noticed that components of registered video signals are changing with heart rate



TABLE I.  
HEART RATES OF FOUR SELECTED PATIENTS

Mean heart rate [bpm](measurement time = 30s)				
Patient Id	Web camera			ECG
	zero crossing		fft	
	face ROI	forehead ROI		
1.	87.56	100.67	91.41	87.89
2.	59.32	59.18	58.01	58.59
3.	94.84	104.35	98.44	99.61
4.	60.07	76.85	59.76	58.66

When analyzing signals from only two channels, detection of a pulse rate was most effective when RG combination was considered (Fig. 10 - 11). However, compared to the analysis using three channels the “pulse” component (2<sup>nd</sup> PC) was more distorted, especially when forehead ROI was analyzed (Fig. 11). Spectra of principal components containing pulse signals are presented in Fig. 12. Pulse component extracted from R and G channels apart from 1 Hz component contains also frequencies from 0.3 – 1 Hz range.

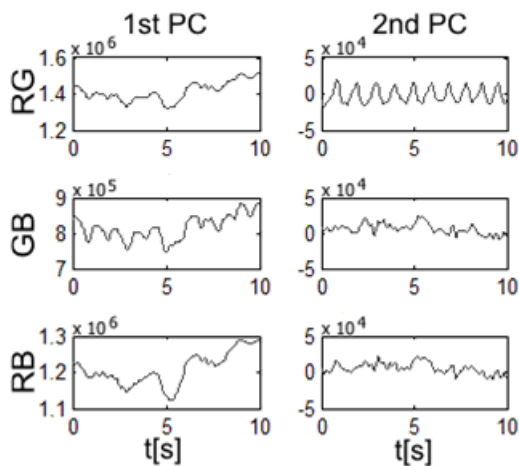


Fig. 10 Principal components for different channel combination: RG, GB, RB; face ROI

#### IV. DISCUSSION

Using the methods detailed in Section III, we estimated pulse rate based on webcam recordings. To reduce both complexity and number of calculations a smaller ROI was selected (the rectangular region of the forehead) as well as fewer color channels and less complex method of analysis was chosen.

The results obtained from independent and principal component analysis' show that those two methods extract the “pulse” component with similar accuracy. However, comparison of time of calculation and other studies [12]

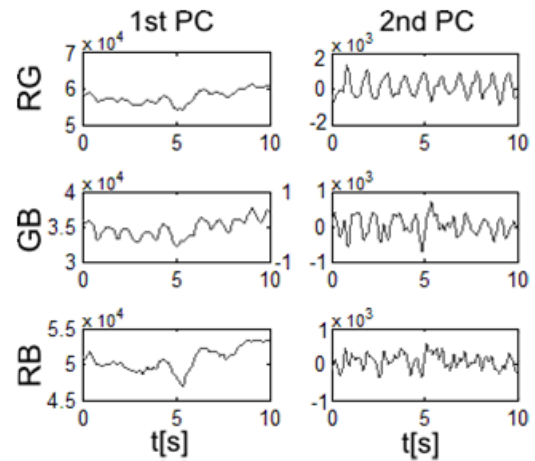


Fig. 11 Principal components for different channel combination: RG, GB, RB; forehead ROI

allows for the conclusion that PCA is less computationally complex so it is reasonable to choose that method to reduce time and complexity of analysis. The obtained results show that when only pulse rate is considered there is no need to use the more computationally complex ICA method. The accuracy of extraction of cardiac pulse signal by PCA is comparable to that obtained by ICA and is sufficient for our purposes. Moreover, the time of calculation obtained for PCA suggests possibility of using this approach in real-time applications.

Selected forehead rectangular area is a region of the face with a temperature relatively constant for healthy male and female of different age. As shown in Fig. 4 the selected part of the forehead area, in contrast to other parts of the face, e.g. nose, has almost the same temperature for examined subjects. Furthermore, assuming that the examined object is kept still it does not contain any “moving” elements, like blinking eyes or moving lips. Based on the analysis performed on the forehead ROI it was possible to determine the pulse rate with sufficient accuracy for the selected group of patients. The results indicate that the selected forehead ROI is representative for the whole face region.

As it is shown in Fig. 10 and Fig. 11 the selected R and G channels contained most of the information about color changes corresponding to the blood volume pulse. So it was possible to reduce the number of analyzed signals from three to two. Decrease in ROI's size or number of channels increases the level of noise (Fig. 12). However, taking into account that these measurements are destined for daily monitoring of vital signs not for clinical purposes, the high accuracy is not the most important factor.

To improve the accuracy of the presented algorithm it is important to provide proper experiment conditions. The object of research needs to be motionless, so the examination have to be performed in the sitting or supine position. The proper lightening conditions are also very important because when analyzed face images are under- or overexposed information about color changes may be lost.



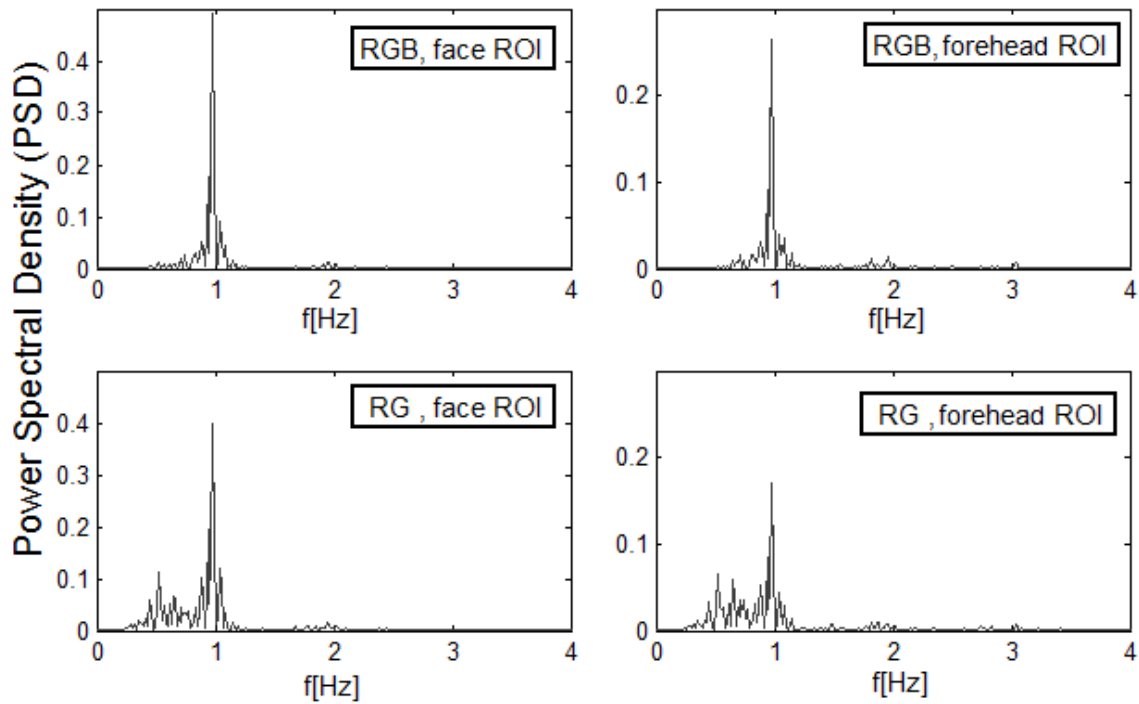


Fig. 12 Spectral density of two different ROI's (face and forehead) and two different channel combination (RGB and RG)

## V. CONCLUSIONS

A simple processing of image data and then applying PCA allows extracting the changeable component containing information of the heart rate. The presented algorithm seems to be quite effective and easy to use in the daily monitoring of home care patients. However, a further study has to be performed on moving persons and the same with more than one camera.

## REFERENCES

- [1] M. Garbey, N. Sun, A. Merla, and I. Pavlidis, "Contact-free measurement of cardiac pulse based on the analysis of thermal imagery," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 8, pp. 1418–1426, Aug. 2007
- [2] S. Ulyanov and V. Tuchin, "Pulse-wave monitoring by means of focused laser beams scattered by skin surface and membranes," *Proc. SPIE*, vol. 1884, pp. 160–167, July 1993
- [3] D. W. Holdsworth, C. J. Norley, R. Frayne, D. A. Steinman and B. K. Rutt, "Characterization of common carotid artery blood-flow waveforms in normal human subjects," *Physiol Meas.*, vol. 20(3), pp. 219–40, Aug. 1999
- [4] K. Aminian, X. Thouvenin, Ph. Robert, J. Seydoux and L. Girardier, "A piezoelectric belt for cardiac pulse and respiration measurements on small mammals," *Engineering in Medicine and Biology Society, 1992 14th Annual International Conference of the IEEE*, vol. 6, pp. 2663–2664, Oct. 29 1992–Nov. 1 1992
- [5] J. Allen, "Photoplethysmography and its application in clinical physiological measurement" *Physiol. Meas.*, vol. 28(3), pp. R1–39, Mar. 2007
- [6] W. Verkrusse, L. O. Svaasand, and J. S. Nelson, "Remote plethysmographic imaging using ambient light," *Opt. Express*, vol. 16, pp. 21434–21445, Dec. 2008
- [7] M. Z. Poh, D. J. McDuff, and R. W. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation," *Opt. Expr.*, vol. 18, pp. 10762–10774, May 2010.
- [8] A. Hyvärinen, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, pp. 411–430, June 2000
- [9] A. Hyvärinen, "Survey on independent component analysis," *Neural Comput. Surveys*, vol. 2, pp. 94–128, 1999
- [10] J. F. Cardoso, "Blind signal separation: Statistical principles," *Proc. of the IEEE*, vol. 86, no. 10, pp. 2009–2025, Oct. 1998
- [11] V. M. Asunción, P. O. Hoyer and A. Hyvärinen, "Equivalence of some common linear feature extraction techniques for appearance-based object recognition tasks," *IEEE Trans Pattern Anal Mach Intell.*, vol. 29(5), pp. 896–900, May 2007
- [12] M. H. Yang, "Kernel Eigenfaces vs. kernel Fisherfaces: face recognition using kernel methods", *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 215–220, 2002

## Pulse pressure velocity measurement – A wearable sensor

Mateusz Moderhak  
Gdansk University of Technology  
ul. Narutowicza 11/12 80-233  
Gdansk, Poland  
Email:  
matmod@biomed.eti.pg.gda.pl

Mariusz Madej, Jerzy Wtorek  
Gdansk University of Technology  
ul. Narutowicza 11/12 80-233  
Gdansk, Poland  
Email: {marmad,  
jaolel}@biomed.eti.pg.gda.pl}

Bart Truyen  
Vrije Universiteit Brussel  
Pleinlaan 2  
B-1050 Brussel  
Email: btruyen@etro.vub.ac.be

*Abstract*—Pulse pressure velocity measurements (PPV) may be a source of useful information on artery state. A 2007 guideline of the European Society of Hypertension recommends assessing arterial stiffness in patients with arterial hypertension, by measuring the PPV. Mechanical changes in the cardiovascular tree involved in the blood ejection have been measured at the thorax level and on the wrist using impedance techniques, in combination with a one channel electrocardiographic signal. Performing impedance measurements on the wrist is a very challenging task because of the very low conductivity changes in combination with the relatively high value of the basal impedance. This is especially a problem, when the dimensions of the measuring probe are required to be adequate for integration into a whole day wearable sensor. In this study, it has been shown that such measurements are indeed possible, and that they may deliver very useful information on pulse pressure velocity when compared to the classical approach based on the measurement of the PPV delay in relation to the ECG signal. However, it has been found that a significant discrepancy may exist between results obtained using the classical approach and the impedance measurement technique proposed here.

**Keywords:** Pressure pulse velocity, impedance technique, body sensors

### I. INTRODUCTION

**P**ULSE pressure wave velocity (PWV) measurements allow evaluation of arterial stiffness. The pressure wave following the ejection of blood by the heart is gradually conveyed to the periphery. Close to the heart the wave velocity is of the order of 5 m/s and gradually increases towards the periphery and then again decreases in very small arteries. However, only the mean value of the wave velocity over a given segment is considered, while the vessel wall characteristics are highly site dependent. On the other hand, pulse wave velocity assessment has the advantage that no blood pressure information (measured invasively) is required. Several studies have shown that wave velocity measurements allow assessment of the haemodynamic profile of a patient as a function of age and pathophysiological conditions [1], [2]. The European

Society of Hypertension recommends in its guidelines evaluation of this parameter in patients with arterial hypertension. It is also suggested that carotid-femoral pulse wave velocities greater than 12 m/s are an indicator of organ damage [1].

In general, apart from geometric non-uniformity, the changing elastic properties and viscous components make the structure of the artery much more complicated. E.g. for the aorta, stiffness is increasing with the distance away from the heart, which is accompanied by increased damping properties due to differential compositions of the wall.

Different approaches for PWV measurement may be applied [1], [6], while in most of them the pulse pressure is measured at two distant points separated by a known distance, and the propagation delay between the two signals is estimated. This time delay is determined between two “corresponding” points on the pressure wave. Measurements should be performed at stable conditions, as the propagation of the pressure wave depends on many factors, including the pressure itself. The difference between results obtained by applying two different approaches in estimation of the PPV is examined in this paper. A first approach is based on the time delay between the R wave and a selected point of the pulse, while the second one is based on the measurement of the time delay between two impedance signals, one measured on the thorax and another at the wrist.

### II. METHODS

Impedance measurements have been already applied in evaluation of blood flow in artery and venous system. However, the most widespread method, called impedance plethysmography, is based on occlusion of venous return in selected part of a limb (a segment) and measurement of impedance change following the change of segment volume [11]. Pairs of measurement electrodes are localized distantly when comparing to limb diameter in this application. It is in contradiction to the presented application. On the other hand, it is known phenomenon that configuration of electrode matrix (a probe) influences a shape of the recorded signal [11].

---

This work was partially supported by the European Regional Development Fund in frame of the project: UDA-POIG.01.03.01-22-139/09-02 -“Home assistance for elders and disabled – DOMESTIC”, Innovative Economy 2007-2013, National Cohesion Strategy.

### A. Theoretical backgrounds

Sensitivity of impedance measurements to conductivity changes localized in examined segment can be evaluated by relation  $S = \nabla \phi_n \cdot \nabla \varphi_n$ , where  $\phi_n$  and  $\varphi_n$  are normalized potential distributions associated with unit current flowing respectively between “current” and “voltage” electrodes. The latter one is a hypothetical one. The impedance change involved by conductivity changes is described by the following relationship [3]

$$\Delta Z = - \int_V \Delta \sigma(x, y, z) \nabla \phi_n \cdot \nabla \varphi_n dv \quad (1)$$

where  $\Delta \sigma(x, y, z)$  - spatially distributed change of conductivity. The relationship (1) allows calculation of impedance change ( $\Delta Z$ ) associated with conductivity change ( $\Delta \sigma$ ) undergone in the subject. A geometry, conductivity distribution and the electrode matrix configuration are reflected in spatial distribution of potentials ( $\phi_n, \varphi_n$ ).

According to the relationship (1) a value of the measured impedance change depends on a volume of the conductivity change. Thus, it is important to estimate how volume changes of an artery reflect pulse pressure propagation along it.

The wave propagation velocity, based on theoretical model in [4], [5], is described by the relationship

$$c^2 = \frac{Eh}{2\rho(1-\mu^2)r_0} \left[ 1 - \frac{2}{ar_0} \frac{J_1(ar_0)}{J_0(ar_0)} \right] \quad (2)$$

where  $c$  is the pulse pressure velocity,  $E$  the modulus of elasticity,  $\mu$  the Poisson's ratio,  $J_0(ar_0)$  and  $J_1(ar_0)$  Bessel functions of the first kind, and  $a^2 = -j\omega\rho/\eta$ , where  $\rho$ ,  $\omega$  and  $\eta$  denote the blood density, circular frequency, and blood viscosity, respectively.

For inviscid fluids,  $\eta = 0$ ,  $a \rightarrow \infty$  the second term in square brackets in eq. 1 becomes zero

$$\frac{2}{ar_0} \frac{J_1(ar_0)}{J_0(ar_0)} = 0. \quad (3)$$

As a result the following relationship is obtained:

$$c^2 = \frac{Eh}{2\rho(1-\mu^2)r_0}. \quad (4)$$

Setting  $\mu = 0$ , (3) reduces to the well-known Moens-Korteweg equation:

$$c_0^2 = \frac{Eh}{2\rho r_0}. \quad (5)$$

The above relationship has been obtained assuming that [4]:

1. The wall material is homogeneous, elastic, isotropic, and follows Hooke's law.
2. The relative variations of the dimensions are small.
3. Variation of length is not possible.

4. There exists only a radial motion; so there is no rotation (in other words rotational symmetry is assumed).
5. The thickness  $h$  of the wall tube is small, as compared with the inner radius  $r_0$ , at the mean pressure in the tube.

However, in the actual measurements the above assumptions are not fulfilled exactly. In spite of this, it is generally accepted that pulse velocity measurement is essential in evaluation of cardiovascular state [6]. The relationship (2) can be rewritten in the following form [7]

$$c = \frac{c_0}{(X - jY)(1 - j\omega W)} \quad (6)$$

where viscous effects of the blood on phase velocity, ( $c$ ), are taken into account by  $X - jY$  term and the second term in the denominator of equation (6) takes into account the effects of wall viscosity as well as a complex Poisson ratio.

### B. Measurement technique

The pressure wave or its function has been measured at the thorax and wrist using two different sensors, both being based on the impedance technique. The thorax sensor is described elsewhere [8]. Fig. 1 shows the electronic circuit of the wrist sensor, which consists of a current source built around two operational amplifiers, that delivers a current of 0.1 mA<sub>pp</sub> at 20 kHz. The high output impedance of the current source ensures a constant current for changes of load impedance up to 10 kΩ. These specifications have been realized by means of an active feedback design.

The voltage arising from the current flowing between electrodes  $I_1$  and  $I_2$  is measured by means of voltage electrodes  $V_1$  and  $V_2$ , and amplified 50X. In the next stage, a programmable amplifier allows to select four levels of amplification, such as to ensure a maximum SNR ratio.

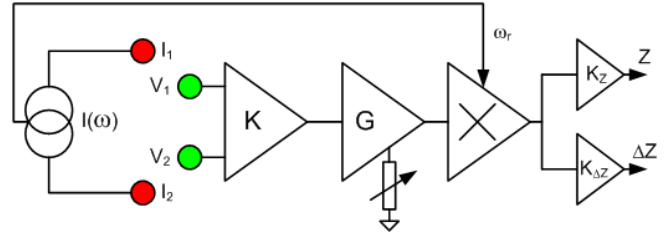


Fig. 1. Schematic diagram of measurement circuit,  $I(\omega)$  – current source,  $K$ ,  $G$ ,  $K_Z$ ,  $K_{\Delta Z}$  – gain of respective stage of bandpass amplifier, mark  $\times$  stands for synchronous demodulator

The amplified signal is then passed to a synchronous demodulator, built around another operational amplifier. The maximum input amplitude is determined by the linear range of the synchronous demodulator.

After demodulation the signal is split between two channels, acting as low-pass and pass-band filters, respectively. The former delivers a signal proportional to the value of the basal impedance  $Z$ , depending on the average electrical and the geometrical properties of the wrist. The amplifier marked  $K_Z$  in Fig. 1, is a low pass filter with a corner frequency equal to 2 Hz. The time-dependent signal,  $\Delta Z$ , reflects the changes of blood volume in the measured segment of the body, and in

general is synchronous with the blood induced conductivity changes in the wrist segment of the hand. The amplifier  $K_{\Delta Z}$ , instead acts as a pass-band filter with middle frequency equal to 11 Hz, a low corner frequency below 0.3 Hz, and high corner frequency equal to 20 Hz. For the middle frequency, this two-stage amplifier has a constant gain equal to 200.

C. Model and realization of the sensor's probe

It was assumed that the sensor should be in the form of a watch with the four electrodes mounted on the watch strap. To allow calculation of the sensitivity function for different sensor configurations, a FEM model of the limb segment was built.

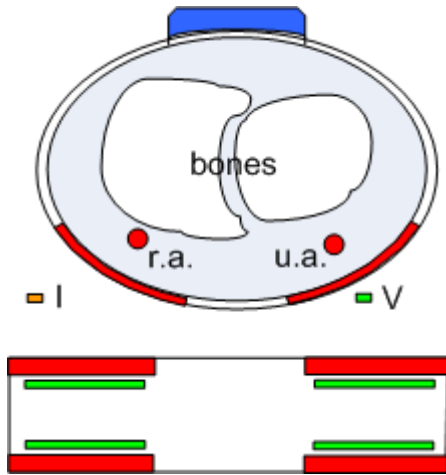


Fig. 2. Schematic representation of a cross-section of the wrist. Construction of the four electrode sensor (current injecting electrodes marked red, voltage measuring electrodes marked green); r.a. indicates the radial artery, while u.a. stands for ulnar artery.

This model was in the form of a cylinder of finite length with four circumferential electrodes. Two of them (marked by letter I) were used for current injection, while the other two (marked V) serve as voltage measurement electrodes. Several variants, differing in geometry, were examined both theoretically and experimentally.

First, a sensor consisting of classical circumferential electrodes was examined for equally spaced electrodes. Then, the voltage electrodes were moved closer to the current electrodes, so that these are no longer equally spaced, but with the external dimensions preserved. This latter sensor configuration is shown in Fig. 2.

D. Signal analysis

The recorded signals were analyzed manually after being collected and imported into Microsoft Excel. Time relations between the impedance signals were calculated and assessed according to the definitions specified in Fig. 3. Although only one time course of the impedance signal is illustrated in Fig. 3, the same parameters were calculated for the impedance signals recorded on the upper thorax and the wrist. The effect of the arm position is known to influence both the arterial and venous pressure [9], [10], and was closer examined in our experiments.

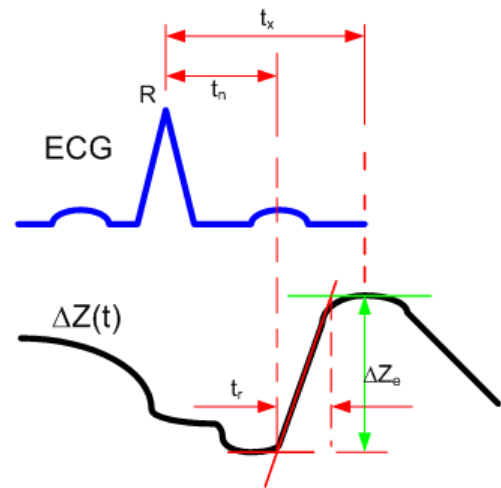


Fig. 3. Definition of time parameters and amplitude of the signal:  $t_n$ , the time delay between R wave in the ECG and the beginning of impedance signal  $\Delta Z_e$ ,  $t_x$ , the time delay between the R wave and the maximum of impedance signal  $\Delta Z_e$ , and  $t_r$ , the rise time of the signal.

Pressure waves were recorded for the arm hanging freely down the body, kept at the heart level, and raised straight above the head.

III. RESULTS

It can be assumed, from relationship (2) that the pulse velocity is larger than 5m/s for the considered arteries (starting from the ascending aorta and finishing at the ulnar or the radial arteries). Moreover, assuming that heart rate is 1 Hz (in fact it is a little bit higher) a wavelength of generated pressure wave of frequency 1 Hz is around 5 m. Higher harmonics are of course respectively shorter.

Sensitivity values were calculated for a circumferential electrode configuration applied to a cylinder of radius R and height  $\pm 10 R$ .

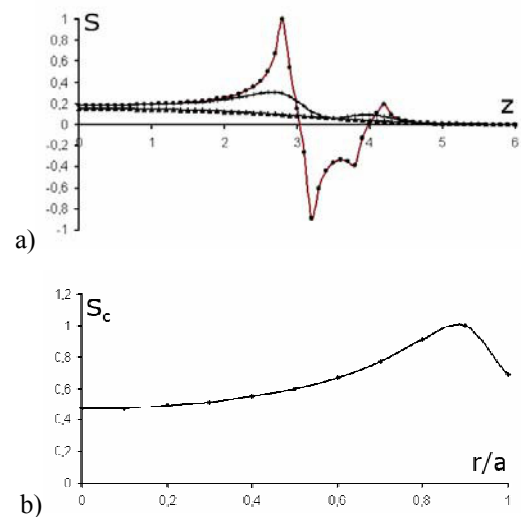


Fig. 4. Local sensitivity function (a) for conductivity changes at 0.5R (filled triangles), 0.9R (crosses) and 0.99R (filled squares), where R is the radius of the cylinder and b) total sensitivity,  $S_e$ , calculated as the sum (integral) of sensitivity S along coordinate "z" for different value of r. An anisotropic model, with anisotropy ratio equal to 2, was assumed.

Anisotropy of the conductivity was assumed, i.e. the ratio  $\sigma_l/\sigma_t = 2, 4, \text{ etc.}$ , with  $\sigma_l$  and  $\sigma_t$  the longitudinal and transfer conductivity, respectively. The voltage electrodes were placed at  $\pm R$ , with the current electrodes positioned at  $\pm 1.3 R$ . Sensitivity values for a conductivity changes localized at  $0.5 R, 0.9 R$  and  $R$  respectively are shown in Fig. 4a. The integral sensitivity (calculated as a sum of sensitivity for a certain depth) is also dependant on artery localization in the segment (Fig. 4b).

The developed sensor was found to allow measurement of impedance changes less than  $15 \text{ m}\Omega$  in magnitude, with the channel bandwidth of the impedance measurements limited to  $20 \text{ Hz}$ , while the basal impedance value ranged from  $10 \Omega$  up to  $60 \Omega$ . However, the impedance measured between "current" electrodes belonged to range  $(200 \pm 20\,000) \Omega$  and strongly depended on types of electrodes used.

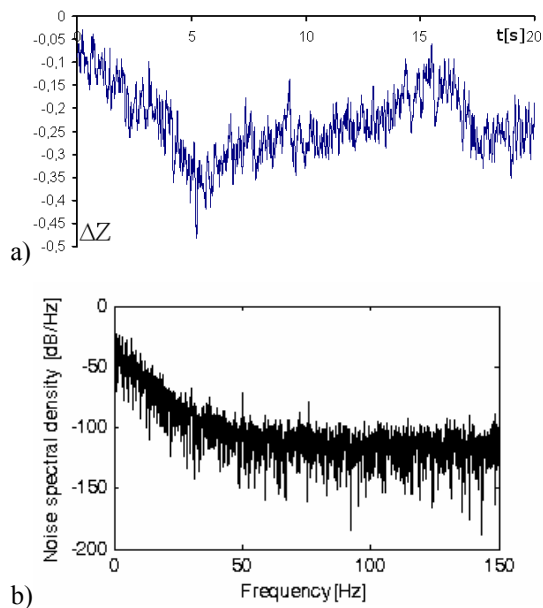


Fig. 5. (a) Noise of the  $\Delta Z$  channel recorded on a resistance of  $1 \text{ k}\Omega$ , (b) resulting power density.

The noise level of the sensor was estimated by means of measurements recorded on a  $1 \text{ k}\Omega$  resistor (Fig. 4). The signals recorded using the thorax and wrist sensors are presented in Fig. 6 and 7, respectively. All signals have been acquired for standing person with the hand hanging freely down the body.

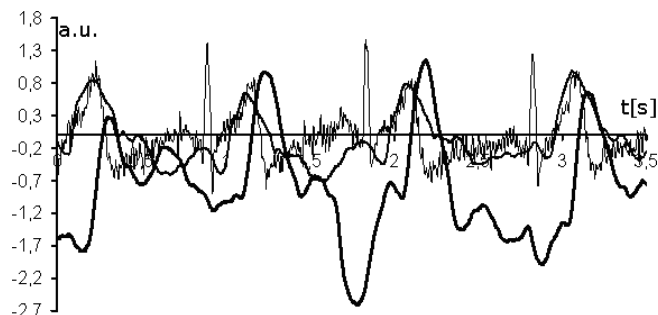


Fig. 6. Recorded signals of electrical heart activity and impedance changes on the upper part of the thorax and the wrist. The signals are marked in more detail in the following figures

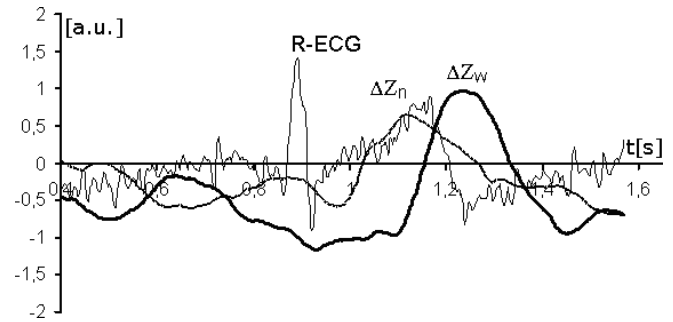


Fig. 7. One cycle of the signals presented in Fig. 6.  $\Delta Z_n$  denotes the impedance change recorded on the upper part of thorax, while  $\Delta Z_w$  is the impedance change recorded at the wrist. The hand was kept hanging down.

All signals presented in Fig. 8 have been obtained with the wrist kept at the level of the heart. Parameters of the signal measured on the thorax are changed slightly in comparison to previous one.

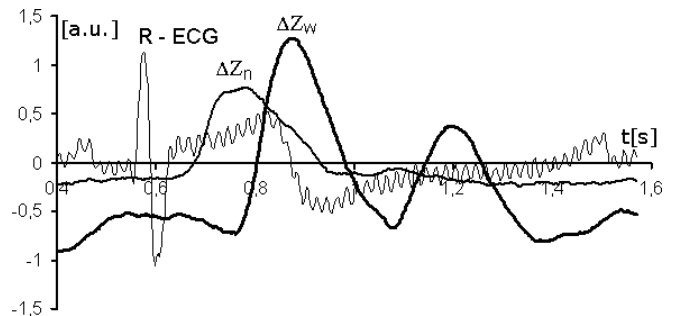


Fig. 8. One cycle of recorded signals with the hand kept at the level of the heart,  $\Delta Z_n$  denotes the impedance change recorded on the upper part of thorax, while  $\Delta Z_w$  is the impedance change recorded at the wrist.

The most significant and noticeable changes are observed for the signal recorded at the "wrist," rather than for the "cardiac" signal recorded on the thorax. However, when raising the hand above the head, both impedance signals show changes (Fig. 9). An enhanced wave coinciding with the QRS complex of the ECG has appeared, probably reflecting the activity of the atria.

Differences in time delay were evaluated for different positions of the arm, i.e. freely hanging along the body, kept at the level of heart, and held straight above the head. Time delays were calculated between the ECG and impedance signals measured on the wrist (Fig. 10), as well as between the impedance signals measured on the thorax and at the wrist (Fig. 11).

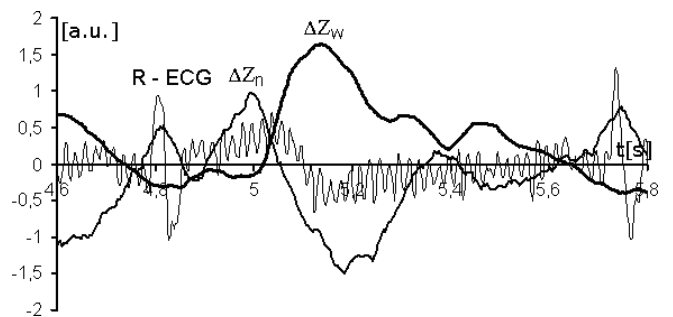


Fig. 9. One cycle of recorded signals with the hand kept above the head,  $\Delta Z_n$  denotes the impedance change recorded on the upper part of thorax, while  $\Delta Z_w$  is the impedance change recorded at the wrist.



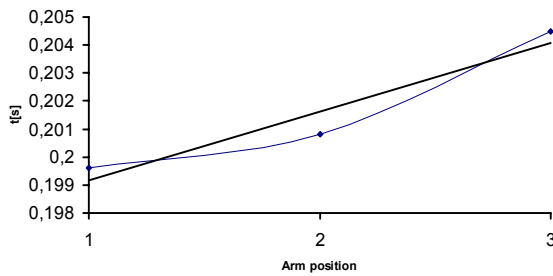


Fig. 10. Time delay between the R wave of the recorded ECG starting point of  $\Delta Z_w$  signal, for three positions of the arm: (1) arm hanging freely down the body, (2) arm kept at the level of the heart, and (3) arm hold straight above head. Indicated in bold is the trend line.

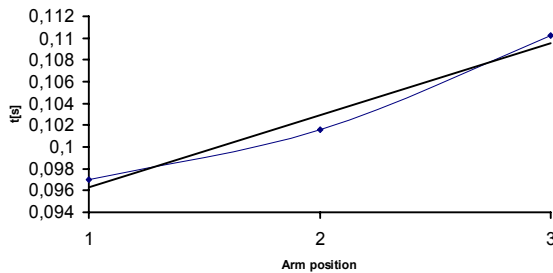


Fig. 11. Time delay between the starting the point of the impedance signal measured on the thorax, and the starting point of  $\Delta Z_w$  as measured at the wrist, for three positions of the arm: (1) arm hanging freely down the body, (2) arm kept at the level of the heart, and (3) arm hold straight above head.

The distance between the heart and the wrist is relatively short so that different measurement methods may yield large discrepancies. The differences observed between the two methods were indeed quite large, often separated by almost a factor 2.

#### IV. DISCUSSION

Recently, the measurement of large artery stiffness, as a factor indicating the development of cardiovascular complications, has become one of the most essential parameters in patients with cardiovascular diseases such as arterial hypertension [1]. It is assumed that the measurement of pulse pressure velocity can contribute useful information on the state of the cardiovascular system.

Typically, impedance plethysmography has been applied to relatively big segments, e.g. lower or upper limb, with the electrode separation assumed bigger than the radius of the segment being measured. In the current application, however, the distance between the electrodes is less than or comparable with the radius of the segment (the distance between current electrodes was equal to 4 cm), which results in significantly higher sensitivities for superficial than deep conductivity changes. With both arteries taken into account, the radial and the ulnar being superficial, this could be considered as rather advantageous effect. However, the sensitivity function of the probe becomes bipolar, especially for regions close to the surface, as seen in Fig. 4a. This effect can decrease the total sensitivity for superficially located arteries (Fig. 4b). In order to minimize the region of negative sensitivity, voltage and current electrodes were

brought close together. The effect can be eliminated completely when a two-electrode technique is utilized. Unfortunately, as two-electrode techniques are known to be very susceptible to motion artefacts they cannot be used in the current application. The proposed measurement probe consists of pairs of electrodes, each containing of a voltage and current electrode positioned closely to each other. The separation between electrode pairs instead should be as large as possible, such as to ensure that larger magnitude impedance signals are recorded.

Further studies are needed to investigate the origin of the recorded signals when utilizing electrical impedance measurements. As the pulse is travelling along the artery, the shape of the recorded signal is influenced by many factors, also by measurement procedures and equipment used. However, assuming the pressure wave velocity of 5 m/s, and a distance between the voltage electrodes equal to 6-8 cm, the time pulse propagation between electrodes is less than 8 milliseconds. So, in a first approximation this effect may be neglected, and the signal source can be assumed arising from a uniform change of volume within the artery. This uniform radial expansion of artery results from response to transmural pressure (difference between pressure inside and outside the artery). It follows from a comparison of the maximal distance between electrodes, equal to  $\sim 8$  cm, with wavelength of the dominating harmonics of pulse wave in vascular tree. In fact, instead of distance between electrodes the length of the artery participating in measurement should be considered. This length can be evaluated using the data presented in Fig. 4. As it is shown this length depends on the artery localization (depth) in the segment. In spite of this, it can be assumed that the impedance measurements reflect changes of the local artery diameter.

According to the relationships (2) and (6) pulse wave velocity is determined both, by the properties of fluid (blood) and vessels. Analysis of these relationships allows the following conclusions: for a large artery, and consequently a large value of the term  $|a|r_0$ , the phase velocity,  $c$ , approaches the phase velocity in an inviscid fluid. On the other hand, for very small values of  $|a|r_0$  the phase velocity becomes very small. Thus, when estimating the artery's properties from the phase velocity measurements certain precautions must be taken. Moreover, some properties of artery's wall depend also on pressure. Thus, its value should be included in estimation procedure.

The recorded signals are also affected by conditions of measurement (Figs. 7-9). Position of the hand, in relation to the heart level, influences essentially shape of impedance signals, both recorded on the thorax and the wrist. However, this does not change significantly time relations between recorded signals. However, noticed a change in time delay between R wave and  $\Delta Z_n$  also can be considered as a "natural" response of cardiovascular system. Note, that slopes of regression lines presented in Figs. 10 and 11 are different. It seems that further experiments should also include other measurement method allowing precise determination of ejection time from left ventricle.

The proposed measurement method generally is known as impedance plethysmography, and is sensitive to conductivity changes undergoing within the electrical field created by the current flowing between the current electrodes, as well as to the geometrical relation between the voltage and current electrodes, and the shape of the examined body segment [11].

It must be underlined that this type of measurements also is susceptible to artefacts arising from wrist movements. This was also the reason to divide the probe into two parts, each located closely to the corresponding artery. Unfortunately, muscles and tendons are also in close proximity to the arteries.

It is observed from the results that the difference between the velocity estimated from measuring the time delay between the R wave of the ECG signal and the pulse pressure at the wrist, and that estimated from the time delay between the two impedance signals, is almost 100 %. In the former method, the time of isovolumetric contraction of the chambers is included. Taking into account that this time is comparable with the time of pulse propagation along the arm, this explains largely the measurement discrepancy. This effect may be amplified in persons having vessels affected by arteriosclerosis, as such vessels demonstrate a higher pulse propagation speed. Heart insufficiency may further increase the time of isovolumetric contraction, and also cause larger measurement variances.

Taking into account the effect of different arm positions on the recorded signal, it seems that supplementary information obtained by means of an accelerometer would be useful.

Despite providing a more precise estimation of the pulse velocity propagation along the arteries in arm, the proposed method also has its drawbacks. As the magnitudes of the conductivity changes are very low, the very high gain required to recover a reasonable and interpretable signal, makes the measurement system susceptible to interferences and artefacts. The total gain used in the proposed system such as to obtain the presented signals, ranged from 2000 up to 5000. Additional artefacts may also arise from changes in geometry of the wrist and tissues involved in operating and moving hand.

## V. CONCLUSIONS

It has been shown that measuring the propagation of pulse pressure along the arm is possible using an impedance technique. Moreover, it has been shown that the conventional method, based on the simultaneous measurement of ECG and pulse, contains a large time delay component arising from the isovolumetric ventricle contraction. However, it should be underlined that the presented impedance technique may also be susceptible to measurement artefacts as a consequence of the high amplification required to obtain appropriate signal levels that allow proper parameterization and interpretation.

## REFERENCES

- [1] M. W. Rajzer, W. Wojciechowska, M. Klocek, I. Palka, M. Brzozowska-Kiszka, and K. Kawecka-Jaszcz, "Comparison of aortic pulse wave velocity measured by three techniques: Complior, Shygmocor, Arteriograph", *Journal of Hypertension*, 28, pp. 2001–2009, 2008.
- [2] Z. Marcinkevics, M. Greve, J. I. Aivars, R. Erts, A.H. Zehtabi, "Relationship between arterial pressure and pulse wave velocity using photoplethysmography during the post-exercise recovery period", *Acta Universitatis Latviensis, Biology*, vol. 753, pp. 59–68, 2009.
- [3] D.B. Geselowitz, An application of electrocardiographic lead theory to impedance plethysmography. *IEEE Trans. Biomed. Eng.*, 1971 Vol. 18, ss. 38-41.
- [4] R.H. Cox Comparison of linearized wave propagation models for arterial blood flow analysis. *J Biomech* 2: 251–265, 1969.
- [5] R.H. Cox, Wave propagation through a Newtonian fluid contained within a thick-walled, viscoelastic tube. *Biophys J* 8: 691–709, 1968
- [6] A.P.G. Hoeks, P J Brands, JM Willigers, and R S Reneman Non-invasive measurement of mechanical properties of arteries in health and disease, *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine* 213: 195–202, 1999
- [7] J.P. Matonick and J. K.-J. Li, A new nonuniform piecewise linear viscoelastic model of the aorta with propagation characteristics, *Cardiovasc. Eng.*, 1(1): 37 – 47, 2001
- [8] J. Wtorek, A. Bujnowski, M. Lewandowska, J. Rumiński, M. Kaczmarek Simultaneous monitoring of heart performance and respiration activity, *Proceedings of the 3rd IEEE Conference on Human System Interactions (HSI)*, Rzeszow, Poland, 2010, pp. 661–665
- [9] A. Mourad, S. Carney, A. Gillies, B. Jones, R. Nanra and P. Trevillian, Arm position and blood pressure: a risk factor for hypertension?, *Hum. Hypertens.* 17(6): 389–95, 2003
- [10] R.T. Netea, J.W. Lenders, P. Smits, T. Thien, Arm position is important for blood pressure measurement, *J Hum Hypertens.* 13(2), 105–9, 1999
- [11] J. Wtorek, *Electroimpedance Techniques in Medicine* (in Polish), ser. Monograph 43. Gdansk, Poland: Gdansk Univ. Technol., 2003.



# Analysis of Correlation Between Heart Rate and Blood Pressure

Artur Poliński  
Department of Biomedical  
Engineering,  
Gdansk University of Technology,  
Gabriela Narutowicza 11/12,  
80-233 Gdansk Wrzeszcz, Poland  
Email: apoli@biomed.eti.pg.gda.pl

Jacek Kot  
Department of Hyperbaric  
Medicine and Sea Rescue,  
Medical University of Gdansk,  
ul. Marii Skłodowskiej-Curie 3a,  
80-210 Gdansk, Poland  
Email: jkot@gumed.edu.pl

Anna Meresta  
Department of Biomedical  
Engineering,  
Gdansk University of Technology,  
Gabriela Narutowicza 11/12,  
80-233 Gdansk Wrzeszcz, Poland  
Email: anna@meresta.eu

**Abstract**—The paper presents an analysis of correlation between heart rate (HR) and blood pressure (BP). The actual data were obtained from three female and one male. The systolic and diastolic blood pressure was measured with the invasive method in the radial artery. The correlation coefficient indicates only linear dependence, so the inverse of HR was also taken into account. Since the measurements can be corrupted by noise the moving average filtering and trend analysis for all data was done. The results of the correlation analysis of this filtered data were similar to results obtained for raw data. The observed correlation coefficient between HR and BP (systolic and diastolic) for whole available data seems a random number. However the short-term correlation is relatively large (about 0.5), but rather unpredictable, since even sign of the correlation coefficient is changing.

## I. INTRODUCTION

OUR goal is to monitor value of blood pressure (BP) in elder people at their homes. Such measurements are made using non-invasive technique. However, such measurements are made rarely and a time interval between two successive measurements is relatively large. Thus, some serious events may be overlooked. It would be very useful and desirable to develop a method allowing continuous monitoring of BP. Such method may use information gained non-invasively, e.g. about heart rate (HR) or length of the RR interval. We would like to know if knowledge of one of these parameters allows us to estimate value of BP. We are not interested in cohort correlation between HR (or length of RR interval) and BP, rather we are interested in such correlation for individuals. We are not interesting in the relation between variability of HR and BP also. However, such models have been already proposed [1], [2]. There is a publication [3], which reports that resting HR is associated to clinic BP over the whole range of BP values and has been observed at any age. Moreover, the correlation is positive and is stronger for systolic than diastolic values of BP. The strong positive correlation between BP and HR was also reported in [4]. There is also report that exercise-induced increase in systolic BP was positively correlated with resting systolic BP, whereas the correlation of exercise-induced HR

increase with resting HR was negative [5]. The correlation between BP and HR is also reported in [6], for men and women. However, the dependence was much stronger for men than for women. We would like to verify all of these studies for applicability of using HR to predict BP values in our project devoted to home assistance for elders and disabled.

## II. MATERIALS AND METHODS

The data were obtained from monitoring of four persons: three female at age of 47, 52 and 70 and one male at age 86. Physiological data recordings were performed using the S5 DATEX/OHMEDA system for monitoring critically ill patients. Digital data were transferred by the serial port to Computer Information System developed in Department of Hyperbaric Medicine and Sea Rescue. The measurements included diastolic and systolic values of pressure. The arterial BP (measured invasively in the radial artery), central venous pressure (measured invasively in a close proximity of the right atrium of the heart) and non-invasive BP (measured from pressure oscillations of the cuff placed on the upper arm) were recorded. HR, arterial BP and central venous pressure were measured, on average, each 30 seconds, while the non-invasive BP was measured, also on average, each 30 minutes.

The correlation coefficient for HR and arterial pressure (systolic (SBP) and diastolic (DBP)) measured using invasive method was calculated. Since the correlation coefficient describes only linear dependence the calculations were repeated for inverse of HR and arterial pressure. The relation between HR and  $1/HR$  is nonlinear, thus we can observe correlation in one case and no correlation in another one.

Patients were monitored only for few days. To obtain more reliable results the invasive BP measurements were correlated with HR, since the number of samples was about 60 times larger in comparison to non-invasive BP measurements. Invasive arterial BP measurements were chosen, since their values were similar to non-invasive measurements.

Three different approaches were tested. First one was based on correlation between raw data. Since the measurements could be corrupted by noise (for example patient movements or rounding of HR and BP values to the nearest integer by measurement system) the signals were also correlated after performing a filtration procedure. A moving aver-

This work was partly supported by European Regional Development Fund concerning the project: UDAPOIG. 01.03.01-22-139/09-00 "Home assistance for elders and disabled – DOMESTIC", Innovative Economy 2007-2013, National Cohesion Strategy.

age (MA) with equal weights, thus a very simple low pass filter was used. The filter lengths were 3, 4, and 5 samples and did not introduce too much smoothing. Another approach was based on least squares approximation. The increase and decrease of BP value was considered. As an indicator of such changes the parameter  $a$  from the linear least square fitting of HR and BP ( $y=ax+b$ ) could be used. It indicated trend of signals. The fitting was done for each signal separately, and the correlation analysis was performed for  $a$  coefficients. The  $a$  coefficients were calculated for 3, 4, 5, 6, 8, and 10 successive samples. It prevented from missing short-term changes, since such short-time changes might be interested in many applications. This procedure was repeated for all successive samples similarly as in the case of MA filtration.

The HR and BP were measured regularly each 30 seconds, however in some cases the measurements were done more often, while in some cases less often. Such changes could indicate medical intervention. So the samples were divided into groups of continuous measurements done each 30 seconds. However, each group could have different length (different number of samples). We decided to make analysis taking into account groups, which length was not less than 50 samples. To check if the size of the groups was influencing the correlation coefficient, the second analysis was done for groups, which length was not less than 100 samples. The correlation coefficient  $r$  and  $p$ -value (the probability of getting a correlation as large as the observed value by random chance, when the true correlation is zero) were calculated for each group of measurement samples for all patients.

### III. RESULTS

All figures show example results of analysis for groups of samples of length not less than 100 samples. Fig. 1 shows SBP dependence on HR. The correlation coefficient was equal to 0.901 ( $p<0.001$ ). Fig. 2 shows results of MA filtration (DBP versus HR). The correlation coefficient was equal to  $-0.708$  ( $p<0.001$ ). Fig. 3 shows coefficients  $a$  for SBP dependence on coefficients  $a$  for HR. The correlation coefficient was equal to 0.669 ( $p<0.001$ ). The change of the correlation coefficient with time for SBP dependence on HR after MA filtration is shown in Fig. 4. Corresponding values of  $p$  for the correlation coefficients presented in Fig. 4 have a large distribution (Fig. 5). However, it appears that the lower value of  $p$  the higher correlation between data (Fig. 6). First 46 correlation coefficients in Fig. 6 have  $p<0.01$ , while first 48  $p<0.05$ . However, correlation is both positive and negative.

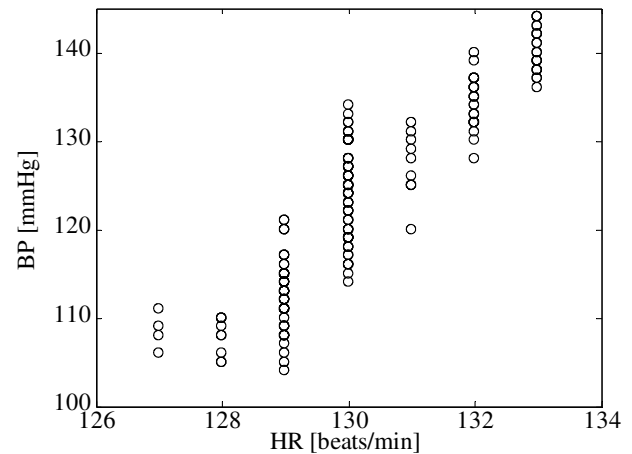


Fig. 1 SBP vs. HR

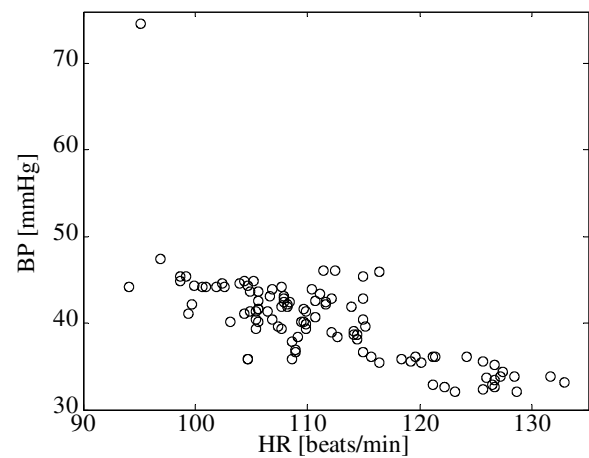


Fig. 2 DBP vs. HR after MA filtering (filter length equal to 4)

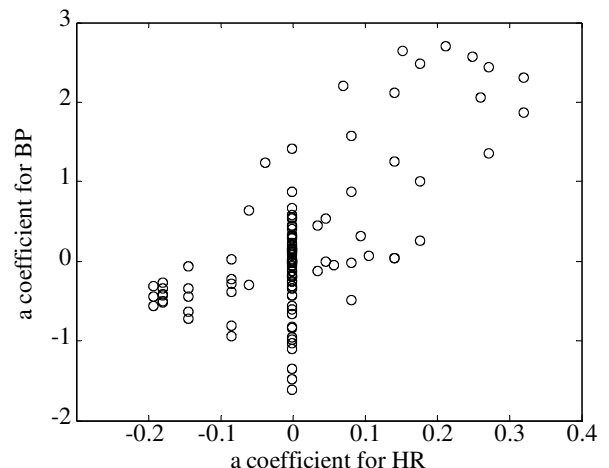


Fig. 3 Results of trend calculation for SBP vs. HR (8 successive samples taken into account)

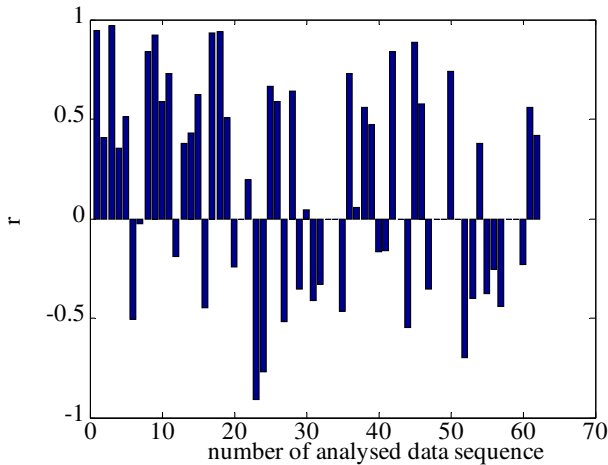


Fig. 4 The correlation coefficient  $r$  between filtered SBP and filtered HR for successive sequences. HR and SBP are processed using MA filter of length equal to 5

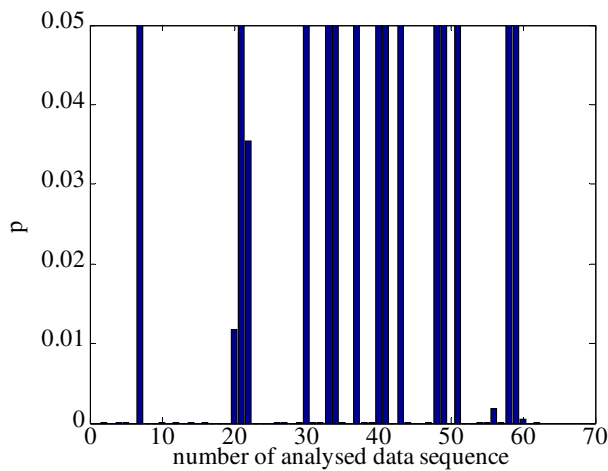


Fig. 5 The  $p$  values corresponding to the correlation coefficients presented in Fig. 4

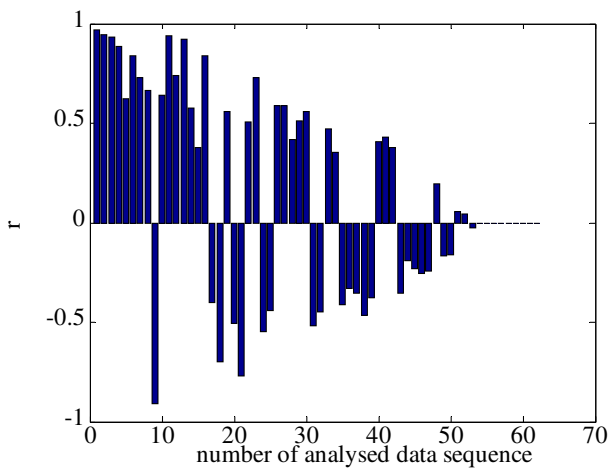


Fig. 6 The correlation coefficient  $r$  from Fig. 4 sorted from lowest to largest value of  $p$

IV. DISCUSSION

Different types of analyses were performed in the study. The properties of raw data, filtered (using MA filter of different length) were analyzed. The trend analysis for different number of samples used in calculation of coefficient “ $a$ ” was also performed. It follows from our study that correlation coefficient is not constant. The value of coefficient is accidental and different for data sequences analyzed. However, it has been found that if there is correlation between HR and BP, then such correlation with opposite sign is between  $1/HR$  and BP. It follows from the fact, that a function  $f(x)=1/x$  is nonlinear, but for relatively small interval far from 0 it can be well approximated by straight line. The obtained results show that the correlation coefficient is a good measure of dependence between HR and BP. All observed dependences were linear or no dependence was observed. No nonlinear dependence was observed.

Assuming that the sequence of data contained at least 100 samples the 20 to 60 correlation coefficients were obtained for each patient and for each type of test performed. This number was larger for shorter sequences (as sequences containing at least 50 samples were also examined). Only the correlation coefficients with  $p < 0.05$  were considered for further analysis. Because of a relatively large number of coefficients their average value was also calculated. Moreover, average value of the absolute values of the correlation coefficients and their variances for each patient and each type of study were calculated. The relative numbers of the significant ( $p < 0.05$ ) correlation coefficients were also calculated in each study. The comparison between results for SBP and DBP and for series of at least 50 and 100 samples were conducted.

First, the average of the correlation coefficients and average of their absolute values for all patients were calculated. These values were calculated using the average patient’s value. The variance between patients has also been considered and the resulting remarks from this examination are presented in the following paragraph.

The average correlation coefficient for HR and BP was about 0.2, but the difference between maximum and minimum value was even about 0.6. As can be seen in Fig. 4 there is very large variation in the correlation coefficients’ for each patient. The results for DBP had smaller variation than for SBP. It is interesting that taking into account absolute value of the correlation coefficients the average value is much higher and depends on type of data (correlated parameters). Observe that correlation coefficients having low absolute value have not been significant (Figs. 4 - 6). Absolute values of correlation coefficients were calculated for SBP versus HR and DBP versus HR basing on raw data. Then, the average values were evaluated and they were, respectively, 0.45 and 0.4. This value increased to 0.5 for SBP after applying MA filtration and remained almost constant independently on the filter length. It was in opposition to correlation between HR and DBP where the result was dependent on filter length and changed from 0.45 to 0.48. Trend analysis has shown an increase in the average of the absolute values of the correlation coefficients with number of samples

taken into account for the calculation of the  $a$  coefficient (from 0.30 to 0.40 for SBP and DBP). A little larger variance in the absolute values of the correlation coefficients was observed for SBP.

In the following analysis the variance of the correlation coefficients for each patient is considered. It shows how the correlation was changing with time of data collecting from patients. The variance of the correlation coefficient for SBP and DBP was similar (a little bit lower for DBP). The lowest variance was obtained for trend analysis, while the largest one for MA filtering. The variance was increasing with the length of the filter (for MA) and number of samples taken into account for the calculation of the  $a$  coefficient.

The ratio of significant ( $p < 0.05$ ) to not significant correlation coefficients depended on the type of analysis (raw data, MA filtered, and trend analysis). The values for DBP were about 10% higher than for SBP. The larger number of significant values was obtained for MA analysis and the ratio grew along with the filter length (from 0.66 to 0.71 for SBP and from 0.73 to 0.77 for DBP). A smaller number of significant values was obtained for trend analysis. However, an observed dependence was not monotonic (values from 0.43 to 0.56 for SBP and 0.48 to 0.60 for DBP). Moreover, there was large interpersonal dependence – the ratio changes even from 0.2 to 0.7 for the same method but different persons for SBP. For DBP the variances were smaller.

Influence of the number of samples utilized in the correlation analysis on results obtained is discussed in the following paragraph. Two types of data were compared – containing 50 and more samples to these ones having at least 100 samples. There was no dependence of average value of the correlation coefficient on the length of the data sequence taken into account. In some patients the correlation coefficient was decreasing and in some cases increasing. It was more stable for DBP, while for SBP very large changes for 2 patients were observed. It can be explained if we look at the Fig. 4. It follows from this figure, that one can expect that average value of the correlation coefficient is a random number. However, comparison of the average of the absolute values of the correlation coefficients showed that in most cases this value was decreasing for longer sequences (similarly for SBP and DBP, 10 % on average and it is not dependent on the method of analysis – raw data, MA filtering, trend analysis). Similar results were obtained for variance analysis, but the decrease was even larger – about 20 %. As it can be expected the number of significant correlation coefficients ( $p < 0.05$ ) increases in general with increasing sequence length. Similar results for SBP and DBP, like in the previous

cases, not dependent on the method of analysis – raw data, MA filtering, trend analysis, were observed. The average increase was 10 %.

Finally, the correlation coefficients between HR and SBP or DBP measured invasively for all data and for each patient were calculated. All the correlation coefficients were significant ( $p < 0.05$ ). The correlation coefficients for HR and SBP were equal to 0.30, 0.31, -0.10, and -0.24 for subsequent patients, while for HR and DBP they were respectively -0.02, 0.14, -0.29, and -0.10. The number of measurements for each patient varied from 8451 to 21248 for each parameter. It suggests that a long-term correlation is rather a random number, as can be expected from Fig. 4.

A medicine taken by the patients and interventions of medical doctors could change cardiovascular relationships, thus also our results. Unfortunately, there was no included any information on any medical treatment or interventions in the data. Maybe the low correlation was observed after drug administration, so taking into account such data may allow predicting BP values for HR. However, it should be remembered that elderly people are taking medicines regularly even non-prescribed by physician.

## V. CONCLUSION

The observed correlation coefficient between HR and BP (systolic and diastolic) for whole available data seems a random number. However, the short-term correlation is relatively large (about 0.5), but rather unpredictable, since even sign of the correlation coefficient is changing.

## REFERENCES

- [1] R. W. de Boer, J. M. Karemaker, J. Strackee, "Relationships between short-term blood-pressure fluctuations and heart-rate variability in resting subjects. I: A spectral analysis approach," *Med. Biol. Eng. Comput.*, vol. 23, pp. 352-358, 1985.
- [2] R. W. de Boer, J. M. Karemaker, J. Strackee, "Relationships between short-term blood-pressure fluctuations and heart-rate variability in resting subjects. II: A simple model," *Med. Biol. Eng. Comput.*, vol. 23, pp.359-364, 1985.
- [3] M. Valentini, G. Parati, "Variables influencing heart rate," *Progress in Cardiovascular Diseases* vol. 52, pp. 11–19, 2009.
- [4] J. Zhang and H. Kesteloot, "Anthropometric, lifestyle and metabolic determinants of resting heart rate. A population study," *European Heart Journal*, vol. 20, pp. 103–110, 1999.
- [5] J Filipovsky, P Ducimetiere and ME Safar, "Prognostic significance of exercise blood pressure and heart rate in middle-aged men," *Hypertension* vol. 20; pp. 333-339, 1992.
- [6] P. Palatini; E. Casiglia; P. Pauletto; J. Staessen; N. Kaciroti; S. Julius, "Relationship of tachycardia with high blood pressure and metabolic abnormalities. A study with mixture analysis in three populations," *Hypertension*, vol. 30, pp. 1267-1273, 1997

# Computer Aspects of Numerical Algorithms

**N**UMERICAL algorithms are widely used by scientists engaged in various areas. There is a special need of highly efficient and easy-to-use scalable tools for solving large scale problems. The workshop is devoted to numerical algorithms with the particular attention to the latest scientific trends in this area and to problems related to implementation of libraries of efficient numerical algorithms. The goal of the workshop is meeting of researchers from various institutes and exchanging of their experience, and integrations of scientific centers

## TOPICS

- Parallel numerical algorithms
- Novel data formats for dense and sparse matrices
- Libraries for numerical computations
- Numerical algorithms testing and benchmarking
- Analysis of rounding errors of numerical algorithms
- Languages, tools and environments for programming numerical algorithms
- Numerical algorithms on GPUs
- Paradigms of programming numerical algorithms
- Contemporary computer architectures
- Heterogeneous numerical algorithms
- Applications of numerical algorithms in science and technology

## PROGRAM COMMITTEE

**Pierluigi Amodio**, Universita' di Bari, Italy  
**Zacharias Anastassi**, Technological Educational Institute of Kalamata, Greece  
**Krzysztof Banaś**, AGH, Poland  
**Luigi Brugnano**, Universita degli Studi, Firenze, Italy  
**Tadeusz Czachórski**, IITiS PAN, Poland  
**Tim Davis**, University of Florida, USA  
**Tugrul Dayar**, Bilkent University, Turkey  
**Stefka Dimova**, FMI, Sofia University "St. Kliment Ohridski", Bulgaria  
**Salvatore Filippone**, Universita di Roma 'Tor Vergata', Italy  
**Mauro Francaviglia**, Università di Torino, Italy  
**Wilfried Gansterer**, University of Vienna, Austria  
**Krassimir Georgiev**, Bulgarian Academy of Sciences, Bulgaria  
**Pawel Gepner**, Intel Corporation, USA  
**Domingo Gimenez**, University of Murcia, Spain  
**George Gravvanis**, Democritus University of Thrace, Greece  
**Andreas Karageorghis**, University of Cyprus, Cyprus  
**Jacek Kierzenka**, The MathWorks, Inc., USA  
**Steve Kirkland**, National University of Ireland Maynooth, Ireland

**Jerzy Klamka**, Silesian University of Technology, Poland  
**William Knottenbelt**, Imperial College London, United Kingdom

**Udo Krieger**, Otto-Friedrich University, Germany

**Anna Kucaba-Pietal**, Rzeszow University of Technology, Poland

**Ivan Lirkov**, IPP BAS, Bulgaria

**Vyacheslav Maksimov**, Institute of Mathematics and Mechanics, UB RAS; Ural Federal University, Russian Federation

**Ami Marowka**, Bar-Ilan University Ramat-Gan, Israel

**Francaviglia Mauro**, Università di Torino, Italy

**Beatrice Meini**, University of Pisa, Italy

**Peter Minev**, University of Alberta, Canada

**Dana Petcu**, West University of Timisoara, Romania

**Ivana Pultarová**, Czech Technical University in Prague, Czech Republic

**Bianca-Renata Satco**, "Stefan cel Mare" University of Suceava, Romania

**Stanislav Sedukhin**, University of Aizu, Japan

**Vladimir V. Sergeichuk**, Institute of Mathematics, National Academy of Sciences, Ukraine

**Natesan Srinivasan**, Indian Institute of Technology, India

**Daniel B. Szyld**, Temple University, USA

**Miklós Telek**, Technical University of Budapest, Hungary

**Miroslav Tůma**, Institute of Computer Science, Academy of Sciences of the Czech Republic, Czech Republic

**Kishor S. Trivedi**, Electrical and Computer Engineering Duke University, USA

**Marek Tudruj**, Institute of Computer Science Polish Academy of Sciences, Poland

**Vasyl Ustimenko**, Maria Curie-Skłodowska University, Poland

**Alexander Vazhenin**, University of Aizu, Japan

**Verena Wolf**, Saarland University, Germany

**Zlatev Zahari**, National Environmental Research Institute, Aarhus University, Denmark

## SIAM REPRESENTATIVE

**Marcin Paprzycki**, IBS PAN and WSM, Poland

## ORGANIZING COMMITTEE

**Beata Bylina**, Maria Curie-Skłodowska University, Poland

**Jarosław Bylina**, Maria Curie-Skłodowska University, Poland

**Przemysław Stpiczynski** (Chairman), Maria Curie-Skłodowska University, Poland



# The incomplete factorization preconditioners applied to the GMRES( $m$ ) method for solving Markov chains

Beata Bylina    Jarosław Bylina

Institute of Mathematics

Marie Curie-Skłodowska University

Pl. M. Curie-Skłodowskiej 5, 20-031 Lublin, Poland

Email: beata.bylina@umcs.pl, jaroslaw.bylina@umcs.pl

**Abstract**—This paper is a review and a comparison of some preconditioners based on incomplete factorizations of matrices — for matrices describing Markov chains. Three preconditioners are considered: ILU(0), ILU3, IWZ(0). Two of them (ILU(0), ILU3) are based on the LU factorization, the latter (IWZ(0))— on the WZ factorization. The preconditioners are investigated in respect of their usability for decreasing number of iterations in a projection method, namely GMRES( $m$ ). To choose the best preconditioner for such methods, authors introduce a measure called *iteration speed-up* ( $p$ ) and some of its relatives, as well as they define a function giving an average number of restarts needed to achieve a given accuracy for matrices from a some set ( $I_s$ ). These measures are studied for two different cases of matrices describing Markov chains to compare influence of the examined incomplete preconditioners for GMRES( $m$ ).

$O(n^2)$  for  $n$  iterations), which is a big disadvantage for huge systems coming from Markov chains. On the other hand, GMRES( $m$ ) (that is, the GMRES method restarted after  $m$  iterations) requires at most  $O(m \cdot n)$  additional space for computations, regardless of the number  $k$  of restarts (because after every restart the working space can be reused), but the achieved accuracy can be comparable (for an appropriate  $m$ , of course, which is often not easy to choose) to the accuracy achieved with GMRES after the same total number of iterations — that is  $k \cdot m$  — but in the latter case the space needed is  $O(k \cdot m \cdot n)$ . So, we investigate a restarted version, GMRES( $m$ ), as a potentially more economical method for huge systems.

## I. INTRODUCTION AND MOTIVATION

**W**HILE modelling probabilities stationary distributions (independent of time) with Markov chains, we obtain a following linear equation system:

$$\mathbf{Q}^T \mathbf{x} = \mathbf{0}, \quad \mathbf{x} \geq \mathbf{0}, \quad \mathbf{x}^T \mathbf{e} = 1, \quad (1)$$

where  $\mathbf{Q}$  is a transition rate matrix,  $\mathbf{x}$  is an unknown vector of states' probabilities and  $\mathbf{e} = (1, 1, \dots, 1)^T$ . The matrix  $\mathbf{Q}$  is a singular square one of size  $n \times n$ , of rank  $n - 1$ , with a weakly dominant diagonal, usually a sparse, large and ill-conditioned one. These traits of  $\mathbf{Q}$  cause the need to treat the system (1) specially.

One of the most popular methods to solve the system 1 is the GMRES method [13]. The full GMRES algorithm (that is: GMRES( $n$ ),  $n$  being the size of the system) is guaranteed to converge in at most  $n$  steps when we used full precision arithmetic (that means: no restarts are needed), but it is not very useful for large systems of equations, because a good approximate solution is often computed quite early, after very few iterations.

Moreover, the traditional GMRES (without restarts) requires quite a lot of space (additional  $O(n)$  for every iteration, so

The very concept of the preconditioning is almost as old as iterative methods [8]. One of the most famous preconditioning techniques is the incomplete factorization of the original matrix  $\mathbf{Q}$ . The idea of the incomplete factorization was presented by Buleev [3], [4] and Varga [15]. The papers that popularized the incomplete factorizations were [9], [10].

There is a need for preconditioners that are fast, stable, scalable, easy to parallelize and that generate a small fill-in. In [1], [2] preconditioners for Krylov subspace methods for solving large singular linear systems arising from Markov modeling are considered.

The incomplete LU (ILU) factorization process computes a sparse lower triangular matrix  $\mathbf{L}$  and a sparse upper triangular matrix  $\mathbf{U}$ . Here we discuss the ILU(0) factorization, the simplest form of the ILU preconditioners. ILU(0) consists in taking the zero pattern as the original matrix  $\mathbf{Q}$ . Using ILU(0) for solving Markov chains was shown in [14].

ILU3 is another kind of incomplete factorization based on the LU factorization. Here, the factors are nonzero only on their three central diagonals — for the matrix  $\mathbf{L}$  it is the main diagonal and the one directly below, and for the matrix  $\mathbf{U}$  it is the main diagonal and the one directly above. This factorization was not used to Markov chains so we wanted to test it.

The incomplete WZ factorization is originally described in some previous works [5]. In [6] we discussed its performance for GMRES. This work is a step forward in the investigation

This work was partially supported within the project N N516 479640 of the Ministry of Science and Higher Education of the Polish Republic (MNiSW) *Modele dynamiki transmisji, sterowania zatkanieniem i jakością usług w Internecie*.



of such incomplete preconditioners and compares some kinds of IWZ and ILU.

We are concerned in the influence of the preconditioner's structure on the GMRES( $m$ ) method, so we are considering three incomplete factorization methods, namely ILU(0), ILU3 i IWZ(0). These methods differ with the structure of the factors and the number of nonzeros. In factorizations IWZ(0) and ILU(0) the number of nonzeros is exactly the same as in the original matrix  $\mathbf{Q}$  but factors ( $\mathbf{L}$  and  $\mathbf{U}$  in ILU(0),  $\mathbf{W}$  and  $\mathbf{Z}$  in IWZ(0)) have different structures. The ILU3 factorization has usually less nonzero elements than the original matrix  $\mathbf{Q}$  and the structure of the factors is similar to ILU(0). ILU3 is less accurate than ILU(0) and IWZ(0), so we can expect worse results.

We consider an impact of the incomplete factorization preconditioners on the GMRES( $m$ ) method for the numerical solution of Markov chains. We study the relationship between the number of iterations, the convergence rate of the GMRES( $m$ ) method and properties of the matrix  $\mathbf{Q}$  and the structure of preconditioners. For better understanding of the behavior of the GMRES( $m$ ) convergence and its preconditioners we introduce some measures — the iteration speed-up among others (see Section IV-A).

The research was carried out for two cases. The first case are matrices that have not got any particular structure and we assume that the matrix row and column ordering is given and cannot be changed. The second case are matrices of a Markov chain known from the literature as the epidemic model and these matrices have got a structure. For all those (sparse) matrices we introduced another measure — matrix density.

The rest of the paper is organized as follows. Section II recalls briefly the incomplete preconditioning. Section III presents two test cases. Section IV describes conducted numerical experiments. Section V contains some conclusions.

## II. INCOMPLETE PRECONDITIONERS

The convergence rate of iterative methods depends on properties of the coefficient matrix of the linear system. If the matrix  $\mathbf{Q}$  is ill-conditioned, this can make the convergence of iterative methods slow. One way to prevent such problems is to transform the system (1) into an equivalent system (that is, having the same solution), but with better numerical properties. Such a transformation can be done by preconditioning, that is by converting the system (1) into:

$$\mathbf{M}^{-1}\mathbf{Q}^T\mathbf{x} = \mathbf{0}, \quad \sum_{i=1}^n x_i = 1, \quad \mathbf{x} \geq \mathbf{0}, \quad (2)$$

where the nonsingular matrix  $\mathbf{M}$  (known as a preconditioner) approximates the matrix  $\mathbf{Q}^T$  in a manner. The system (2) has the same solutions as (1) but it is (hopefully) better conditioned.

The matrix  $\mathbf{M}$  should have the following properties:

- its use should entail low memory requirements;
- its inverse should be cheaply applicable;
- the transformed problem (2) should converge faster (in shorter computational time) than the original problem.

Of course, there is a clear conflict among these three requirements, especially for the construction of general purpose preconditioners.

Generally, computing and using a good preconditioner is an expensive task consisting of finding the matrix  $\mathbf{M}$  and its inverse. If the preconditioning is to be used, that cost should be refunded by a reduced number of iterations needed to acquire a required accuracy — or by using the same preconditioner for various linear systems.

The preconditioner matrix is usually built on the base of the original coefficients of the matrix  $\mathbf{Q}$ .

### A. ILU(0) preconditioner

The incomplete LU factorization (denoted ILU) is based on the well known LU factorization, where a lower triangular matrix (with ones on the diagonal)  $\tilde{\mathbf{L}}$  and an upper triangular matrix  $\tilde{\mathbf{U}}$  are found and where the preconditioner matrix  $\mathbf{M} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}$  is a kind of approximation for the matrix  $\mathbf{Q}^T$ .

There are many variants of ILU, the most straightforward being ILU(0) [14]. In ILU(0) the computations are conducted as in the traditional (complete) LU factorization (that is, the Gaussian elimination), but any new nonzero element ( $l_{ij}$  and  $u_{ij}$ ) arising in the process is dropped if it appears in the place of a zero element in the original matrix  $\mathbf{Q}^T$ . Hence, the factors together have the same number of nonzeros as the original matrix  $\mathbf{Q}^T$ . Thereby, the most important problem of the factorization of sparse matrices — the fill-in (which consists in appearing nonzero elements in new matrices on the places of zero elements in the original matrix, what makes dense the output factors and renders impossible their packed storage) — is eliminated. At the expense of accuracy, of course.

After ILU(0) we have:

$$\mathbf{Q}^T = \tilde{\mathbf{L}}\tilde{\mathbf{U}} + \mathbf{R}_{LU}, \quad (3)$$

where  $\tilde{\mathbf{L}}$  and  $\tilde{\mathbf{U}}$  are (respectively) the lower triangular matrix and the upper triangular matrix and the remainder matrix  $\mathbf{R}_{LU}$  is hoped to be small in a sense.

Let  $\mathbf{M} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}$ , then  $\mathbf{M}^{-1} = \tilde{\mathbf{U}}^{-1}\tilde{\mathbf{L}}^{-1}$  and the equation (2) takes the shape:

$$\tilde{\mathbf{U}}^{-1}\tilde{\mathbf{L}}^{-1}\mathbf{Q}^T\mathbf{x} = \mathbf{0}, \quad \sum_{i=1}^n x_i = 1, \quad \mathbf{x} \geq \mathbf{0}. \quad (4)$$

Let  $\mathbf{S}_{LU} = \tilde{\mathbf{U}}^{-1}\tilde{\mathbf{L}}^{-1}\mathbf{Q}^T$ . Now, the equation (4) takes the shape:

$$\mathbf{S}_{LU}\mathbf{x} = \mathbf{0}, \quad \sum_{i=1}^n x_i = 1, \quad \mathbf{x} \geq \mathbf{0}. \quad (5)$$

The following Octave code was used to generate the ILU(0) factors used in this article. For a given matrix  $\mathbf{Q}$  two triangular matrices are constructed. The matrix ( $\mathbf{L}$ ) is lower triangular (the main diagonal of  $\mathbf{L}$  is filled with ones only) and the matrix ( $\mathbf{U}$ ) — upper triangular. The time complexity is  $O(n^3)$ .

```

function [L,U]=ilu0(Q)
[n,n]=size(Q); U=Q; L=zeros(n);
for i=1:n, L(i,i)=1.0; end;
for k=2:n,
    for i=1:k-1,
        if Q(k,i) != 0,
            L(k,i)=U(k,i)/U(i,i);
            for j=i+1:n,
                U(k,j)=U(k,j)-L(k,i)*U(i,j);
            end;
        end;
    end;
for i=1:n,
    if Q(k,i)==0, U(k,i)=0; end;
end;
for i=2:n,
    for j=1:n,
        if i>j, U(i,j)=0; end;
    end;
end;
end;

```

### B. ILU3 preconditioner

ILU3 is an incomplete LU factorization conducted quite similarly to ILU(0), but the structure of output matrices  $L$  and  $U$  have nothing to do with the structure of the input matrix  $Q^T$ . The output matrices simply consist of diagonals: the main one and its lower neighbor diagonal ( $L$ ) or the main one and its upper neighbor diagonal ( $U$ ).

As for now, this factorization was not used in solving Markov chains. However, it is quite fast ( $O(n)$ ) and easy to use in parallel. The code is shown below.

```

function [L,U]=ilu3(Q)
[n,n]=size(Q); U=Q; L=zeros(n);
for i=1:n, L(i,i)=1.0; end;
for k=2:n,
    i=k-1;
    L(k,i)=U(k,i)/U(i,i);
    U(k,k)=U(k,k)-L(k,i)*U(i,k);
    if (k<n),
        U(k,k+1)=U(k,k+1)-L(k,i)*U(i,k+1);
    end;
end;
for j=1:n,
    for i=1:n,
        if (i>j), U(i,j)=0; end;
        if (i+2<=j), U(i,j)=0; end;
    end;
end;
end;

```

### C. IWZ(0) preconditioner

The incomplete WZ (denoted IWZ) factorization is originally described in a previous works [5]; here we only recall it. The WZ factorization (on which IWZ is based) consists in decomposition of the given matrix ( $Q^T$  in the paper) into a product of two matrices:  $W$  and  $Z$  (Fig. 1).

The incomplete WZ factorization (IWZ) is based on the WZ factorization described above, where we find matrices  $\tilde{W}$  and

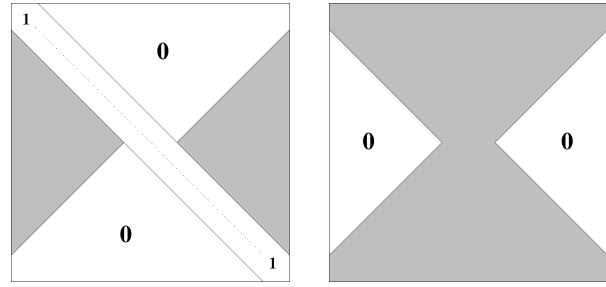


Fig. 1. The form of the output matrices in the WZ factorization (left:  $W$ ; right:  $Z$ )

$\tilde{Z}$  (of the form of the matrices  $W$  and  $Z$  shown in Fig. 1) and the product  $\tilde{W}\tilde{Z}$  is a kind of approximation for the matrix  $Q^T$ .

In IWZ(0) the computations are conducted as in the complete WZ factorization, but any new nonzero elements ( $w_{ij}$  and  $z_{ij}$ ) arising in the process are dropped if they appear in the place of a zero element in the original matrix  $Q^T$ . Hence, the factors together have the same number of nonzeros as the original matrix  $Q^T$ . It is worth noting that we got the inverse of  $\tilde{W}$  very easily, because [16]:

$$\tilde{W}^{-1} = (-1) \cdot (\tilde{W} - I) + I \quad (6)$$

$$\text{(just like } W^{-1} = (-1) \cdot (W - I) + I). \quad (7)$$

After IWZ(0) we have:

$$Q^T = \tilde{W}\tilde{Z} + R_{WZ}, \quad (8)$$

where  $\tilde{W}$  and  $\tilde{Z}$  are (respectively) matrices of the form of  $W$  and  $Z$  from Fig. 1 and the remainder matrix  $R_{WZ}$  is supposed to be small in a sense.

The time complexity is  $O(n^3)$  — just like for ILU(0).

Here is an Octave code that shows the sequence of operations that must be followed and that provides some feel for the applicability of the algorithm. This (and previous, that is for ILU(0) and ILU3) code is not meant to represent production versions of the algorithms.

```

function [W,Z]=iwz0(Q)
[n,n]=size(Q); Z=Q; W=zeros(n);
for i=1:n, W(i,i)=1.0; end;
for k=1:n/2-1,
    k2=n-k+1;
    det=Z(k,k)*Z(k2,k2)-Z(k2,k)*Z(k,k2);
    for i=k+1:k2-1,
        if Q(i,k)!=0,
            W(i,k)=(Z(k2,k)*Z(i,k2)-Z(k2,k2)*Z(i,k))/det;
        end;
        if Q(i,k2)!=0,
            W(i,k2)=(Z(k,k2)*Z(i,k)-Z(k,k)*Z(i,k2))/det;
        end;
    end;
    for j=k+1:k2-1,
        if Q(i,j)!=0,
            Z(i,j)=Z(i,j)+W(i,k)*Z(k,j)+W(i,k2)*Z(k2,j);
        end;
    end;

```

TABLE I  
THE ESSENTIAL CHARACTERISTICS OF THE MATRICES USED IN THE TESTS

Group	matrix ID	$n$	$nz$	$d$
A	1	100	1190	11.9
B	2	100	388	3.9
A	3	1500	37955	25.3
B	4	1500	5873	3.9
A	5	3000	120590	40.2
B	6	3000	11636	3.9

```

end;
end;
end;
for i=1:n,
    for j=1:n,
        if ((i>j)&(i<n-j+1))
            | ((i<j)&(i>n-j+1)),
                Z(i,j)=0; end;
        end;
    end;
end;
end;

```

### III. TEST CASES

Here we shortly present two cases which were used to investigate the influence of our preconditioners (ILU(0), ILU3 and IWZ(0)) on the convergence of GMRES( $m$ ). We assume that the investigated matrix cannot be a subject for any reordering.

#### A. Case I

The matrices of the Case I were created randomly, with given some parameters as  $n$  (the number of rows) and  $nz$  (the number of nonzeros) as well as the range of the elements outside the diagonal. For every nonzero element, its indices (that is the number of its row and the number of its column) were randomly (uniformly) chosen as well as its value. Then, the diagonal elements were computed (from:  $q_{ii} = -\sum_{j \neq i} q_{ij}$ ) to get a correct transition rate matrix.

In Table I the essential characteristics of the matrices are presented ( $n$  is the number of rows and columns of the matrix,  $nz$  is the number of nonzeros in the matrix,  $d = nz/n$ ).

For these matrices we can observe, that the matrices might have the same size and a different value of  $d$ . The matrices were divided into two groups, the first group include matrices with  $d > 8$  (Group A), the second group include matrices with  $d \leq 8$  (Group B).

The structure of the matrix with ID=3 is shown in Fig. 2. We can see that the matrices from Case I have no particular structure. It is worth noting that IWZ(0) and ILU(0) for them have not such a structure either, because their structures are based on structures of  $\mathbf{Q}$ . And ILU3 preconditioner is just a tri-diagonal — as usual.

#### B. Case II

The matrix of Case II was generated from a standard two-dimensional Markovian model [7], [11]. This particular example has been taken from [11], [12]. The states of the chain are described with two numbers  $(u, v)$ ,  $u = 0, \dots, N_x$ ,

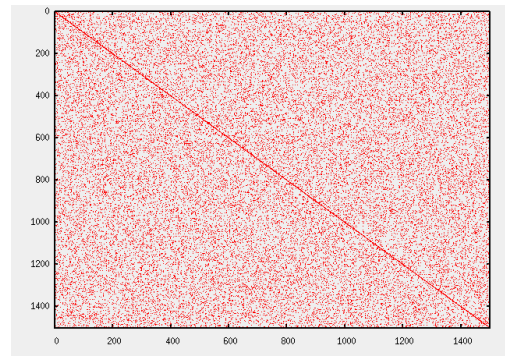


Fig. 2. The structure of the matrix with ID=3 of Case I

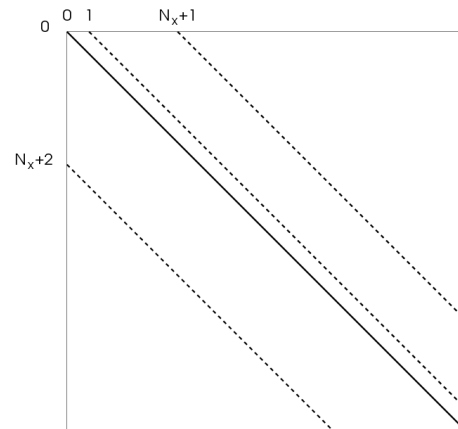


Fig. 3. The structure of the matrix of Case II

$v = 0, \dots, N_y$  (here  $N_x = 64$ ,  $N_y = 16$ ). The matrix describing the two-dimensional Markov chain has a structure shown in Fig. 3.

It is worth noting that for the matrix of Case II, the investigated preconditioners are also structured. It is so because ILU(0) and IWZ(0) inherit their structures after the original matrix (Fig. 3). But ILU3 is here somewhat a special case, because here it reduces to the Jacobi preconditioner (that is  $\mathbf{L} = \mathbf{I}$  and  $\mathbf{U} = \text{diag}(q_{ii})$ ).

### IV. EXPERIMENTAL RESULTS

The main goal of the numerical experiments was to test the incomplete factorization preconditioners in respect of their usability for the GMRES( $m$ ) method for matrices arising from Markov chains. The experiment was performed on a Pentium IV 2.8GHz computer with 1GB RAM under the Debian GNU/Linux operating system. We used a high-level programming language, namely Octave (a GNU equivalent of MATLAB).

A vector  $\mathbf{x}^{(0)} = (x_i^{(0)})$  with  $x_i^{(0)} = \frac{1}{i}$  was chosen as an initial vector. As a measure of accuracy of the solution we chose the 2-norm of the residual:

$$\varepsilon^{(k)}(\mathbf{Q}) = \|\mathbf{0} - \mathbf{Q}^T \mathbf{x}^{(k)}\|_2 = \|\mathbf{Q}^T \mathbf{x}^{(k)}\|_2. \quad (9)$$

TABLE II

NUMBER OF ITERATIONS NEED TO ACHIEVE A GIVEN ACCURACY  $\varepsilon^{(k)} < 10^{-16}$  FOR THE SELECTED VALUES OF THE PARAMETER  $m$  FOR CASE I (MATRICES FROM GROUP A ARE PRINTED NORMALLY, THOSE FROM GROUP B ARE DISTINGUISHED WITH A BOLD FONT)

$m$	method	matrix ID					
		1	<b>2</b>	3	<b>4</b>	5	<b>6</b>
$m = 1$	GMRES( $m$ )	45	<b>87</b>	44	<b>185</b>	42	<b>271</b>
	IWZ(0)G-( $m$ )	14	<b>29</b>	11	<b>33</b>	10	<b>34</b>
	ILU(0)G-( $m$ )	14	<b>28</b>	12	<b>33</b>	11	<b>33</b>
	ILU3G-( $m$ )	31	<b>66</b>	24	<b>80</b>	21	<b>84</b>
$m = 5$	GMRES( $m$ )	8	<b>17</b>	7	<b>30</b>	6	<b>32</b>
	IWZ(0)G-( $m$ )	3	<b>6</b>	3	<b>7</b>	2	<b>7</b>
	ILU(0)G-( $m$ )	4	<b>6</b>	3	<b>7</b>	3	<b>7</b>
	ILU3G-( $m$ )	6	<b>14</b>	5	<b>16</b>	5	<b>17</b>
$m = 10$	GMRES( $m$ )	4	<b>8</b>	4	<b>14</b>	3	<b>15</b>
	IWZ(0)G-( $m$ )	2	<b>3</b>	2	<b>4</b>	1	<b>4</b>
	ILU(0)G-( $m$ )	3	<b>4</b>	3	<b>5</b>	3	<b>4</b>
	ILU3G-( $m$ )	3	<b>7</b>	3	<b>8</b>	3	<b>9</b>

Above (and throughout the whole paper), as  $k$ , we consider the number of external iterations of GMRES( $m$ ), that is the number of restarts.

The accuracy has been studied experimentally for the matrices of Case I and Case II. We studied both the number of iterations needed to achieve a given accuracy, and the convergence rate. The stop condition used here is that the 2-norm of the residual (that is  $\varepsilon^{(k)}(\mathbf{Q}) = \|\mathbf{Q}^T \mathbf{x}^{(k)}\|_2$ ) is less than  $10^{-16}$  (such a chosen value is quite real if we are to find probabilities of unlikely but important events — as a packet loss or a channel jamming — precisely). To improve the readability we did not use  $\varepsilon^{(k)}(\mathbf{Q})$  in the results presentation, but rather:

$$\text{acc}(\mathbf{Q}, i) = -\log_{10} \varepsilon^{(i)}(\mathbf{Q}). \quad (10)$$

#### A. Number of iterations

Table II shows numbers of iterations (external iterations, that is restarts) used to achieve a given accuracy for selected parameters  $m$  for four methods: GMRES( $m$ ) alone (denoted GMRES( $m$ )) and GMRES( $m$ ) preconditioned with IWZ(0) (denoted IWZ(0)GMRES and ( $m$ )IWZ(0)G-( $m$ )), ILU(0) (denoted ILU(0)GMRES( $m$ ) and ILU(0)G-( $m$ )) and ILU3 (denoted ILU3GMRES( $m$ ) and ILU3G-( $m$ )).

For a deeper analysis of the influence of the preconditioners on the method, we consider some more measures.

Let  $I(M, \mathbf{A}, \varepsilon)$  denote a number of restarts (that is external iterations) needed to achieve the 2-norm of the residual less than  $\varepsilon$  for a given matrix  $\mathbf{A}$  with a given method  $M$ . In other words,  $I(M, \mathbf{A}, \varepsilon)$  is a minimal  $k$  for which defined in (9)  $\varepsilon^{(k)}(\mathbf{A}) < \varepsilon$ .

Let us define  $p(PM, \mathbf{A}, \varepsilon)$ , which shows the relationship between the number of iterations needed to achieve a given accuracy  $\varepsilon$  with a method  $M$  with no preconditioner and the same method  $M$  with a preconditioner  $P$  — both for the same matrix  $\mathbf{A}$ . We call it *iteration speed-up* and define as follows:

$$p(PM, \mathbf{A}, \varepsilon) = \frac{I(M, \mathbf{A}, \varepsilon)}{I(PM, \mathbf{A}, \varepsilon)}. \quad (11)$$

Now, let  $Is(M, Z, \varepsilon)$  be an average number of restarts needed to achieve a given accuracy for matrices from a set  $Z$ :

$$Is(M, Z, \varepsilon) = \text{avg}_{\mathbf{A} \in Z} I(M, \mathbf{A}, \varepsilon). \quad (12)$$

Next, we define  $ps(PM, Z, \varepsilon)$  which shows the relationship between the average number of iterations needed to achieve a given accuracy  $\varepsilon$  with a method  $M$  with no preconditioner and the same method  $M$  with a preconditioner  $P$  — for a set  $Z$  of matrices.

$$ps(PM, Z, \varepsilon) = \frac{Is(M, Z, \varepsilon)}{Is(PM, Z, \varepsilon)}. \quad (13)$$

At last, we are going to define some more characteristics with  $p$  defined above (11). They will be the maximal and minimal  $p$  for a given matrix  $\mathbf{A}$  solved with a preconditioned GMRES( $m$ ) (for  $m \in \{1, \dots, m_0\}$ ; with a preconditioner  $P$ ) as well as  $m$  giving that  $p$ :

$$p_{\max}(PGMRES(m), m_0, \mathbf{A}, \varepsilon) = \max_{1 \leq i \leq m_0} p(PGMRES(i), \mathbf{A}, \varepsilon), \quad (14)$$

$$p_{\min}(PGMRES(m), m_0, \mathbf{A}, \varepsilon) = \min_{1 \leq i \leq m_0} p(PGMRES(i), \mathbf{A}, \varepsilon), \quad (15)$$

$$m_{\max}(PGMRES(m), m_0, \mathbf{A}, \varepsilon) = \arg \max_{1 \leq i \leq m_0} p(PGMRES(i), \mathbf{A}, \varepsilon), \quad (16)$$

$$m_{\min}(PGMRES(m), m_0, \mathbf{A}, \varepsilon) = \arg \min_{1 \leq i \leq m_0} p(PGMRES(i), \mathbf{A}, \varepsilon). \quad (17)$$

Whenever we omit the parameter  $\varepsilon$  (as in  $p(\text{ILU}(0)\text{GMRES}(m), \mathbf{Q})$  where we mean iteration speed-up of the GMRES method restarted after  $m$  inner iterations preconditioned with the ILU(0) factorization — and similar), we assume  $\varepsilon = 10^{-16}$ .

Fig. 4 shows  $ps(PGMRES(m), \mathbf{A})$  and  $ps(PGMRES(m), \mathbf{B})$  — that is the relationship between the parameter  $m$  and the average number of restarts (external iterations) needed to achieve the given accuracy for  $P$  being preconditioners: ILU(0), ILU3, IWZ(0). The average number of iterations is counted for two groups of matrices: Group A with  $d > 8$  (the matrices 1, 3, 5) and Group B with  $d \leq 8$  (the matrices 2, 4, 6).

Some conclusions that can be drawn from Table II and Fig. 4 are following.

- With the increase of the parameter  $m$  the average number of iterations needed to achieve the assumed accuracy of the method GMRES( $m$ ), IWZ(0)GMRES( $m$ ), ILU3GMRES( $m$ ) and ILU(0)GMRES( $m$ ) decreases.
- Regardless of the size of the matrix, the number of iterations needed to achieve a given accuracy is practically the same and depends on the value of  $d = nz/n$  (see Group A versus Group B).
- In the methods IWZ(0)GMRES( $m$ ), ILU(0)GMRES( $m$ ) and ILU3GMRES( $m$ ) the number of outer iterations (that is, restarts) needed to achieve the given convergence is

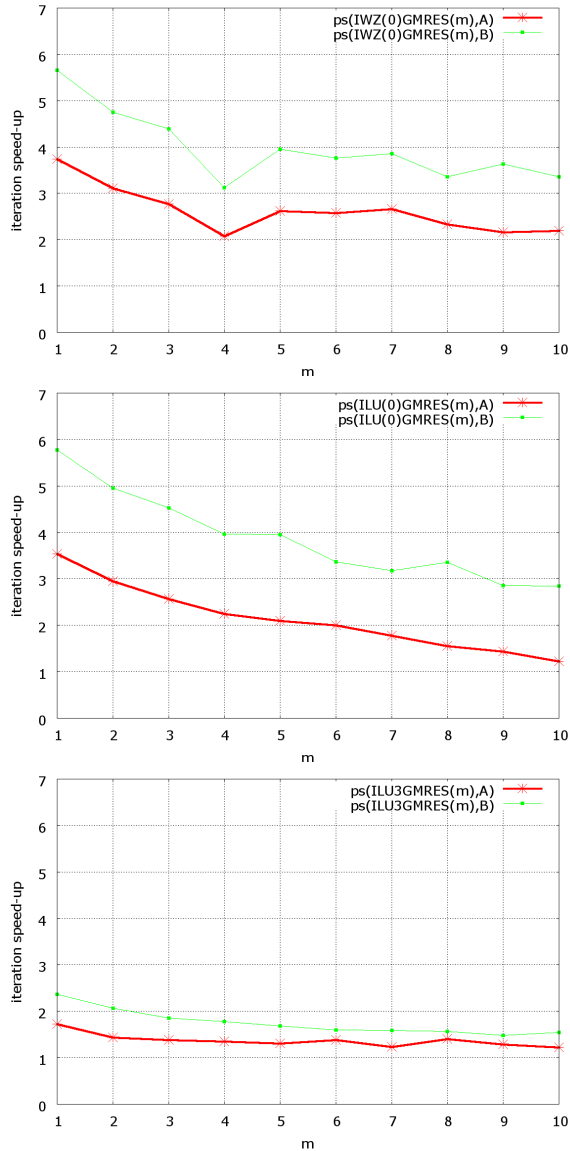


Fig. 4. The relationship between the parameter  $m$  and the average number of restarts (external iterations) needed to achieve the accuracy  $10^{-16}$  for the matrices of Group A ( $d > 8$ ) and Group B ( $d \leq 8$ ) for Case I with preconditioners IWZ(0) (top); ILU(0) (middle); ILU3 (bottom)

less than in the GMRES( $m$ ) method, regardless of the parameter  $m$  and the parameter  $d$ .

- The value of the iteration speed-up ( $p$ ) for the matrices of Group A is not less than for those of Group B.
- For the matrices of Group B, the growth of  $p(\text{ILU}(0)\text{GMRES}(m), \mathbf{Q})$  and  $p(\text{IWZ}(0)\text{GMRES}(m), \mathbf{Q})$  is more uniform than for the matrix of Group A.

Table III shows the minimal and maximal values of  $p$  (that is  $p_{\min}$  and  $p_{\max}$  from (14) and (15)) and the values of the parameter  $m$  for which those values of  $p$  are reached ( $m_{\min}$  and  $m_{\max}$  from (16) and (17)) — in Case I.

Table IV provides values  $p(\text{ILU}(0)\text{GMRES}(m), \mathbf{Q})$ ,

TABLE III  
MAXIMAL AND MINIMAL VALUES OF THE CHARACTERISTICS  $p$  AND THE RESPECTIVE VALUES OF  $m$  FOR MATRICES OF CASE I

precond.	value	matrix ID					
		1	2	3	4	5	6
IWZ(0)	$p_{\max}$	3.21	3.00	4.00	5.61	4.20	<b>7.97</b>
	$m_{\max}$	9	8	1	2	1	1
	$p_{\min}$	2.00	2.50	2.00	3.50	2.00	3.75
	$m_{\min}$	8	2	7	10	6	9
ILU(0)	$p_{\max}$	3.21	3.11	3.67	5.69	3.82	<b>8.21</b>
	$m_{\max}$	1	1	1	1	1	1
	$p_{\min}$	1.33	2.00	1.33	2.80	1.00	3.20
	$m_{\min}$	10	10	10	10	10	9
ILU3	$p_{\max}$	1.45	1.32	1.83	2.33	2.00	<b>3.23</b>
	$m_{\max}$	1	1	1	2	1	1
	$p_{\min}$	1.18	1.11	1.25	1.67	1.00	1.60
	$m_{\min}$	3	8	7	9	10	9

TABLE IV  
VALUES OF  $p(\text{ILU}(0)\text{GMRES}(m), \mathbf{Q})$ ,  $p(\text{IWZ}(0)\text{GMRES}(m), \mathbf{Q})$ ,  $p(\text{ILU3GMRES}(m), \mathbf{Q})$  FOR THE MATRIX  $\mathbf{Q}$  OF CASE II FOR DIFFERENT VALUES  $m$

$m$	$p(\text{IWZ}(0)\text{G}-(m))$	$p(\text{ILU}(0)\text{G}-(m))$	$p(\text{ILU3G}-(m))$
5	<b>5.07</b>	<b>7.89</b>	<b>0.45</b>
14	5.00	5.00	0.25
25	4.50	3.00	0.16
29	3.50	2.33	0.15
33	3.00	2.00	0.19
41	2.00	1.33	0.17
49	1.50	1.00	0.14
61	1.50	1.50	0.16

$p(\text{IWZ}(0)\text{GMRES}(m), \mathbf{Q})$ ,  $p(\text{ILU3GMRES}(m), \mathbf{Q})$  for the matrix from Case II. In that table we omit  $\mathbf{Q}$  because there is only one such a matrix.

From Tables III and IV it can be concluded the following.

- With the growth of parameter  $m$  (where  $m$  changes from 1 to 10) the value of  $p$  for IWZ(0) and for ILU(0) decreases in both cases.
- IWZ(0) and ILU(0) improve (that is: decreases) the number of restarts for both cases — even 8 times.
- The ILU3 preconditioner in Case I always reduces the number of restarts. However, in Case II it always spoils the performance.
- IWZ(0) and ILU(0) behave similarly for both cases.

### B. Convergence rate of GMRES( $m$ )

1) *Case I:* Fig. 5 presents the relationship between the number of iterations and the achieved accuracy  $\text{acc}(\mathbf{Q}, i)$  (see (10)) for the matrices with ID 3 and 4 (from Group A and B, respectively) for the methods GMRES( $m$ ), IWZ(0)GMRES( $m$ ), ILU(0)GMRES( $m$ ) and ILU3GMRES( $m$ ) for two selected values of parameter  $m$ . The values of  $m$  were chosen after the analysis of Table III and Fig. 4. The maximal iteration speed-up is for  $m = 1$  (regardless of the preconditioner) and the minimal iteration speed-up is usually for  $m = 10$ .

Fig. 5 shows that the higher value of the parameter  $m$ , the more rapidly convergent is the method GMRES( $m$ ). Analogously, the higher value of the parameter  $m$  means that IWZ(0)GMRES( $m$ ), ILU(0)GMRES( $m$ ) and

ILU3GMRES( $m$ ) methods are faster convergent regardless of the matrix.

The convergence curve  $acc(\mathbf{Q}, i)$  as a function of  $i$  is almost of the same shape for any particular parameter  $m$  for the GMRES( $m$ ) method and the IWZ(0)GMRES( $m$ ), ILU(0)GMRES( $m$ ) and ILU3GMRES( $m$ ) methods — only for the preconditioned GMRES( $m$ ) methods the curve is shifted upwards. It means that these methods are faster convergent than the GMRES( $m$ ) method.

The best preconditioner for GMRES( $m$ ), regardless of the parameter  $m$  and of the properties of the matrix, is IWZ(0).

For the the matrix of ID 3 from Group A and  $m = 10$  (Fig. 5, higher middle) all the investigated preconditioners give similar results.

2) *Case II*: Fig. 6 shows the relationship between the number of iterations and the accuracy  $acc(\mathbf{Q}, i)$  (10) for the GMRES( $m$ ) method and the IWZ(0)GMRES( $m$ ), ILU(0)GMRES( $m$ ), ILU3GMRES( $m$ ) methods for  $m = 5$  and  $m = 49$ . The values of  $m$  was based on Table IV — the maximal iteration speed-up (regardless of the preconditioner) is usually for  $m = 5$  and minimal — for  $m = 49$ .

Fig. 6 and Table IV show the following conclusions.

- The preconditioner ILU3 is completely not fitted for Case II, worsening the accuracy of the method GMRES( $m$ ). As we noticed in the end of Section III-B, ILU3 reduces to the Jacobi preconditioner for such banded matrices as in Case II and it causes such a poor accuracy, because the Jacobi preconditioner is rather little effective.
- The best preconditioner is ILU(0). However, as  $m$  grows, the iteration speed-up declines.

For the matrix of Case II it is not always the case, that IWZ(0) improves the convergence rate, because that matrix has a banded structure. ILU(0), being diagonally arranged, seems to be more consistent with the matrix's structure than IWZ(0) — the structure of the latter “crosses” the structure of the original matrix and thus spoils it.

## V. CONCLUSION

Those numerical experiments helped us understand the effect of incomplete preconditioners on the convergence of preconditioned Krylov subspace methods — like GMRES( $m$ ).

Using IWZ(0) and ILU(0) improves the convergence of the GMRES( $m$ ) method (that is, decreases the number of the iterations needed to achieve the required accuracy — even 8 times) used to solve sparse linear equation systems connected to Markov chains.

Both preconditioners — IWZ(0) and ILU(0) — work similarly in terms of the iteration speed-up for the described method and matrices. However, ILU3 appears to be completely useless in the presented cases.

The rate of the convergence of the projection method GMRES( $m$ ) as well as the preconditioned one does not depend on the size of the matrix  $\mathbf{Q}$ .

The speed of convergence in terms of numbers of iterations (restarts) of GMRES( $m$ ) depends on the structure of the matrix. There were tested matrices for two different cases and

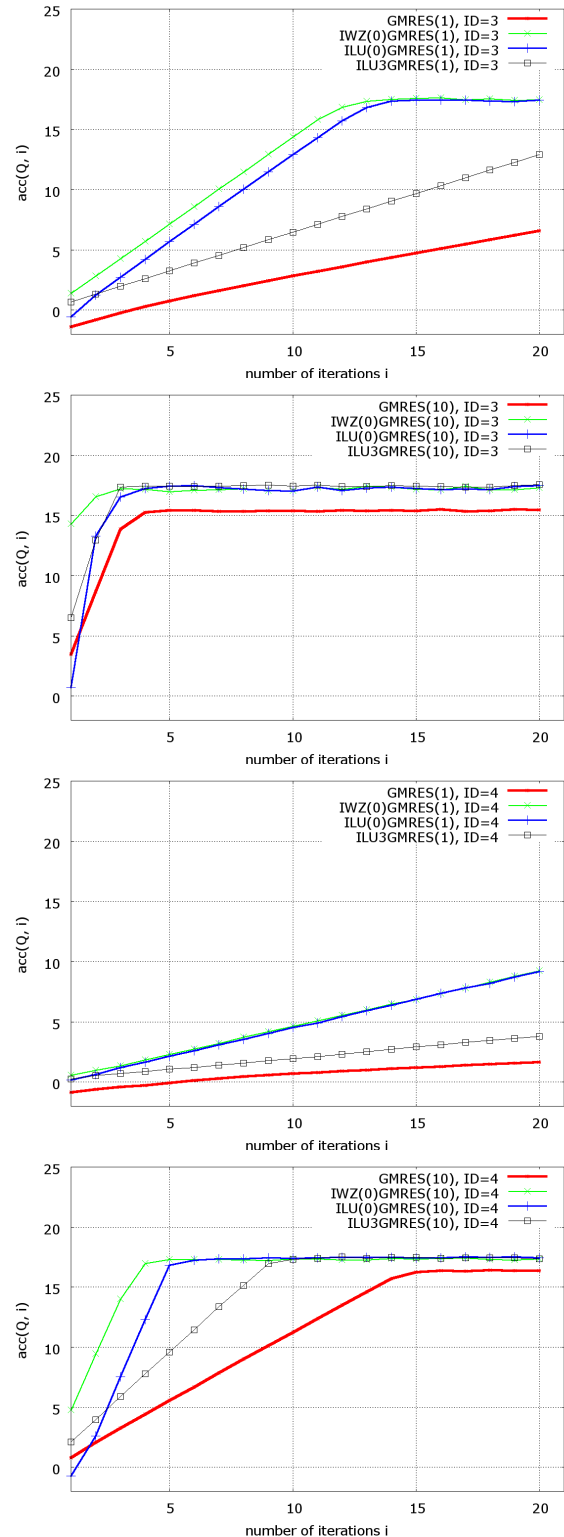


Fig. 5. The plot of the achieved accuracy  $acc(\mathbf{Q}, i)$  as a function of  $i$  for various matrices and values of  $m$ : matrix ID = 3 of Group A,  $m = 1$  (top); matrix ID = 3 of Group A,  $m = 10$  (higher middle); matrix ID = 4 of Group B,  $m = 1$  (lower middle); matrix ID = 4 of Group B,  $m = 10$  (bottom)

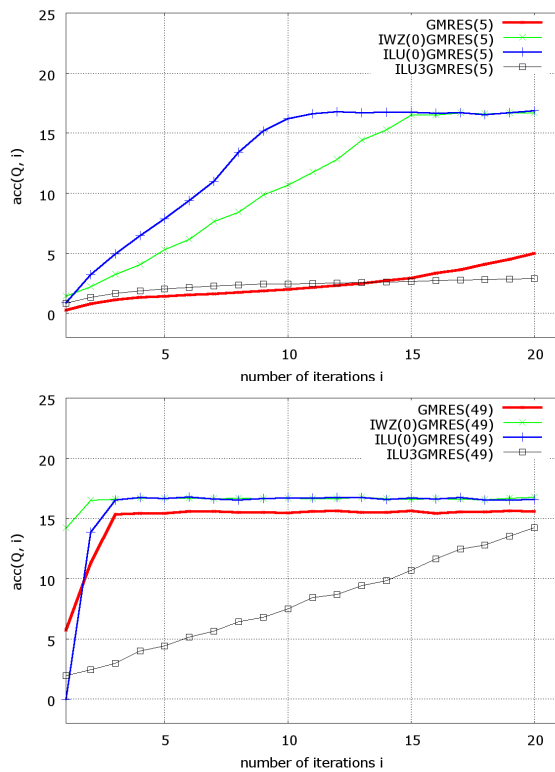


Fig. 6. The plot of the achieved accuracy  $acc(\mathbf{Q}, i)$  as a function of  $i$  for the matrix of Case II for various values of the parameter  $m$ :  $m = 5$  (top);  $m = 49$  (bottom)

they were characterized by the fact that the matrices from case I had no structure, the matrix from case II had a really clear structure.

The matrices of Case I were further distinguished by their density, given by  $d = nz/n$ . And the set of the matrices which had a low value of the parameter  $d$  ( $d < 8$ ) gave slower convergence and they require additional techniques to improve the rate of the convergence. This technique could be some

more complicated preconditioning.

## REFERENCES

- [1] M. Benzi, B. Ucar: Block Triangular preconditioners for M-matrices and Markov chains. *ETNA (Electronic Transactions on Numerical Analysis)*, Vol. 26, pp.209–227, 2007.
- [2] M. Benzi, B. Ucar: Product preconditioning for Markov chain problems, *Proceedings of the 2006 Markov Anniversary Meeting (Charleston, SC, 12-14 June 2006)*, Boson Books, Raleigh, NC, 2006, pp. 239–256.
- [3] N. I. Buleev: A numerical method for solving two-dimensional diffusion equations. *At. Energ.* 6 (1959), p. 338. [In Russian.]
- [4] N. I. Buleev: A numerical method for solving two- and three-dimensional diffusion equations. *Mat. Sb.* 51 (1960), p. 227. [In Russian.]
- [5] B. Bylina, J. Bylina: Incomplete WZ decomposition algorithm for solving Markov chains, *Journal of Applied Mathematics*, vol. 1 (2008), n. 2, p. 147–156.
- [6] B. Bylina, J. Bylina: The experimental analysis of GMRES convergence for solution of Markov chains, *Proceedings of the International Multi-conference on Computer Science and Information Technology*; October 18–20, 2010, Wisięta, Poland, 5 (2010), ISSN 1896-7094, ISBN 978-83-60810-27-9, IEEE Catalog Number CFP1064E-CDR, pp. 281–288.
- [7] T. Dayar, W. J. Stewart.: Comparison of partitioning techniques for two-level iterative solvers on Large, Sparse Markov chains, *SIAM Journal on Scientific Computing* 21, p. 1691 (2000)
- [8] C. G. J. Jacobi: Über eine neue Auflösungsart der bei der Methode der kleinsten Quadrate vorkommenden linearen Gleichungen. *Aston. Nachrichten* 22 (1845), p. 297.
- [9] D. S. Kershaw: The incomplete Cholesky conjugate gradient method for the iterative solution of systems of linear equations. *J. Comput. Phys.* 26 (1978), p. 43.
- [10] J. A. Meijerink, H. A. van der Vorst: An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Math. Comput.* 31 (1977), p. 148.
- [11] P. K. Pollett, D. E. Stewart: An Efficient Procedure for Computing Quasi-Stationary Distributions of Markov Chains with Sparse Transition Structure, *Advances in Applied Probability* 26 (1994), p. 68.
- [12] C. J. Ridler-Rowe: On a Stochastic Model of an Epidemic, *Advances in Applied Probability*, vol. 4, 1967, p. 19–33.
- [13] Y. Saad, M. H. Schultz: GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems, *SIAM Journal of Scientific and Statistical Computing*, 7, 1986, p. 856–869.
- [14] W. Stewart: *Introduction to the Numerical Solution of Markov Chains*, Princeton University Press, Chichester, West Sussex 1994.
- [15] R. S.Varga: Factorizations and normalized iterative methods. In: *Boundary Problems in Differential Equations* (ed. by R. E. Langer), Univ. Wisconsin Press, Madison (1960).
- [16] P. Yalamov, D. J. Evans: The WZ matrix factorization method, *Parallel Computing* 21 (1995), p. 1111.



# Cache-Aware Matrix Multiplication on Multicore Systems for IPM-based LP Solvers

Mujahed Eleyat  
Miriam AS, Halden &  
IDI, NTNU, Trondheim  
Norway  
Email: mujahed@miriam.as

Lasse Natvig, Jørn Amundsen  
Department of Computer and Information Science (IDI)  
Norwegian University of Science and Technology (NTNU)  
Trondheim, Norway  
Email: lasse@idi.ntnu.no, jorn.amundsen@ntnu.no

**Abstract**—We profile GLPK, an open source linear programming solver, and show empirically that the form of matrix multiplication used in interior point methods takes a significant portion of the total execution time when solving some of the Netlib and other LP data sets. Then, we discuss the drawbacks of the matrix multiplication algorithm used in GLPK in terms of cache utilization and use blocking to develop two cache-aware implementations. We apply OpenMP to develop parallel implementations with load balancing. The best implementation achieved a median speedup of 21.9 when executed on a 12-core AMD Opteron.

**Keywords:** sparse matrix multiplication, cache optimization, interior point methods, multicore systems.

## I. INTRODUCTION

**D**URING recent years, processor designers have moved away from uniprocessor systems to multicore systems. This shift is mainly due to manufactures inability to continue enhancing the performance of single-core processors [1]. Increasing clock speeds requires higher voltage and causes, consequently, too much heat to dissipate. On the other hand, using deeper pipelines and other advanced architectural techniques have yielded decreasing improvements. In addition, and due to the speed gap between main memory and the processing cores, there has been more demand for an efficient cache system to allow exploiting the collective processing power [2]. As a result, multicore programmers need not only to provide a parallel implementation of the application, but they also have to take cache utilization into consideration for efficient utilization of the multicore system. Techniques to reduce cache and TLB misses depend on the application memory access pattern, for example, tiling/blocking [3] is the most popular method for applications with poor exploitation of temporal locality.

A Linear Programming (LP) solver is one of many compute-intensive applications that could benefit from the high multicore performance. It works as a decision maker that chooses values of many variables to achieve a goal (maximum profit, best resource allocation, etc.) while satisfying a set of constraints that are specified as mathematical equalities and inequalities [5]. If we have  $m$  constraints and  $n$  variables, the LP-problem in standard form can be written as:

$$\text{minimize } z = c^T x, \text{ subject to } Ax = b, x \geq 0,$$

where  $x$  is an  $n$ -dimensional column vector,  $c^T$  is an  $n$ -dimensional row vector,  $A$  is a  $m \times n$  matrix, and  $b$  is an  $m$ -dimensional column vector.

Solving LP problems in an efficient way is crucial for industrial and scientific fields, especially since an application might need to solve large problems and/or a long sequence of problems. For example, Miriam Regina, a network gas flow simulator developed by Miriam AS [4], solves thousands of LP problems to make a single allocation of gas flow in the network. On the other hand, it needs to solve bigger LP instances for the simulation to cover large networks that span the national boundaries.

The motivation for investigating matrix multiplication in interior point methods (IPM) [5, 6] is that it takes a large fraction of the total computation time when solving some of the data sets. In addition, it is a special form of multiplication of the form  $ADA^T$ , where  $A$  is a sparse matrix,  $D$  is a diagonal matrix, and  $A^T$  is the transpose of  $A$ . Moreover, sparse multiplication is a form of irregular computation that is much more challenging to accelerate than dense multiplication. On the other side, the structure of the multiplication result is constant through all IPM iterations, a fact that may be used to enhance its computation performance.

In this paper, we profile serial GLPK [7], an open source LP solver, and present empirical results showing that matrix multiplication takes a relatively long time to compute for some Netlib and miscellaneous problems [8, 9]. We also analyse memory access patterns of sparse multiplication and develop cache-aware algorithms that reduce the rate of cache and TLB misses. Moreover, a parallel version is also provided while trying to exploit the cache hierarchy of the multicore system.

The paper is organized as follows: Section II gives a brief overview of the AMD Opteron compute node used. Then, compute-intensive parts of the LP solver are introduced in section III. Section IV explains GLPK implementation of sparse matrix multiplication while section V describes techniques to enhance cache utilization. We present related work in section VI and conclude with experimental results and future work.

## II. MULTI-CORE HARDWARE

Introduced in 2009, the 64-bit Istanbul processor is the first 6-core AMD Opteron<sup>®</sup> processor and is available for 2-, 4-, and 8-socket systems, with clock speeds ranging from 2.0 to 2.8 GHz [10].

Fig. 1 shows a simplified block diagram. The processor has six cores, three levels of cache, a crossbar connecting the cores, the System Request Interface, the Memory controller, and three HyperTransport 3.0 links. The memory controller supports DDR2 memory with a bandwidth of up to 12.8 GB/s. In addition, the HyperTransport 3.0 links provide an aggregate bandwidth of 57.6 GB/s and are used to allow communication between different Istanbul processors. Each core has two levels of cache, a 512 KB L2 cache, 64 KB data cache and 64 KB instruction cache. However, all cores share a 6 MB L3 cache.

AMD Opteron multiprocessor systems are based on the cache coherent Non-Uniform Memory Access (ccNUMA) architecture. Each processor is connected directly to its own dedicated memory banks and it uses HT links to communicate with I/O busses and the other processor(s). Fig. 2 shows a block diagram of a 2-socket system.

## III. GLPK AND IPM COMPUTATIONAL KERNELS

The GLPK (GNU Linear Programming Kit) package is a set of ANSI C routines contained into a callable library and intended for solving large-scale linear programming, mixed integer programming, and other related problems [7]. GLPK has routines for solving LP problems using either simplex or one of the primal-dual interior point methods (IPMs), namely the Mehrotra's predictor-corrector method [6]. This method, as well as other primal-dual interior point methods, keeps repeating a set of matrix operations, until it converges to an optimal solution. Every iteration of the algorithm includes the following computations [11]:

- 1) Sparse matrix-matrix multiplication of the form  $S = PAD(PA)^T$ , where  $P$  is a permutation matrix stored

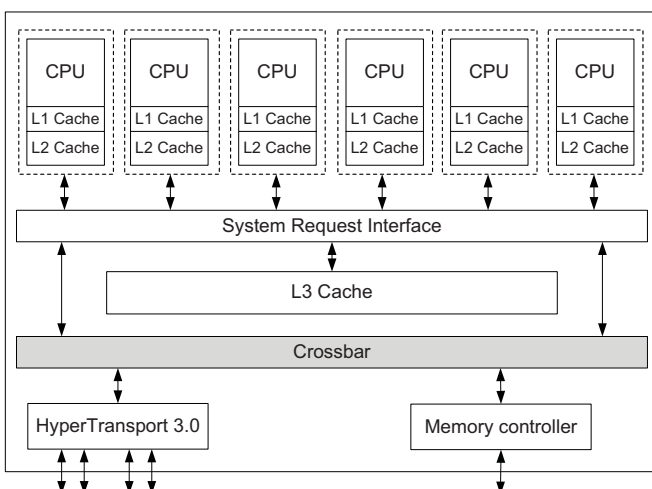


Fig. 1. Simplified block diagram of an AMD Opteron Istanbul processor.

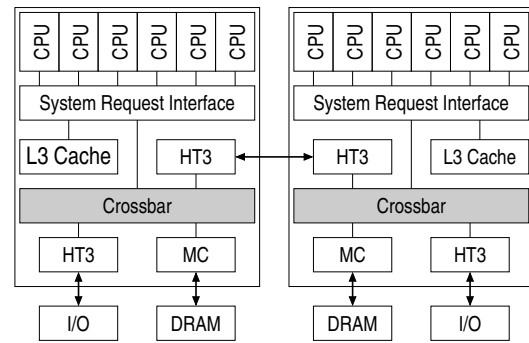


Fig. 2. Block diagram of a 2-socket system

as a single dimensional array,  $A$  is the sparse working constraint matrix stored using the CRS format,  $D$  is a diagonal matrix stored using a single dimensional array, and  $(PA)^T$  is the transpose of matrix  $PA$ . The output matrix  $S$  is a symmetric positive definite matrix.

- 2) Cholesky factorization of a symmetric sparse matrix  $S$ , the result of step 1, into  $LL^T$  where  $L$  is the lower factor matrix and  $L^T$  is the transpose of  $L$ .
- 3) Backward/forward solving using Cholesky factors.

### A. Compressed row storage CRS

Since most practical problems are very sparse, GLPK uses compressed row storage (CRS) to store the constraint matrix and other matrices used in the IPM algorithm. CRS is a general storage format that makes no assumptions about the sparsity structure of the matrix [12]. As shown in Table I, CRS uses three contiguous arrays to store a sparse matrix  $A$ :  $A_{\text{val}}$  stores all nonzero elements row by row,  $A_{\text{ind}}$  holds the column indices of the nonzeros, and  $A_{\text{ptr}}$  holds the offset of each row into  $A_{\text{val}}$ . CRS is usually used to access the matrix row by row, while another format called compressed column storage (CCS) is used when a column by column access is needed. Similar to CRS, CSS uses three arrays to store the matrix, however, it stores the nonzeros column by column and  $A_{\text{ptr}}$  have pointers to the start of columns in  $A_{\text{val}}$ .

### B. Time analysis of IPM computational kernels

Time analysis of serial IPM-based GLPK has been performed when solving big data sets taken from Netlib and the BPMPD website [9]. Size, number of nonzeros, and sparsity, fraction of nonzero elements of the matrix, of each data set

TABLE I  
A MATRIX  $A$  AND ITS CRS STORAGE

$$A = \begin{bmatrix} 0 & 3 & 0 & 0 & 1 \\ 4 & 1 & 0 & 0 & 0 \\ 0 & 5 & 9 & 2 & 0 \\ 6 & 0 & 0 & 5 & 3 \\ 0 & 0 & 5 & 9 & 0 \end{bmatrix}$$

$A_{\text{val}}$	3	1	4	1	5	9	2	6	5	3	5	9
$A_{\text{ind}}$	2	5	1	2	2	3	4	1	4	5	3	4
$A_{\text{ptr}}$	1	3	5	8	11	13	(nonzero count + 1)					

are shown in Table II. Results of time analysis are presented in Table III and they show that Cholesky factorization and sparse matrix multiplication of the form  $ADA^T$  are the two most computationally expensive tasks of the solver. However, a few of the data sets, namely BPMPD, CZPROB, and NEMSEMM1, show that a considerable amount of time is spent executing other parts of the code. These results show the importance of accelerating Cholesky factorization and sparse multiplication in order to enhance the performance of IPM based LP solvers.

IV. ORIGINAL IMPLEMENTATION OF GLPK MATRIX MULTIPLICATION AND CACHE PROBLEMS

As mentioned earlier, the sparse matrix product  $S$  in IPM has the form  $S = PAD(PA)^T$ . This product is computed in two phases: symbolic and numeric. The symbolic phase is performed once and is used to determine the nonzero structure of  $S$  for use in the numeric phase.

The numeric phase, which is implemented based on Gustavson’s algorithm [15], is executed every iteration to determine the numeric values of  $s_{i,j \geq i}$  of  $S$ . Algorithm 1 shows the pseudocode of the GLPK serial implementation of  $S = PAD(PA)^T$  where  $A$  is an  $m \times n$  matrix and  $P$  is stored in a permutation vector  $\pi$ . The matrix product is computed row by row (line 1) and the permutation is applied at lines 2 and 5. If row  $k$  of  $S$  has  $n_{nz}$  nonzeros, then its computation requires multiplying row  $i$  ( $i_p$  after permutation) of  $A$  by  $D$  and by other  $n_{nz}$  rows of  $A$ . Since rows have different sparsity, row  $i_p$  is decompressed into a vector  $w$  (line 3) as illustrated in Algorithm 2. Each nonzero in row  $i$  of  $S$  is finally obtained by performing a dot product between  $Dw$  and a row of  $A$  (line 6).

The GLPK implementation of  $S = PAD(PA)^T$  suffers from the following problems with regard to cache utilization:

- Because of the sparsity of  $A$ , a small fraction of the values in  $D$  and  $w$  need to be read for the computation of each nonzero of  $S$ . However, these values are scattered irregularly over large vectors,  $D$  and  $w$ , that don’t have room in L1 and L2 cache. Such irregular access pattern will cause a high cache miss rate.

TABLE II  
INFORMATION ABOUT THE TEST DATA SETS

Problem name	Column count	Row count	Nonzeros	Sparsity $\times 10^{-4}$
FIT2D	10525	21024	150042	6.8
CZPROB	929	3333	10022	32.4
NEMSEMM1	5668	74151	1036227	24.7
WORLD	47259	79053	220891	0.6
NSCT2	23003	37563	697738	8.1
BPMPD	33841	1144020	3450992	0.9
OLIVIER	11144	22977	108562	4.2
BASILP	9872	14286	596697	42.3
DFL001	6084	12243	35658	4.8
QAP12	3192	8856	38304	13.5
QAP15	6330	22275	94950	6.7

TABLE III  
PROFILING SERIAL GLPK

Problem name	$ADA^T$ (%)	Cholesky (%)	Bck/fwd solver (%)	Others (%)
FIT2D	98.8	0.3	0.1	0.8
CZPROB	69.1	7.7	2.2	21.0
NEMSEMM1	66.3	13.7	1.0	19.0
WORLD	6.3	81.4	4.8	7.5
NSCT2	6.1	92.6	0.4	0.9
BPMPD	34.9	13.8	1.5	49.8
OLIVIER	33.4	57.5	3.0	6.1
BASILP	11.4	85.8	0.8	2.0
DFL001	0.2	98.7	0.8	0.3
QAP12	0.1	99.2	0.6	0.2
QAP15	0.0	98.7	0.3	1.0

**Algorithm 1** Serial implementation of  $S = PAD(PA)^T$ , using  $A_\alpha$  for row  $\alpha$  of  $A$ .

```

1: for  $i = 1 \rightarrow m$  do
2:    $i_p = \pi(i)$ 
3:    $w = (A_{i_p})^T$  {see Algorithm 2}
4:   for  $j = i \rightarrow m$  and  $s_{ij} \neq 0$  do
5:      $j_p = \pi(j)$ 
6:      $s_{ij} = A_{j_p} Dw$ 
7:   end for
8: end for
    
```

- Although the rows of  $A$  have small number of values that are stored contiguously, the permutations makes it difficult to benefit from data locality and might cause much TLB misses for matrices with high number of nonzeros [13].

V. CACHE-AWARE MATRIX MULTIPLICATION

Trying to exploit cache and avoid the problems mentioned in the previous section, we use 1D and 2D partitioning and develop techniques to avoid the overhead of accessing zero blocks and zero block rows. Both extensions of the original algorithm avoid the negative effect of permutation by performing it during the blocking phase, i.e. the rows are permuted in memory before they are split into several blocks. This allows more uniform access to rows of partitions during multiplication. The new algorithms are explained in the following subsections.

A. 1D partitioning of the matrix  $A$

The method is based on a vertical partitioning of  $PA$  into blocks  $A^{(1)}, A^{(2)}, \dots, A^{(v)}$ , where  $v$  is the number of vertical partitions. Moreover, partitioning is made once since  $A$  is

**Algorithm 2** Decompression of a row of matrix  $A$  into  $w$ .

```

1: for  $k = A_{\text{ptr}}(i_p) \rightarrow A_{\text{ptr}}(i_p + 1)$  do
2:    $l = A_{\text{ind}}(k)$ 
3:    $w(l) = A_{\text{val}}(k)$ 
4: end for
    
```

constant and only  $D$  changes through the IPM iterations. In addition, each of the blocks is stored in memory as an independent matrix using CRS. Algorithm 3 shows the pseudocode of the 1D algorithm.  $D$  and  $w$  are accessed in smaller chunks  $D^{(1)}, D^{(2)}, \dots, D^{(v)}$  and  $w'$  whose size depends on the width of  $A$ -partitions. The goal is to have  $D^{(\cdot)}$ 's and  $w'$  that can fit into L1/L2 cache and be reused through loop iterations at line 5.

One of the drawbacks of 1D partitioning is that many of the partitioned rows have no elements (zero rows) which waste cycles on loading and comparing  $A_{ptr}$  values. Fig. 3A shows an example of a vertical partitioning where 60% of the blocked rows have no elements. In fact, the percentage of zero rows is much higher in real problems as the matrices are much more sparse than the one shown in Fig. 3A. Another drawback is the extra storage of ptr array of each partition.

To avoid wasting time on zero partition rows, a higher level of compressed storage is used to efficiently access the nonzero rows of partitions. The matrix, as shown in Fig. 3A, is treated as an  $m \times v$  matrix and a new second level of CRS structure (only ptr and ind) is added and used to access nonzero partition rows. This adds more to storage requirements, but has a good effect on performance. The added  $Parts_{ptr}$  and  $Parts_{ind}$  are shown Fig. 3B, and the associated multiplication algorithm is shown in Algorithm 4.

### B. 2D partitioning of the matrix $A$

Data blocking is a well known technique to utilize data spatial locality [20] and it is well suited for dense matrix multiplication. We try to apply the same technique to sparse matrix multiplication by dividing matrix  $A$  into  $M \times N$  blocks and consequently matrix  $S$  into  $M \times M$  blocks all stored using CRS. Computation of an  $S$  block is achieved by multiplying two rows of  $A$  blocks as shown in Algorithm 5. Blocks have different sparsity and many  $A$  and  $S$  blocks may have no elements (zero blocks). Different sparsity of different blocks is a reason why blocking is not as efficient as when dealing with dense matrices. However, trying to exploit the existence of zero blocks we add extra information to  $S$  blocks as shown in the following:

- Each block of  $S$  has an array that stores the indices of nonzero rows.

---

**Algorithm 3** Vertically partitioned implementation of  $S = PAD(PA)^T$ , using  $A_{\alpha}^{(p)}$  for row  $\alpha$  of partition  $A^{(p)}$ .

---

**Require:**  $A \leftarrow PA$  {performed when partitioning}

```

1: for  $i = 1 \rightarrow m$  do
2:   for partition  $p = 1 \rightarrow v$  do
3:      $w' = (A_i^{(p)})^T$ 
4:     for  $j = i \rightarrow \text{mand}_{s_{ij}} \neq 0$  do
5:        $s_{ij} += A_j^{(p)} D^{(p)} w'$ 
6:     end for
7:   end for
8: end for
```

---

**Algorithm 4** Extension of Algorithm 3 with second level CRS

**Require:**  $A \leftarrow PA$  {performed when partitioning}

```

1: for  $i = 1 \rightarrow m$  do
2:   for  $t = Parts_{ptr}(i) \rightarrow Parts_{ptr}(i+1)$  do
3:     partition  $p = Parts_{ind}(t)$ 
4:      $w' = (A_i^{(p)})^T$ 
5:     for  $j = i \rightarrow \text{mand}_{s_{ij}} \neq 0$  do
6:        $s_{ij} += A_j^{(p)} D^{(p)} w'$ 
7:     end for
8:   end for
9: end for
```

---

**Algorithm 5** 2D partitioning of matrix  $A$

```

1: for  $I = 1 \rightarrow M$  do
2:   for  $J = 1 \rightarrow M$  do
3:     for  $K = 1 \rightarrow N$  do
4:        $S_{I,J} += A_{I,K} A_{J,K}$ 
5:     end for
6:   end for
7: end for
```

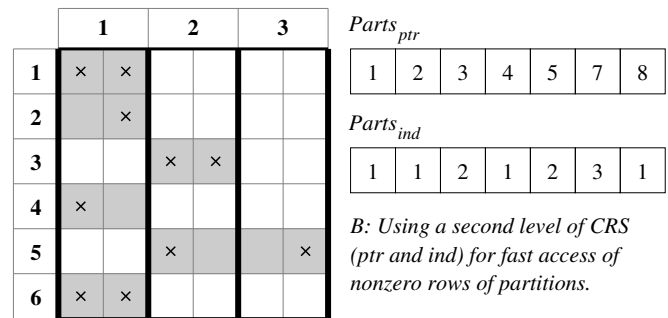
---

- Each block of  $S$  has an array of indices of participating pairs of  $A$  blocks.

The goal of the first point is to allow utilization of the already known  $S$  matrix structure. However, the second point aims at avoiding accessing  $A$  blocks that don't participate into computation of an  $S$  block. Two pairs of  $A$  blocks participate in the computation if their product produces one or more nonzeros, which is simply determined by checking if they have at least one common index of nonzero columns. We determine participating blocks just after the symbolic phase and use it through all IPM iterations. Suppose that matrix  $A$  has  $M \times N$  blocks then  $S$  has  $M \times M$  blocks as shown in Algorithm 5.

### C. Parallel computation of $S$ and load balancing

Parallelization of original GLPK implementation and our implementations of the sparse matrix multiplication have been achieved with OpenMP, mainly for loop parallelization. The original GLPK implementation is parallelized by parallelizing the for loop shown at line 1 in Algorithm 1, causing each core



A: Blocking  $A$  into 3 vertical partitions

Fig. 3. 1D partitioning of matrix  $A$

B: Using a second level of CRS (ptr and ind) for fast access of nonzero rows of partitions.

Listing 1. Parallel block based matrix multiplication

```

#pragma omp parallel for
for row = 1 to M
  for col = 1 to M
    ...
A. Parallel 2D algorithm without load balancing

shares = assign_shares(num_cores, num_nonzeros[]);
#pragma omp parallel for
for i = 1 to num_cores
  for row = shares[i].from to shares[i].to
    for col = 1 to M
      ...
B. Parallel 2D algorithm with load balancing

```

to compute a chunk of  $S$  rows. Similarly, 1D partitioning of  $A$  also uses the same principle as the outer loop iterates over rows of  $S$ . Finally, multiplication based on 2D partitioning of  $A$  is parallelized by parallelizing the for loop shown at line 1 of Algorithm 5, causing each core to compute a chunk of rows of  $S$  blocks.

Due to different levels of sparsity within the same matrix in general, and as will be seen in the performance results section, our parallel implementation suffers from load imbalance. To address this, we divide  $S$  into a number of shares that equals the number of cores, and assign one core to each share. An ideal share would be any share whose number of nonzeros equals the number of nonzeros in  $S$  divided by the number of cores. Therefore, we try to assign shares such that they differ as less as possible from an ideal share. To force a core to compute one share, we add an outer loop over shares and apply the `omp parallel for` construct to the new added loop as shown in Listing 1.

A share is composed of a number of consecutive  $S$  rows in the original and 1D algorithms, but it is made of a number of consecutive rows of blocks in the 2D algorithm, making it harder to determine shares that are close to an ideal share. For the 2D algorithm, we try to determine shares that are bigger than an ideal share within a specified tolerance. If  $t$  denotes tolerance and  $d$  denotes number of nonzeros in an ideal share, then a share can have up to  $(1+t)d$  nonzeros. In our implementation, we start trying a 5% tolerance and decide all the shares except the last one. If the size of the last share doesn't satisfy the tolerance constraint, we keep increasing the tolerance by 5% and repeat the algorithm until all shares respect the tolerance constraint.

## VI. RELATED WORK

Our implementations, although not intended for general sparse matrix multiplication, are based on the classical Gustavson algorithm [15] using compressed row storage of matrices. That algorithm is also used in Csparse [19] and Matlab [21] and is proven to be optimal with respect to number

of operations and storage space of general sparse matrices.

Algorithms for sparse multiplication are developed with focus on optimizing number of operations and storage requirements, however they only perform better than Gustavson's algorithm when working on certain class of matrices. For example, Park et al. [17] built an efficient algorithm that is based on a compact storage of banded and triangular matrices. On the other hand, Buluç et al. [16] introduced what they called the doubly compressed sparse column (DCSC) which uses less space than compressed column storage (CSC) for storing hypersparse matrices, matrices where number of nonzeros is less than the dimension of the matrix. Such matrices may be the result of a 2D partitioning of sparse matrices for parallel processing.

To our knowledge, Sulatycke et al. are the only researchers who presented sparse matrix algorithms that take efficiency of caches into consideration [14]. Their cache aware algorithms are based on interchanging loops of a standard multiplication algorithm. Moreover, they presented a parallel version that is based on static and dynamic splitting of matrix rows among several threads. However, their experiments were conducted on up to 1000 x 1000 10% sparse matrices, which are much smaller and less sparse than those tested in this paper.

Most recent research about sparse matrix multiplication have been performed by Buluç et al. In [16], they discussed the scalability limitations of matrix multiplication on thousands of processors. Moreover, they developed a sequential hypersparse matrix multiplication algorithm using the DCSC sparse storage to overcome the presented limitations. Parallel implementation was simulated by dividing input matrices using 2D blocking decomposition, excluding other costs like updates and parallelization overheads. Based on their work in [22], load imbalance, hiding communication costs, and additions of submatrices, are the main challenges of parallelizing sparse multiplication. In addition, they have also analysed the scalability of using 1D and 2D block decomposition to divide the work among the processors and show analytically and experimentally that the 2D based algorithms are more scalable than those based on 1D blocking.

## VII. PERFORMANCE RESULTS AND CONCLUSION

### A. Cache aware matrix multiplication

Our experiments are performed on a 2 x 6 cores AMD Opteron (Istanbul 2431) compute node. All code including GLPK 4.43 is compiled with GCC 4.4.3, with optimization level 3 (-O3). Moreover, execution time of only the first IPM iteration has been measured when solving each of the data sets because iterations caused by solving one data set take the same amount of execution time. Table IV reports the execution time of original GLPK implementation and the new two implementations of the serial matrix multiplication, executed on a single core. Speedup is calculated taking the original implementation as a baseline. The following can be concluded:

- 1) The new 1D and 2D algorithms execute faster than the original one for all data sets. However, FIT2D is

accelerated much more than other data sets. This can be explained by the unique nonzero structure of its  $PA$  as shown in Fig. 4. The figure is created by placing a dot in the location of each nonzero element, i.e. the horizontal thin bar in the figure represents a group of adjacent dense rows in the matrix. The nonzero structure of this matrix is special because most rows have two values while the last few rows are dense. The original implementation is slow because it accesses most of the  $D$  values when one of the dense rows in the bottom of  $PA$  is involved, causing much L1 and L2 cache misses. However, the 1D and 2D implementations utilize the cache and improve locality of access as explained in section V.

- 2) Different data sets are accelerated by different values due to the difference in sparsity. Moreover, the distribution of nonzeros is different among different data sets.
- 3) 2D avoids accessing nonzero blocks and blocks whose multiplication doesn't result in any nonzeros, while 1D avoids accessing nonzero rows of partitions. 2D cause more performance when nonzeros are concentrated in chunks causing a lot of nonparticipating blocks to be avoided.

### B. Size of partitions/blocks

Table V shows sizes of partitions/blocks that cause optimal speedup for both implementations and for different data sets. The results show that the optimal dimensioning of block-/partitions are more related to the distribution of nonzeros than to the sparsity of data sets. If we fix partition size in the 1D implementation to 100 and the block size in the 2D implementation to  $100 \times 100$ , the speedup of NSCT2 and BAS1LP is reduced by 16% and 14% respectively. However, the speedup of 2D partitioning for CZPROB, NEMSEMM1, NSCT2, BPMPD, OLIVIER, and BAS1LP is reduced by an average of 8%.

### C. Parallel sparse matrix multiplication

The performance of the parallel matrix multiplication for original implementation and our implementations before and after load balancing is shown in Figs. 5-9. The results show the high importance of the load balancing. In addition, they show that problems that have longer serial execution time scale better than those which have relatively lower execution time.

TABLE IV  
SERIAL TIMINGS OF ORIGINAL AND NEW IMPLEMENTATIONS

Problem name	Orig. [s]	1D [s]		Speedup	
		1	3	1	3
FIT2D	2.455	0.0261	0.0227	94.1	108.0
CZPROB	0.004	0.0006	0.0007	6.6	5.7
NEMSEMM1	0.279	0.0783	0.0684	3.6	4.1
WORLD	0.049	0.0292	0.0414	1.7	1.2
NSCT2	2.015	1.6290	1.2490	1.2	1.6
BPMPD	0.433	0.1047	0.1211	4.1	3.6
OLIVIER	0.088	0.0164	0.0180	5.4	4.9
BAS1LP	0.992	0.7724	0.5933	1.3	1.7

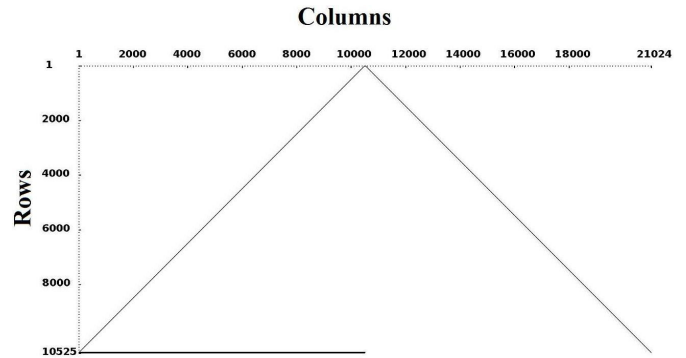


Fig. 4. Nonzero structure of FIT2D after permutation.

Although both implementations show a comparable speedup when executed serially, the later has better speedup when both are executed in parallel. The median speedup of the 1D and 2D implementations is 12.0 and 21.9. Table VI shows the speedup achieved when executing the implementations on 10 cores with the execution time of the original serial algorithm as a baseline. We chose to show the results on 10 cores since some data sets show bad performance when executed on 11 and 12 cores.

To have a more clear view, we show the parallel performance of NEMSEMM1 as an example, in Fig. 10. The figure shows that speedup doesn't increase smoothly with increasing number of cores. This behaviour is due to two main reasons. First, a strange varying OpenMP overhead is observed. It is measured as the difference between the matrix multiplication time and the execution time of the thread that takes most time to finish computing its share. Second, because nonzeros can be concentrated in a small part(s) of the matrix, using nonzeros to divide the shares among threads doesn't always guarantee that load balancing will be improved. For example, in the 2D algorithm, one thread might be responsible for computing many very sparse blocks, while a second one might be responsible for computing a much lower number of dense blocks. The overhead caused by these two reasons can have a relatively big effect on performance as shown when using 11 and 12 cores.

## VIII. CONCLUSION AND FUTURE WORK

An efficient LP solver is crucial for many scientific and industrial applications. However, most research has been fo-

TABLE V  
SIZES OF PARTITIONS/BLOCKS

Problem name	Sparsity $\times 10^{-4}$	1D width	2D width x height
FIT2D	6.8	100	100 x 100
CZPROB	32.4	100	100 x 50
NEMSEMM1	24.7	100	100 x 50
WORLD	0.6	100	100 x 100
NSCT2	8.1	1500	100 x 50
BPMPD	0.9	100	100 x 70
OLIVIER	4.2	100	600 x 625
BAS1LP	42.3	400	200 x 50



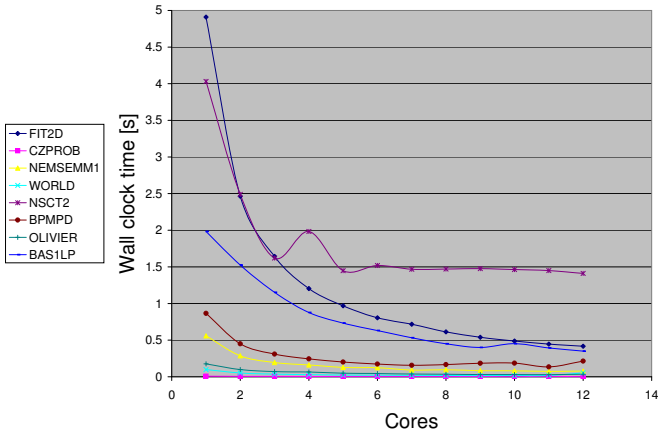


Fig. 5. Parallel original GLPK implementation without load balancing.

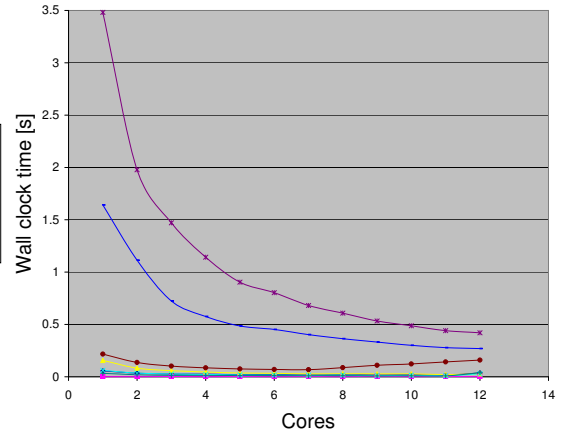


Fig. 7. Parallel 1D algorithm with load balancing.

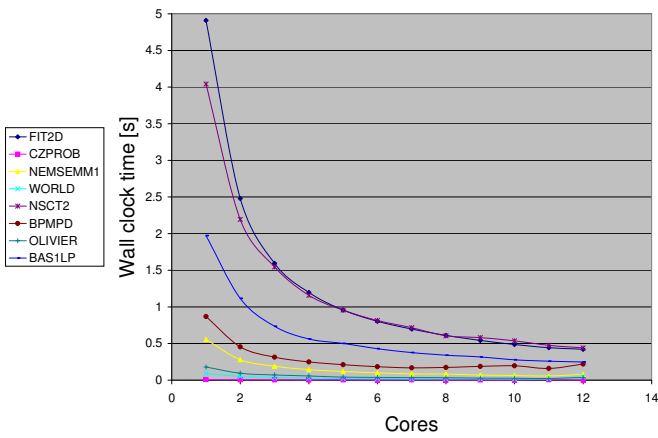


Fig. 6. Parallel original GLPK implementation with load balancing.

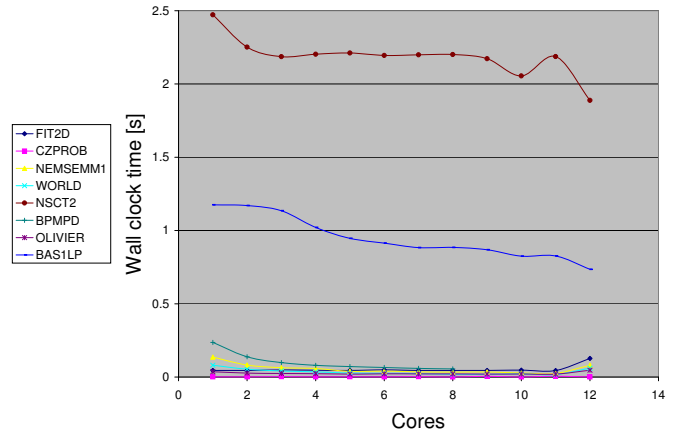


Fig. 8. Parallel 2D without load balancing.

used on an efficient Cholesky factorization since it is the most expensive computation in interior point methods. We showed that, similar to Cholesky factorization, sparse matrix multiplication in IPM-based solvers use a relatively high percentage of the total execution time when solving some big data sets, and proposed two cache-aware implementations of the sparse multiplication algorithm used in GLPK. Moreover, we used OpenMP to parallelize the multiplication and developed

a simple, but efficient technique for load balancing.

Due to many zero rows of very sparse blocks, CRS and CCS are not optimal wrt. space for storing very sparse blocks, but we had to use them for two reasons,

- Block sparsity varies a lot even in the same data set

TABLE VI  
PARALLEL SPEEDUP OF THE TWO ALGORITHMS ON 10 CORES

Problem name	Speedup	
	1D	2D
FIT2D	270.3	374.4
CZPROB	15.6	23.9
NEMSEMM1	19.7	28.8
WORLD	7.4	8.9
NSCT2	8.3	9.5
BPMPD	7.0	24.1
OLIVIER	18.8	19.8
BAS1LP	6.6	10.2
<b>Median Speedup</b>	<b>12.0</b>	<b>21.9</b>

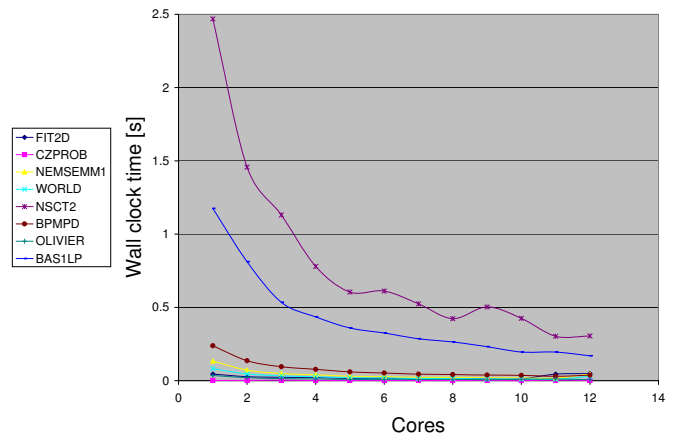


Fig. 9. Parallel 2D with load balancing.



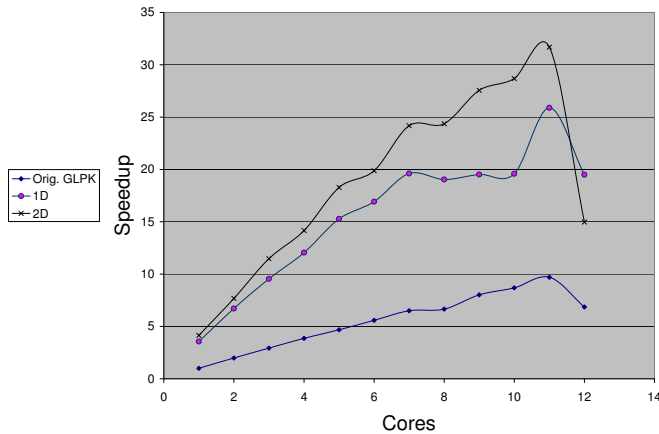


Fig. 10. NEMSEMM1 with load balancing. Original serial execution time is used as a baseline.

- Our algorithms requires very fast access to rows/columns.

It might be possible to use different storage and computation mechanisms for blocks based on their sparsity. One approach to accomplish this is to use 2D partitioning but using larger blocks, and then choose the appropriate storage mechanism and block multiplication procedure based on the sparsity level. In case of dense (or close to dense blocks), another level of blocking can be performed to better utilize the cache.

Since blocking is our main technique of exploiting cache, it is interesting to try our algorithms on other multicore systems that have different cache systems.

#### REFERENCES

- [1] S. H. Fuller and L. I. Millett, "The Future of Computing Performance: Game Over or Next Level," *IEEE Computer*, vol. 44, pp. 31–38, January 2011.
- [2] M. V. Wilkes, "The Memory Gap and the Future of High Performance Memories," *ACM Computer Architecture News*, vol. 29, pp. 2–7, March 2001.
- [3] M. S. Lam, E. E. Rothberg, and M. E. Wolf, "The Cache Performance and Optimizations of Blocked Algorithms," *Proc. 4th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems*, California, pp. 63–74, April 1991.
- [4] Miriam AS, available at <http://www.miriam.as> (accessed September 2010)
- [5] D. G. Luenberger and Y. Ye, "Linear and Nonlinear Programming," *Springer Science*, 3rd ed., N.Y., 2007.
- [6] S. Mehrotra, "On the Implementation of a Primal-Dual Interior Point Method", *SIAM J. on Optim.*, pp. 575–601, 1992.
- [7] GLPK (GNU Linear Programming Kit), available at <http://gnu.org/software/glpk/> (accessed: September 2010).
- [8] The NETLIB LP Test Problem Set, available at <http://www.numerical.rl.ac.uk/cute/netlib.html> (accessed September 2010).
- [9] BPPMD Home Page, available at <http://www.sztaki.hu/~meszaros/bppmd/> (accessed September 2010).
- [10] Paul G. Howard, "Six-Core AMD Opteron processor Istanbul," white paper, Microway Inc., 2009.
- [11] M. Smelyanskiy, V. W. Lee, D. Kim, A. D. Nguyen, and P. Dubey, "Scaling Performance of Interior-Point Method on Large-Scale Chip Multiprocessor System", *Proc. ACM/IEEE Conf. Supercomputing (SC07)*, 2007.
- [12] R. Shahnaz, A. Usman, and I. R. Chughtai, "Review of Storage Techniques for Sparse Matrices", *IEEE INMIC 2005 Conf. Proc.*, pp. 1–7, December 2005.
- [13] K. Kaspersky, "Code Optimization: Effective Memory Usage," *A-List Publishing*, Wayne, Pennsylvania, 2003.
- [14] P. D. Sulatycke and K. Ghose, "Caching Efficient Multithreaded Fast Multiplication of Sparse Matrices," *Proc. Merged Int'l Symp. Par. Proc. and Par. and Distr. Proc.*, pp. 117–124, 1998.
- [15] F. G. Gustavson, "Two Fast Algorithms for Sparse Matrices: Multiplication and Permuted Transposition," *ACM Trans. Math. Software*, vol. 4, pp. 250–269, 1978.
- [16] A. Buluç and J. R. Gilbert, "On the Representation and Multiplication of Hypersparse Matrices," *IPDPS*, IEEE, pp. 1–11, 2008.
- [17] S. C. Park, J. P. Draayer, and S. Q. Zheng, "Fast Sparse Matrix Multiplication," *Comp. Phys. Comm.*, vol. 70, pp. 557–568, 1992.
- [18] R. Yuster and U. Zwick, "Fast Sparse Matrix Multiplication," *ACM Trans. on Algorithms*, vol. 1, pp. 2–13, 2005.
- [19] T. A. Davis, "Direct Methods for Sparse Linear Systems," *Soc. for Ind. and Appl. Math.*, 2006.
- [20] M. Kowarschik and C. Weiss, "An Overview of Cache Optimization Techniques and Cache-Aware Numerical Algorithms," *LNCIS*, vol. 2625, pp. 213–232, 2003.
- [21] J. R. Gilbert, C. Moler, and R. Schreiber, "Sparse Matrices in MATLAB: Design and Implementation," *SIAM J. Matrix Anal. and Appl.*, vol. 13, pp. 333–356, 1992.
- [22] A. Buluç and J. R. Gilbert, "Challenges and Advances in Parallel Sparse Matrix-Matrix Multiplication", *Int'l Conf. on Par. Proc. (ICPP'08)*, pp. 503-510, September 2008.

# Developing an OO Model for Generalized Matrix Multiplication: Preliminary Considerations

Maria Ganzha, Marcin Paprzycki  
 Systems Research Laboratory  
 Polish Academy of Sciences  
 01-447 Warszawa, Poland  
 Email: Maria.Ganzha@ibspan.waw.pl

Stanislav G. Sedukhin  
 Distributed Parallel Processing Laboratory  
 The University of Aizu  
 Aizuwakamatsu City, Fukushima 965-8580, Japan  
 Email: sedukhin@u-aizu.ac.jp

**Abstract**—Recent changes in computational sciences force reevaluation of the role of dense matrix multiplication. Among others, this resulted in a proposal to consider generalized matrix multiplication, based on the theory of algebraic semirings. The aim of this note is to outline an initial object oriented model of the generalized matrix-multiply-add operation.

## I. INTRODUCTION

THE DENSE matrix multiplication appears in many computational problems. Its arithmetic complexity ( $O(n^3)$ ) and inherent data dependencies pose a challenge for reducing its *run-time* complexity. There exist three basic approaches to decrease the execution time of dense matrix multiplication.

(1) Reducing the number of (time consuming) scalar multiplications, while increasing the number of (much faster) additions; see, discussion and references in [1]. These approaches had very good theoretical arithmetical complexity, and worked well when implemented on computers with a single processor and main memory. However, due to complex data access patterns they became difficult to efficiently implement on computers with hierarchical memory. Furthermore, recursive matrix multiplication requires extra memory; e.g. Cray's implementation of Strassen's algorithm required extra space of  $2.34 * N^2$  for matrices of size  $N \times N$ .

(2) Parallelization of matrix multiplication, which is based on one of four classes of schedules ([2]): (i) Broadcast-Compute-Shift; (ii) All-Shift-Compute (or Systolic); (iii) Broadcast-Compute-Roll; and (iv) Compute-Roll-All (or Orbital). The latter is characterized by regularity and locality of data movement, maximal data reuse without data replication, recurrent ability to involve into computing all matrix data at once (retina I/O), etc.

(3) Combination of these approaches, where irregularity of data movement is exaggerated through the complexity of the underlying hardware. Interestingly, work on recursive (and recursive-parallel) matrix multiplication seems to be subsiding, as the last known to us paper comes from 2006 [3].

Note that, in sparse matrix algebra the main goal was to save memory; achieved via indexing structures storing information about non-zero elements (resulting in complex data access patterns; [4]). However, nowadays the basic element becomes a dense block while regularity of data access compensates for the multiplications by zero [5].

Generalized matrix multiplication appears in the Algebraic Path Problem (APP), examples of which include: finding the most reliable path, finding the critical path, finding the maximum capacity path, etc. Here, a generalized is based on the algebraic theory of semirings (see [6] and references collected there). Note that, standard linear algebra (with its matrix multiplication) is an example of an algebraic (matrix) semiring. Application of algebraic semirings to “unify through generalization” a large class of computational problems, should be viewed in the context of recent changes in CPU architectures: (1) popularity of fused multiply-add (FMA) units, which take three scalar operands and produce a result of  $c \leftarrow a \cdot b + c$  in a single clock cycle, (2) increase of the number of cores-per-processor (e.g. recent announcement of 10-core processors from Intel), and (3) success of GPU processors (e.g. the Fermi architecture from Nvidia and the Cypress architecture from AMD) that combine multiple FMA units (e.g. the Fermi architecture delivers in a single cycle 512 single-precision, or 256 double-precision FMA results).

Finally, the work reported in [4] illustrates a important aspect of highly optimized codes that deal with complex matrix structures. While the code generator, is approximately 6,000 lines long, the generated code is more than 100,000 lines. Therefore, when thinking about fast matrix multiplication, one needs to consider also the programming cost required to develop and later update codes based on complex data structures and movements.

## II. ALGEBRAIC SEMIRINGS IN SCIENTIFIC CALCULATIONS

Since 1970's, a number of problems have been combined into the *Algebraic Path Problem* (APP; see [7]). The APP includes problems from linear algebra, graph theory, optimization, etc. while their solution draws from theory of semirings.

A closed semiring  $(S, \oplus, \otimes, *, \bar{0}, \bar{1})$  is an algebraic structure defined for a set  $S$ , with two binary operations: addition  $\oplus : S \times S \rightarrow S$  and multiplication  $\otimes : S \times S \rightarrow S$ , a unary operation called *closure*  $*$  :  $S \rightarrow S$ , and two constants  $\bar{0}$  and  $\bar{1}$  in  $S$ . Here, we are particularly interested in the case when the elements of the set  $S$  are matrices. Thus, following [7], we introduce a matrix semiring  $(S^{n \times n}, \oplus, \otimes, \star, \bar{0}, \bar{1})$  as a set of  $n \times n$  matrices  $S^{n \times n}$  over a closed scalar semiring  $(S, \oplus, \otimes, *, \bar{0}, \bar{1})$  with two binary operations, matrix addition

## 1) Matrix Inversion Problem:

$$(\alpha) a(i, j) = a(i, j) + \sum_{k=0}^{N-1} a(i, k) \times a(k, j);$$

$$(\omega) c = a \times b + c;$$

## 2) All-Pairs Shortest Paths Problem:

$$(\alpha) a(i, j) = \min\{a(i, j), \min_{k=0}^{N-1}[a(i, k) + a(k, j)]\};$$

$$(\omega) c = \min(c, a + b);$$

## 3) Minimum Spanning Tree Problem:

$$(\alpha) a(i, j) = \min\{a(i, j), \min_{k=0}^{N-1}[\max(a(i, k), a(k, j))]\};$$

$$(\omega) c = \min[c, \max(a, b)].$$

Fig. 1. Matrix and scalar semiring operations for sample APP problems

$\oplus : S^{n \times n} \times S^{n \times n} \rightarrow S^{n \times n}$  and matrix multiplication  
 $\otimes : S^{n \times n} \times S^{n \times n} \rightarrow S^{n \times n}$ , a unary operation called *closure of a matrix*  $\star : S^{n \times n} \rightarrow S^{n \times n}$ , the zero  $n \times n$  matrix  $\bar{0}$  whose all elements equal to  $\bar{0}$ , and the  $n \times n$  identity matrix  $\bar{I}$  whose all main diagonal elements equal to  $\bar{1}$  and  $\bar{0}$  otherwise. Here, matrix addition and multiplication are defined as usually in linear algebra. Note that, special cases matrices that are non-square, symmetric, structural, etc., while not usually considered in the theory of semirings, are also handled by the above provided definition.

The existing blocked algorithms for solving the APP, are rich in generalized block MMA operations, which are their most compute intensive parts [8]. In the generalized block MMA, addition and multiplication originate from any semiring (possibly different than the standard numerical algebra). In Figure 1 we present the relation between the scalar multiply-add operation ( $\omega$ ), and the corresponding MMA kernel ( $\alpha$ ), for different semirings for three sample APP applications; here,  $N$  is the size of the matrix block (see, also [8]).

Overall, the generalized MMA is one of key operations for the APP problems, including MMA-based numerical linear algebra algorithms, which include block-formulations of linear algebraic problems (as conceptualized in the level 3 BLAS operations [9], and applied in the LAPACK library [10]).

## III. MATRIX OPERATIONS AND COMPUTER HARDWARE

In 1970's it was realized that many matrix algorithms consist of similar building blocks (e.g. a vector update, or a dot-product). As a result, Cray computers provided optimized vector operations:  $y \leftarrow y + \alpha x$ , while IBM supercomputers featured optimized dot products. This resulted also in creation of libraries of routines for scientific computing (e.g. the *scilib* library on the Cray's, and the *ESSL* library on the IBM's). Separately, the first hardware implementation of the fused multiply-add (FMA) operation was delivered in the IBM RS/6000 workstations [11]. Following this path, most current processors from IBM, Intel, AMD, NVidia, and others, include scalar floating-point FMA [12]. Observe that, the basic arithmetic operations: add and multiply, are performed by the FMA unit by making  $a = 1.0$  (or  $b = 1.0$ ) for addition, or  $c = 0.0$  for multiplication. Therefore, the two fundamental constants, 0.0 and 1.0, have to be available in the hardware. Therefore, processors that perform the FMA implement in hardware the scalar (+,  $\times$ ) semiring.

Obviously, the FMAs speed-up ( $\sim 2\times$ ) the solution of scientific, engineering, and multimedia algorithms based on the linear algebra (matrix) transforms [13]. On the other hand, lack of hardware support penalizes APP solvers from other semirings. For instance, in [8] authors have showed that the "penalty" for lack of a generalized FMA unit in the Cell/B.E. processor may be up to 400%. Obviously, this can be seen from the "positive side." Having hardware unit fully supporting operations listed in Figure 1 would speed up solution of APP problems by up to 4 times. Interestingly, we have just found that the AMD Cypress GPU processor supports the (min, max)-operation through a single call with 2 clock cycles per result. Therefore, the Minimum Spanning Tree problem (see, Figure 1) could be solved substantially more efficiently than previously realized. Furthermore, this could mean that the AMD hardware has build-in elements corresponding to  $-\infty$  and  $\infty$ . This, in turn, could constitute an important step towards hardware support of generalized scalar FMA operations needed to realize many APP kernels(see, also [8]).

In the 1990's three designs for parallel computers have been tried: (1) array processors, (2) shared memory parallel computers, and (3) distributed memory parallel computers. After a number of "trials-and-errors," combined with progress in miniaturization, we witness the first two approaches joined within a processor and such processors combined into large machines. Specifically, while vendors like IBM, Intel and AMD develop multi-core processors with slowly increasing number of cores, the Nvidia and the AMD develop array processors on the chip. However, all these designs lead to processors consisting of thousands of FMA units.

## IV. PROPOSED GENERALIZED MULTPLY-AND-ADD

Based on these considerations, in [14] we have defined a generic generalized matrix-multiply-add operation (MMA),

$$C \leftarrow \text{MMA}[\otimes, \oplus](A, B, C) : C \leftarrow A \otimes B \oplus C,$$

where the  $[\otimes, \oplus]$  operations originate from different matrix semirings. Note that, like in the scalar FMAs, generalized matrix *addition* and *multiplication*, can be implemented by making an  $n \times n$  matrix  $A$  (or  $B$ ) =  $\bar{0}$  for addition, or a matrix  $C = \bar{I}$  for multiplication (see, Section II).

Obviously, the generalized MMA resembles the `_GEMM` operation from the level 3 BLAS. Therefore, in [14] it was shown that, except for the triangular solve, BLAS 3 operations can be expressed in terms of the generalized MMA (in the linear algebra semiring). Note also that the proposed approach supports the idea that, in future computer hardware, data manipulation will be easiest to complete through matrix multiplication. Specifically, since the MMA represents a linear transformation of a vector space, it can be used for reordering of matrix rows/columns, matrix rotation, transposition, etc. Furthermore, operations like global reduction and broadcast can be easily obtained via matrix multiplication (see, [2]).

In summary, the proposal put forward in [14] covers three important aspects: a) it subsumes the level 3 BLAS, b) it generalizes the MMA, to encompass most of APP kernels,

and c) allows for new way of writing APP kernels, optimized for computers consisting of large number of processors with thousands of generalized FMA cores each, and simplified to support code maintainability.

## V. STATE-OF-THE-ART IN OBJECT ORIENTED BLAS

While our work extends and generalizes dense matrix multiplication, its object oriented (OO) realization should be conceptually related to OO numerical linear algebra, including the OO BLAS. Here, we briefly introduce selected OO realizations of numerical linear algebra in general, and BLAS in particular: MTL, uBLAS, TNT, Armadillo, and Eigen.

The uBLAS project ([15]) was focused on design of a C++ library that provided BLAS functionality. An additional goal of uBLAS was to evaluate if the abstraction penalty, resulting from object orientation, is acceptable. The uBLAS was guided by: (i) Blitz++ [16], POOMA [17], and MTL [18]. Data found on the Web indicates that the project was completed around 2002 and later subsumed into the BOOST [19] library.

The TNT project ([20]) is a collection of interfaces and C++ reference implementations that includes, among others, operations on multidimensional arrays and sparse matrices. The library, while not updated since 2004, can still be downloaded from the project Web site.

The MTL project remained active until around 2008-09, when the last paper/presentation concerning the MTL 4 was reported. Interestingly, this project provided not only an open source library, but also a paid one (more optimized).

There are two projects that are vigorously pursued today: Armadillo ([21]; last release on June 29, 2011) and Eigen ([22]; last release on May 30, 2011). Both support an extensive set of matrix operations. While Eigen seems to be focused on vector processor optimizations, Armadillo “links” with vendor optimized matrix libraries: MKL and ACML.

## VI. INITIAL OBJECT ORIENTED MODEL

Following the ideas described above, in Figure 2 we depict our proposed OO model for the generalized matrix multiplication. Here, we see the interface to be made available to the user. It will allow to instantiate needed matrices and develop code with generalized matrix operations: matrix addition, matrix multiplication, and the MMA. We also define the abstract class *scalar\_Semiring* needed to define operations of the *scalar* semiring (operations on elements of matrices).

The main class is the *Matrix* class. It describes how the matrix operations defined in the interface class are realized. Among others, it contains the MMA function, which is used to actually realize the matrix operations. The implementation of the MMA function is to be provided by the user, or by the hardware vendor (in a way similar to the vendor-optimized implementation of BLAS kernels).

Finally, we depict a sample specialization of the *scalar\_Semiring* class (for the Shortest Path Problem; see, Figure 1), with elements from the  $R_+$ ,  $\bar{0} = 1$ ,  $\bar{1} = \infty$ ,  $\oplus = \min$ , while  $\otimes = +$ . Based on this specific scalar semiring (and, possibly, a generalized FMA unit), the appropriate MMA operation is implemented in the *Matrix* class.

## VII. SAMPLE REALIZATION

Let us now recall that one of our goals is to simplify code development (in a way that the BLAS simplified it 30+ years ago). Therefore, code should be as simple as possible, with details of the implementation hidden. With this in mind, let us start from defining the interfaces. The *Matrix\_interface* defines operations available to the user to write her codes (for simplicity, our description is limited to square matrices).

```
interface Matrix_interface {
    Init(n); //initialisation of square matrix nxn
    Matrix matrix0(n)
        { /*generalized 0 matrix*/ }
    Matrix matrix11(n)
        { /*generalized identity matrix*/ }
    Matrix operator +
        { /*generalized matrix addition A+B*/ }
    Matrix operator *
        { /*generalized matrix product A*B*/ }
    Matrix Column_Permutation (A,i,j)
        { /*generalized permutation of column i
            and j in matrix A*/ }
    Matrix Row_Permutation (A,i,j)
        { /*generalized permutation of column i and
            j in matrix A*/ }
};
```

Here, user can create matrix objects, zero and identity matrices (for a given semiring). Furthermore, we define generalized matrix operators and two permutation matrices. This interface can be extended to include other user-defined operations. Next, we define the *scalar\_Semiring* class.

```
abstract class scalar_Semiring {
public:
    //T — type of element;
    zero, one:T;
    +: c=a+b;
    *: c=a*b;
};
```

It specifies generalized operations *addition* and *multiplication* as well as elements  $\bar{0}$  and  $\bar{1}$ . It has to be provided by the user, to “select” the semiring.

With the two interfaces in place, we can define the core class *Matrix*, which is **not** made visible to the user (is internal to the realization of the generalized MMA). It inherits the *scalar\_Semiring* interface, and implements the *Matrix\_interface* interface.

```
class Matrix inherit scalar_Semiring
    implement Matrix_interface {
T: type of element; /*double, single, ... */
n: int;
// Methods
Init(n); //initialisation of square matrix nxn
Matrix matrix_0(n) { /*generalized 0 matrix*/ }
Matrix matrix_1_1(n) { /*generalized identity matrix*/ }
Matrix matrix_IP(i,j,n)
    { /*generalized identity matrix with interchanged
        columns i and j*/ }
Matrix A+B { return MMA(A,M1: matrix_1_1(n),B) }
Matrix A*B { return MMA(A,B,M0: matrix_0(n)) }
Matrix Column_Permutation (A,i,j){
    P=matrix_IP(i,j,n);
    O=matrix_0(n);
    return MMA(P,A,O) }
Matrix Row_Permutation (A,i,j){
    P=matrix_IP(i,j,n);
    O=matrix_0(n);
    return MMA(A,P,O) }
...
private MMA(A,B,C: Matrix(n)) {
```



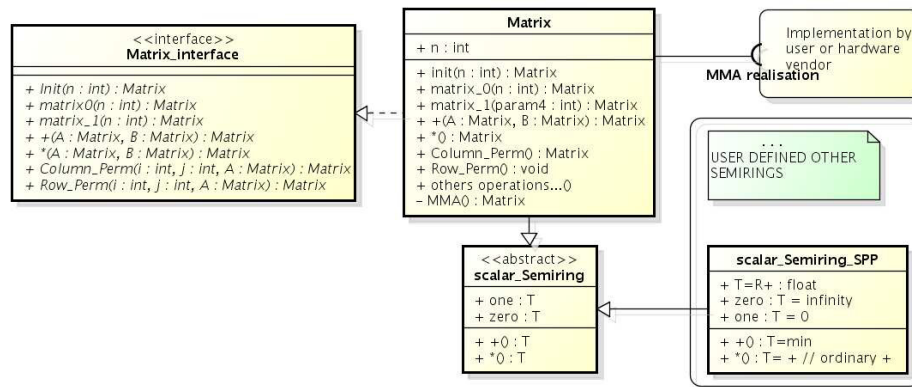


Fig. 2. General schema of the proposed object oriented model of generalized matrix multiplication

```

return vendor/user specific realization of
MMA = C+A*B where+ and * are from
class scalar_Semiring }
}

```

The key part of this class is the private function MMA. This is the actual ( user/vendor specific) realization of the MMA operation. As a result, user can perform operations:  $A \oplus B$ ,  $A \otimes B$ , or  $C \oplus A \otimes B$ , written in the code as  $A + B$ ,  $A * B$ , or  $C + A * B$  without any knowledge of their actual hardware/software realization. Observe also, that matrix column permutation, and row permutation have been defined in operations *Column\_Permutation* and *Row\_Permutation*, which are implemented through a call to the MMA function with appropriate matrices (see, also [14]).

Finally, in the next snippet we show the class *scalar\_Semiring* rewritten for the Shortest Path Problem (see, Figure1). After defining this class the user can simply apply the generalized MMA operation within the solver.

```

/*T = R+ PLUS infinity , + = min , * = + ,
ZERO = infinity , ONE = 0; */

```

```

class scalar_Semiring {
    zero="infinity";
    one=0;
    a+b = min(a,b);
    a*b = a+b;
}

```

## VIII. CONCLUDING REMARKS

The aim of this paper was to propose the object model for the generalized matrix multiplication. The proposed approach is not language specific and presented at a very high level. Next, we will proceed with its more detailed realization in most important languages used in scientific computing.

## REFERENCES

- [1] S. Robinson, "Towards an optimal algorithm for matrix multiplication." *SIAM News*, vol. 38, no. 9, 2005.
- [2] S. G. Sedukhin, A. S. Zekri, and T. Myiazaki, "Orbital algorithms and unified array processor for computing 2D separable transforms," *Parallel Processing Workshops, International Conference on*, vol. 0, pp. 127–134, 2010.
- [3] F. Song, S. Moore, and J. Dongarra, "Experiments with Strassen's Algorithm: from Sequential to Parallel," in *International Conference on Parallel and Distributed Computing and Systems (PDCS06)*. ACTA Press, November, 2006.
- [4] M. Martone, S. Filippone, P. Gepner, M. Paprzycki, and S. Tucci, "Use of hybrid recursive csr/coo data structures in sparse matrices-vector multiplication," in *IMCSIT*, 2010, pp. 327–335.
- [5] J. L. Gustafson, "Algorithm leadership," *HPCwire*, vol. Tabor Communications, April 06, 2007.
- [6] S. G. Sedukhin, T. Miyazaki, and K. Kuroda, "Orbital systolic algorithms and array processors for solution of the algebraic path problem," *IEICE Transactions on Information and Systems*, vol. E93.D, no. 3, pp. 534–541, 2010.
- [7] D. J. Lehmann, "Algebraic structures for transitive closure," *Theor. Comput. Sci.*, vol. 4, no. 1, pp. 59–76, 1977.
- [8] S. G. Sedukhin and T. Miyazaki, "Rapid\*Closure: Algebraic extensions of a scalar multiply-add operation," in *CATA*, T. Philips, Ed. ISCA, 2010, pp. 19–24.
- [9] J. J. Dongarra, J. D. Croz, I. Duff, and S. Hammarling, "A set of level 3 basic linear algebra subprograms," *ACM Trans. Math. Software*, vol. 16, pp. 1–17, 1990.
- [10] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide*, 3rd ed. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1999.
- [11] R. K. Montoye, E. Hokenek, and S. L. Runyon, "Design of the IBM RISC System/6000 floating-point execution unit," *IBM J. Res. Dev.*, vol. 34, no. 1, pp. 59–70, 1990.
- [12] S. Mueller, C. Jacobi, H.-J. Oh, K. Tran, S. Cottier, B. Michael, H. Nishikawa, Y. Totsuka, T. Namatame, N. Yano, T. Machida, and S. Dhong, "The vector floating-point unit in a synergistic processor element of a CELL processor," *Computer Arithmetic, 2005. ARITH-17 2005. 17th IEEE Symposium on*, pp. 59–67, June 2005.
- [13] F. G. Gustavson, J. E. Moreira, and R. F. Enenkel, "The fused multiply-add instruction leads to algorithms for extended-precision floating point: applications to Java and high-performance computing," in *CASCON '99: Proceedings of the 1999 conference of the Centre for Advanced Studies on Collaborative Research*. IBM Press, 1999, p. 4.
- [14] S. Sedukhin and M. Paprzycki, "Generalizing matrix multiplication for efficient computations on modern computers," to appear.
- [15] "uBLAS—Basic Linear Algebra Library," [http://www.boost.org/doc/libs/1\\_46\\_1/libs/numeric/ublas/doc/index.htm](http://www.boost.org/doc/libs/1_46_1/libs/numeric/ublas/doc/index.htm).
- [16] "BLITZ++," <http://www.oonumerics.org/blitz/>.
- [17] "Parallel Object-Oriented Methods and Applications," <http://acts.nersc.gov/pooma/>.
- [18] "Matrix Template Library," <http://www.simunova.com/en/node/24>.
- [19] "Boost C++ library," <http://www.boost.org/>.
- [20] "Template Numerical Toolkit," <http://math.nist.gov/tnt/>.
- [21] "Armadillo: C++ linear algebra library," <http://arma.sourceforge.net/>.
- [22] "Eigen," [http://eigen.tuxfamily.org/index.php?title=Main\\_Page](http://eigen.tuxfamily.org/index.php?title=Main_Page).

# Parallel alternating directions algorithm for 3D Stokes equation

Ivan Lirkov  
Institute of Information  
and Communication Technologies  
Bulgarian Academy of Sciences  
Acad G. Bonchev, bl. 25A  
1113 Sofia, Bulgaria  
ivan@parallel.bas.bg  
<http://parallel.bas.bg/~ivan/>

Marcin Paprzycki Maria Ganzha  
Systems Research Institute  
Polish Academy of Sciences  
ul. Newelska 6  
01-447 Warsaw, Poland  
paprzyck@ibspan.waw.pl  
maria.ganzha@ibspan.waw.pl  
<http://www.ibspan.waw.pl/~paprzyck/>  
<http://inf.ug.edu.pl/~mganzha/>

Paweł Gepner  
Intel Corporation  
Pipers Way  
Swindon Wiltshire SN3 1RJ  
United Kingdom  
pawel.gepner@intel.com

**Abstract**—We consider the 3D time dependent Stokes equation on a finite time interval and on a uniform rectangular mesh, written in terms of velocity and pressure.

For this problem, a parallel algorithm, based on a recently proposed direction splitting approach, is applied. Here, the pressure equation is derived from a perturbed form of the continuity equation, where the incompressibility constraint is penalized in a negative norm induced by the direction splitting. The scheme used in the algorithm is composed of: (a) pressure prediction, (b) velocity update, (c) penalty step, and (d) pressure correction. In order to achieve good parallel performance, the solution of the Poisson problem for the pressure correction is replaced by a solution to a sequence of one-dimensional second order elliptic boundary value problems (in each spatial direction). The efficiency and scalability of the proposed approach are tested on two distinct parallel computers and the experimental results are analyzed.

## I. INTRODUCTION

**T**HE OBJECTIVE of this note is to analyze the parallel performance of a novel fractional time stepping technique, based on a direction splitting strategy, developed to solve the incompressible Navier-Stokes equations.

Computational fluid dynamics (CFD) has undergone tremendous development as a discipline. This has been made possible by progresses in many fronts, including numerical algorithms for the Navier-Stokes equations, grid generation and adaptation, turbulence modeling, flow visualization, as well as the dramatic increase of computer CPU and network speeds.

Finding an approximate solution of the Navier-Stokes equations can be done by a large range of numerical methods. Among these, finite element methods are used mostly by mathematicians, while spectral methods and finite volume methods are favored by engineers and physicists. One reason for this difference in computational practices is that an advantage of finite volume methods over finite element ones lies primarily in ease of their physical interpretation and in simpler implementation. Currently, nearly all production-class flow solvers are based on second-order numerical methods, either finite volume [9], [10], [23], [25], finite difference [33], or finite element [4], [11], [12], [18], [20]. They are capable of delivering,

within a few hours, design-quality Reynolds Averaged Navier-Stokes results with several million cells (degrees of freedom) on various Beowulf-style cluster computers.

The efficient solution of the discretized Navier-Stokes equations necessitates rapidly convergent iterative methods. The two main approaches available here are: (i) preconditioned Krylov subspace methods [30], and (ii) multigrid methods [35], [36], [38]. These two approaches can be combined by using one or more multigrid cycles as preconditioners for the Krylov-type methods. Most of recent papers on the iterative solution of the discretized Navier-Stokes equations are devoted to block preconditioners [3], [7], [21], [31]. Here, more recent contributions include the preconditioning based on the augmented Lagrangian approach [3], and the least-squares commutator preconditioner generalized to the stabilized finite element discretizations of the Oseen problem [8]. Other relevant work includes the development of ILU-type preconditioners for saddle-point problems [28], and SIMPLE-type block preconditioners [29].

Alternatively, one could start with the “physics-based” iterative solution methods for the Navier-Stokes equations [26], [27] and develop preconditioners based on these techniques as described in [22]. In this case, the system is transformed by the factorization into component systems that are essentially convection-diffusion and Poisson type operators. The result is a system to which multi-level methods and algebraic multi-level methods (AMG) can be successfully applied. In recent years there has been tremendous interest in the mathematical development and practical implementation of discontinuous Galerkin finite element methods (DGFEMs) for the discretization of compressible fluid flow problems, [2], [6], [19]. The key advantages of these schemes are that the DGFEMs provide robust and high-order accurate approximations, particularly in transport-dominated regimes, and that they are locally conservative. Moreover, they provide considerable flexibility in the choice of the mesh design. Indeed, the DGFEMs can easily handle non-matching grids and non-uniform, even anisotropic, polynomial approximation degrees.

Projection schemes were first introduced in [5], [34] and they have been used in CFD for about forty years. During these years, such techniques went through some evolution, but the main paradigm, consisting of decomposing vector fields into a divergence-free part and a gradient, has been preserved (see [14] for a review of projection methods). In terms of computational efficiency, projection algorithms are far superior to the methods that solve the coupled velocity-pressure system. This feature makes them the most popular techniques in the CFD community for solving the unsteady Navier-Stokes equations. The computational complexity of each time step of the projection methods is that of solving one vector-valued advection-diffusion equation, plus one scalar-valued Poisson equation with Neumann boundary conditions. Note that, for large scale problems, and large Reynolds numbers, the cost of solving the Poisson equation becomes dominant.

The alternating directions algorithm, initially proposed in [13], reduces the computational complexity of the action of the incompressibility constraint. The key idea is to modify the projection paradigm, in which the vector fields are decomposed into a divergence-free part plus a gradient part. Departure from the standard projection methods has been proved to be very efficient for solving variable density flows (see, for instance, [15], [16]). In the new method, the pressure equation is derived from a perturbed form of the continuity equation, in which the incompressibility constraint is penalized in a negative norm, induced by the direction splitting. The standard Poisson problem for the pressure correction is replaced by series of one-dimensional second-order boundary value problems. This technique was proved to be stable and convergent; for details see [13]. Furthermore, a very sketchy assessment indicated that it has good potential for parallelization.

In this note we follow the proposal introduced in [13] and study its performance characteristics on two different computers. One of them is an Intel-Xeon-processor-based cluster, while the other is an IBM Blue Gene supercomputer. Experimental results reported in Section IV confirm the preliminary assessment provided in [13].

## II. STOKES EQUATION

Let us first define the problem to be solved. We consider the time-dependent Navier-Stokes equations on a finite time interval  $[0, T]$ , and in a rectangular domain  $\Omega$ . Since the non-linear term in the Navier-Stokes equations does not interfere with the incompressibility constraint, we focus our attention on the time-dependent Stokes equations written in terms of velocity  $\mathbf{u}$  and pressure  $p$ :

$$\begin{cases} \mathbf{u}_t - \nu \Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \times (0, T) \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \times (0, T) \\ \mathbf{u}|_{\partial\Omega} = 0, \quad \partial_n p|_{\partial\Omega} = 0 & \text{in } (0, T) \\ \mathbf{u}|_{t=0} = \mathbf{u}_0, \quad p|_{t=0} = p_0 & \text{in } \Omega \end{cases}, \quad (1)$$

where  $\mathbf{f}$  is a smooth source term,  $\nu$  is the kinematic viscosity, and  $\mathbf{u}_0$  is a solenoidal initial velocity field with a zero normal trace. In our work, we consider homogeneous Dirichlet boundary conditions on the velocity.

To solve thus described problem, we discretize the time interval  $[0, T]$  using a uniform mesh. Furthermore, let  $\tau$  be the time step used in the algorithm.

## III. PARALLEL ALTERNATING DIRECTIONS ALGORITHM

For thus introduced problem, let us describe the proposed parallel solution method. In [13], Guermond and Mineev introduced a novel fractional time stepping technique for solving the incompressible Navier-Stokes equations. Their approach is based on a direction splitting strategy. They used a singular perturbation of the Stokes equation with a perturbation parameter  $\tau$ . The standard Poisson problem for the pressure correction was replaced by series of one-dimensional second-order boundary value problems. The focus of their work was to show numerical properties of the proposed approach (e.g. its stability and convergence). However, they also very briefly indicated its potential for efficient parallelization. Therefore, to describe the parallel solution approach, let us start from the overview of the alternating directions method.

### A. Formulation of the Scheme

The scheme used in the Guermond-Mineev algorithm is composed of the following parts: (i) pressure prediction, (ii) velocity update, (iii) penalty step, and (iv) pressure correction. Let us now describe an algorithm that uses the direction splitting operator

$$A := \left(1 - \frac{\partial^2}{\partial x^2}\right) \left(1 - \frac{\partial^2}{\partial y^2}\right) \left(1 - \frac{\partial^2}{\partial z^2}\right).$$

- *Pressure predictor*

Denoting by  $p_0$  the pressure field at  $t = 0$ , the algorithm is initialized by setting  $p^{-\frac{1}{2}} = p^{-\frac{3}{2}} = p_0$ . Next, for all  $n \geq 0$ , a pressure predictor is computed as follows

$$p^{*,n+\frac{1}{2}} = 2p^{n-\frac{1}{2}} - p^{n-\frac{3}{2}}. \quad (2)$$

- *Velocity update*

In the *velocity update* step, the velocity field is initialized by setting  $\mathbf{u}^0 = \mathbf{u}_0$ , and for all  $n \geq 0$  the velocity update is computed by solving the following series of one-dimensional problems

$$\frac{\xi^{n+1} - \mathbf{u}^n}{\tau} - \nu \Delta \mathbf{u}^n + \nabla p^{*,n+\frac{1}{2}} = \mathbf{f}|_{t=(n+\frac{1}{2})\tau},$$

$$\frac{\eta^{n+1} - \xi^{n+1}}{\tau} - \frac{\nu}{2} \frac{\partial^2 (\eta^{n+1} - \mathbf{u}^n)}{\partial x^2} = 0, \quad (3)$$

$$\frac{\zeta^{n+1} - \eta^{n+1}}{\tau} - \frac{\nu}{2} \frac{\partial^2 (\zeta^{n+1} - \mathbf{u}^n)}{\partial y^2} = 0, \quad (4)$$

$$\frac{\mathbf{u}^{n+1} - \zeta^{n+1}}{\tau} - \frac{\nu}{2} \frac{\partial^2 (\mathbf{u}^{n+1} - \mathbf{u}^n)}{\partial z^2} = 0, \quad (5)$$

where  $\xi^{n+1}|_{\partial\Omega} = \eta^{n+1}|_{\partial\Omega} = \zeta^{n+1}|_{\partial\Omega} = \mathbf{u}^{n+1}|_{\partial\Omega} = 0$ .

- *Penalty step*

in the *Penalty step*, the intermediate parameter  $\phi$  is approximated by solving  $A\phi = -\frac{1}{\tau}\nabla \cdot \mathbf{u}^{n+1}$ . Owing to the definition of the direction splitting operator  $A$ , this is



done by solving the following series of one-dimensional problems:

$$\begin{aligned} \theta - \theta_{xx} &= -\frac{1}{\tau} \nabla \cdot \mathbf{u}^{n+1}, & \theta_x|_{\partial\Omega} &= 0, \\ \psi - \psi_{yy} &= \theta, & \psi_y|_{\partial\Omega} &= 0, \\ \phi - \phi_{zz} &= \psi, & \phi_z|_{\partial\Omega} &= 0, \end{aligned} \quad (6)$$

- *Pressure update*

The last sub-step of the algorithm consists of *updating the pressure*:

$$p^{n+\frac{1}{2}} = p^{n-\frac{1}{2}} + \phi - \chi \nu \nabla \cdot \frac{\mathbf{u}^{n+1} + \mathbf{u}^n}{2} \quad (7)$$

The algorithm is in a standard incremental form when the parameter  $\chi = 0$ ; while the algorithm is in a rotational incremental form when  $\chi \in (0, \frac{1}{2}]$ .

### B. Parallel Algorithm

The proposed algorithm uses a rectangular uniform mesh combined with a central difference scheme for the second derivatives for solving equations (3–5) and (6). Thus the algorithm requires only the solution of tridiagonal linear systems.

The parallelization is based on a decomposition of the domain into rectangular sub-domains. Let us associate with each such sub-domain a set of integer coordinates  $(i_x, i_y, i_z)$ , and identify it with a given processor. The linear systems, generated by the one-dimensional problems that need to be solved in each direction, are divided into systems for each set of unknowns corresponding to the internal nodes, for each block that can be solved independently by a direct method. The corresponding Schur complement for the interface unknowns between the blocks that have an equal coordinate  $i_x, i_y$ , or  $i_z$  is also tridiagonal and can therefore be easily directly inverted. The overall algorithm requires only exchange of the interface data (between sub-domains), which allows for a very efficient parallelization with an expected efficiency comparable to that of explicit schemes.

## IV. EXPERIMENTAL RESULTS

As stated above, the main goal of our current work is to evaluate the performance of the proposed approach; to experimentally confirm the initial positive assessment found in [13]. Therefore, to assess the performance of the proposed approach, we have solved the problem (1) in  $\Omega = (0, 1)^3$ , for  $t \in [0, 2]$ , with Dirichlet boundary conditions. The discretization in time was done with the time step  $10^{-2}$ , while the parameter in the pressure update sub-step was  $\chi = \frac{1}{2}$ , and the kinematic viscosity was  $\nu = 10^{-3}$ . The discretization in space used mesh sizes  $h_x = \frac{1}{n_x-1}$ ,  $h_y = \frac{1}{n_y-1}$ , and  $h_z = \frac{1}{n_z-1}$ . Thus, the equation (3) resulted in linear systems of size  $n_x$ , while equation (4) resulted in linear systems of size  $n_y$ , and equation (5) in linear systems of size  $n_z$ . The total number of unknowns in the discrete problem was  $800 n_x n_y n_z$ .

To solve the problem, a portable parallel code was designed and implemented in C, while the parallelization has been facilitated using the MPI library [32], [37]. In the code, we used the LAPACK subroutines DPTTRF and DPTTS2 (see [1]) for solving tridiagonal systems of equations, resulting

from equations (3), (4), (5), and (6), for the unknowns corresponding to the internal nodes of each sub-domain. The same subroutines were used to solve the tridiagonal systems with the Schur complement.

The parallel code has been tested on an Intel processor-based cluster computer system (Sooner), located in the Oklahoma Supercomputing Center (OSCER), and the IBM Blue Gene/P machine at the Bulgarian Supercomputing Center. In our experiments, times have been collected using the MPI provided timer and we report the best results from multiple runs. In what follows, we report the elapsed time  $T_c$  in seconds using  $c$  cores, the parallel speed-up  $S_c = T_1/T_c$ , and the parallel efficiency  $E_c = S_c/c$ .

Table I represents the results collected on the Sooner, which is a Dell Intel Xeon E5405 (“Harpertown”) quad core-based Linux cluster. It has 486 Dell PowerEdge 1950 III nodes, and two quad core processors per node. Each processor runs at 2 GHz. Processors within each node share 16 GB of memory, while nodes are interconnected through a high-speed InfiniBand network (for additional details concerning the machine, see <http://www.oscer.ou.edu/resources.php>). We have used an Intel C compiler, and compiled the code with the following options: “-O3 -march=core2 -mtune=core2.” Note that, even though such approach would be possible, we have not attempted at a two-level parallelization, where the OpenMP would be used within multi-core processors (or possibly within each computational node, where 8 cores reside), while the MPI would be used for the “between-nodes” parallelization. We have decided that such approach would not be warranted for the initial performance evaluation. However, due to the promising nature of our results, we plan to pursue such two-level parallelization in the near future, especially in view of increasing number of computational cores that are to appear within both Intel and AMD families of processors. Such approach may also be applicable for multi-core GPU-type processors (e.g. based on the Fermi or the Cypress architectures). We plan to explore usability of GPU processors, for the problem at hand, in the future.

It has to be noted that our code needs 11 GB of memory for solving the problem for  $n_x = n_y = n_z = 400$ . Since the memory on one node of the Sooner is 16 GB, it is the largest size of the discrete problem that can be solved on a single node. Therefore, results presented in Table I represent the largest problems we were able to solve.

The sets of results in each “column-box” of Table I were obtained for an equal number of unknowns per core. For large discrete problems, the execution time in one and the same “column-box” is much larger on two processors (8 cores) than on one processor, but on more processors the time is approximately constant. The obtained execution times confirm that the communication time between processors is larger than the communication time between cores within one processor. Also, the execution time for solving one and the same discrete problem decreases with increasing the number of cores, which shows that the communication in our parallel algorithm is mainly local.

TABLE I  
EXECUTION TIME ON SOONER.

$c$	$n_x$	$n_y$	$n_z$	$T_c$	$n_x$	$n_y$	$n_z$	$T_c$	$n_x$	$n_y$	$n_z$	$T_c$	$n_x$	$n_y$	$n_z$	$T_c$
1	50	50	50	18.96	50	50	100	41.46	50	100	100	101.01	100	100	100	205.96
2	50	50	100	20.11	50	100	100	49.09	100	100	100	107.91	100	100	200	236.44
4	50	100	100	22.16	100	100	100	50.53	100	100	200	145.75	100	200	200	344.56
8	100	100	100	37.16	100	100	200	113.61	100	200	200	280.45	200	200	200	571.77
16	100	100	200	48.22	100	200	200	129.06	200	200	200	283.17	200	200	400	625.50
32	100	200	200	48.80	200	200	200	116.61	200	200	400	283.29	200	400	400	629.75
64	200	200	200	39.95	200	200	400	117.94	200	400	400	286.85	400	400	400	581.27
128	200	200	400	51.20	200	400	400	134.07	400	400	400	291.26	400	400	800	644.10
256	200	400	400	55.14	400	400	400	126.39	400	400	800	315.44	400	800	800	669.79
512	400	400	400	47.30	400	400	800	129.97	400	800	800	308.08	800	800	800	624.17
1024	400	400	800	59.18	400	800	800	212.37	800	800	800	437.97	800	800	1600	995.06
1	100	100	200	437.87	100	200	200	989.29	200	200	200	2122.42	200	200	400	4280.93
2	100	200	200	513.81	200	200	200	1078.89	200	200	400	2238.08	200	400	400	4579.28
4	200	200	200	661.82	200	200	400	1461.90	200	400	400	3251.23	400	400	400	6808.54
8	200	200	400	1273.53	200	400	400	2754.40	400	400	400	5792.10				
16	200	400	400	1374.05	400	400	400	2775.11	400	400	800	5615.87				
32	400	400	400	1294.36	400	400	800	2687.45	400	800	800	5642.81				
64	400	400	800	1296.88	400	800	800	2803.95	800	800	800	5882.27				
128	400	800	800	1409.56	800	800	800	2840.15	800	800	1600	5740.12				
256	800	800	800	1373.87	800	800	1600	2853.72	800	1600	1600	5854.39				
512	800	800	1600	1391.43	800	1600	1600	2941.25	1600	1600	1600	6153.34				
1024	800	1600	1600	1574.67	1600	1600	1600	3171.37								

TABLE II  
SPEED-UP ON SOONER.

$n_x$	$n_y$	$n_z$	$c$									
			2	4	8	16	32	64	128	256	512	1024
100	100	100	1.91	4.08	5.54	13.05	29.26	68.78	126.52	167.47	187.22	230.99
100	100	200	1.85	3.00	3.85	9.08	25.86	60.01	123.94	157.16	318.42	335.50
100	200	200	1.93	2.87	3.53	7.67	20.27	59.35	114.35	196.78	393.56	463.59
200	200	200	1.97	3.21	3.71	7.50	18.20	53.13	118.27	215.01	459.83	672.63
200	200	400	1.91	2.93	3.36	6.84	15.11	36.30	83.62	201.98	411.38	683.22
200	400	400	1.92	2.71	3.20	6.41	14.00	30.73	65.74	159.85	380.78	677.28
400	400	400	1.93	2.72	3.19	6.67	14.29	31.83	63.52	146.37	391.15	814.68

TABLE III  
PARALLEL EFFICIENCY ON SOONER.

$n_x$	$n_y$	$n_z$	$c$									
			2	4	8	16	32	64	128	256	512	1024
100	100	100	0.954	1.019	0.693	0.816	0.914	1.075	0.988	0.654	0.366	0.226
100	100	200	0.926	0.751	0.482	0.568	0.808	0.938	0.968	0.614	0.622	0.328
100	200	200	0.963	0.718	0.441	0.479	0.633	0.927	0.893	0.769	0.769	0.453
200	200	200	0.984	0.802	0.464	0.468	0.569	0.830	0.924	0.840	0.898	0.657
200	200	400	0.956	0.732	0.420	0.428	0.472	0.567	0.653	0.789	0.803	0.667
200	400	400	0.962	0.678	0.400	0.401	0.437	0.480	0.514	0.624	0.744	0.661
400	400	400	0.963	0.679	0.399	0.417	0.447	0.497	0.496	0.572	0.764	0.796

The somehow slower performance on Sooner using 8 cores is clearly visible. The same effect was observed during our previous work (see [24]). There are some factors which could play role for the slower performance using both processors and all available cores within each node. Generally they are a consequence of the limitations of the memory subsystems and their hierarchical organization in modern computers. One such factor might be the limited bandwidth of the main memory bus. This causes the processors literally to “starve” for data, thus, decreasing the overall performance. Since the L2 cache memory is shared among each pair of cores within the processors, this boost the performance of programs utilizing only single core within such pair (it can use the whole cache to

itself). Conversely, this leads for somehow decreased speedups when all cores are used. For memory intensive programs, these factors play crucial role for the codes’ performance. At this stage we have run into some technical problems attempting at running the code with specific number of cores per processor within each node. We will try to establish a way to explicitly evaluate this effect in the future.

To provide an analytical view on performance, the speed-up obtained on Sooner is reported in Table II and the parallel efficiency is shown in Table III. Here, let us recall that the discrete problem with  $n_x = n_y = n_z = 400$  requires 11 GB of memory and that is why we report the speed-up and the parallel efficiency on Sooner only for problems with  $100 \leq$

$n_x, n_y, n_z \leq 400$ . Specifically, for larger problems we could not run the code on a single computational unit (within a node with only 16 GB of memory) and thus neither speed-up nor efficiency could be calculated.

Increasing the number of cores, the parallel efficiency decreases on 8 cores, and after that it increases. This effect is particularly visible in the case of smaller problems. Specifically, a super-linear speed-up (and thus efficiency of more than 100%) is observed for  $n_x = n_y = n_z = 100$  on 4 and 64 cores. The main reasons for this fact can be related to the well-known fact that splitting a large problem into smaller sub-problems helps memory management. In particular, it allows for better usage of cache memories of individual parallel processors. Interestingly, the effect of performance dip on 8 cores is visible even for the largest reported problems (for  $n_x = n_y = n_z = 400$ ), where the efficiency increases all the way to  $c = 512$  cores. Overall, it can be stated that the performance of the code on the Sooner is more than promising for solving large problems using the proposed method.

Table IV represents execution times collected on the IBM Blue Gene/P machine at the Bulgarian Supercomputing Center. It consists of 2048 compute nodes with quad core PowerPC 450 processors (running at 850 MHz). Note that, here, a single node has only 4 cores (not 8 cores in 2 processors, as in the case of the Sooner). Each node has 2 GB of RAM (amount much smaller than the 16GB of RAM on the Sooner). For the point-to-point communications a 3.4 Gb 3D mesh network is used. Reduction operations are performed on a 6.8 Gb tree network (for more details, see <http://www.scc.acad.bg/>); thus the networking within the Blue Gene/P has much larger throughput than on the Sooner. We have used the IBM XL C compiler and compiled the code with the following options: “-O5 -qstrict -qarch=450d -qtune=450”. Again, no attempt at the two-level parallelization was made (in this case it would be even less worthy the effort, with only 4 cores per node). Due to the limits of memory available per node (2 GB), we did run into a more severe restrictions on the problem size than in the case of the Sooner. Therefore, the largest system that we were able to solve on a single node was for  $n_x = n_y = n_z = 200$ . However, in the case of the Blue Gene we were able to run jobs with up to 1024 nodes, which allowed us to solve large problems (up to size  $n_x = n_y = 1600, n_z = 3200$ ).

We observed that using 2 or 4 cores per processor leads to slower execution, e.g. the execution time for  $n_x = n_y = n_z = 800, c = 512$  is 982 seconds using 512 nodes, 1079.22 seconds using 256 nodes, and 1212.62 seconds using 128 nodes. This shows that (as expected) the communication between processors is faster than the communication between cores of one processor using the MPI communication functions.

In order to get better parallel performance we plan to align the decomposition of the computational domain into sub-domains, with the topology of the compute nodes in the Blue Gene connectivity network. In such way we will minimize the communication time in the parallel algorithm.

To complete the analysis of the performance of the IBM Blue Gene/P, Table V shows the obtained speed-up, while the

parallel efficiency is presented in Table VI. Recall, that due to memory limitations, the largest problem solvable on a single node was  $n_x = n_y = n_z = 200$ , thus limiting available speed-up and efficiency data. Observe that a super-linear speed-up is observed on up to 128 cores of the supercomputer. There are at least two causes for the higher speed-up: individual processors of the supercomputer are slower than these of the Sooner, while the communication is faster (due to the, above mentioned, special networking used in the Blue Gene). As a result, the single-processor data can be seen as relatively “slow” while in the case of multiple nodes the performance gain from using multiple processing units is boosted by the speed of the interconnect (combined with the decrease in sub-problem sizes). It is also worthy observing that as the problem size increases, the parallel efficiency increases as well (e.g. on 4096 cores, it raises from 21% to 44%). This again shows the overall parallel robustness of the approach under investigation.

Finally, we have decided to compare head-to-head both computers. To this effect, computing times obtained on both parallel systems are shown in Fig. 1, while the obtained speed-up are shown in Fig. 2.

Execution times on the Blue Gene/P are substantially larger than that on the Sooner (for the same number of cores); e.g. for the  $n_x = n_y = n_z = 200$  discrete problem on 64 cores the solution time on the Sooner is  $\sim 40$  seconds, whereas on the Blue Gene it is  $\sim 100$  seconds (and, recall that on the Blue Gene we observe a super-linear speed-up for up to 128 cores). This difference decreases, in relative terms, as the problems size and the number of cores increase; e.g. for the discrete problem of size  $n_x = 800, n_y = n_z = 1600$  the solution time on 256 cores of the Sooner is  $\sim 5854$  seconds, while on the Blue Gene it is  $\sim 8177$  seconds. In other words, the parallel efficiency obtained on the supercomputer is better. For instance, the execution time on single core on Sooner is 3.6 times faster than on the Blue Gene/P, in comparison with 1.4 times faster performance on 256 cores. This indicates, among others that the networking infrastructure of the Blue Gene supercomputer is superior to that of the Sooner cluster. Therefore, as the total number of nodes increases, the initial advantage of Sooner decreases. Interestingly, we have run some initial experiments on the IBM Blue Gene/P in the West University of Timisoara (for details, see <http://hpc.uvt.ro/infrastructure/bluegenep/>). The main hardware difference between the two machines is the 4GB per node memory of the Timisoara machine, which should result in it being more powerful. However, times obtained on that machine were substantially worse. When checking the reason we have found out that while the Sofia Center machine runs optimized LAPACK 3.2, the Timisoara Canter runs unoptimized LAPACK 3.3.1. Since we have run the same code using the same compiler options, this was the only difference we could spot. Obviously, we plan to investigate this issue further and will not report any obtained results. However, we mention this here as one of the “lessons learned.” Library software that is not fully optimized may degrade performance of a code and may not be immediately noticeable

TABLE IV  
EXECUTION TIME ON IBM BLUE GENE/P.

$c$	$n_x$	$n_y$	$n_z$	$T_c$	$n_x$	$n_y$	$n_z$	$T_c$	$n_x$	$n_y$	$n_z$	$T_c$	$n_x$	$n_y$	$n_z$	$T_c$
1	50	50	50	93.95	50	50	100	205.13	50	100	100	411.26	100	100	100	886.70
2	50	50	100	96.67	50	100	100	194.68	100	100	100	417.83	100	100	200	890.07
4	50	100	100	98.53	100	100	100	212.01	100	100	200	434.32	100	200	200	901.87
8	100	100	100	97.86	100	100	200	212.58	100	200	200	424.77	200	200	200	911.64
16	100	100	200	100.94	100	200	200	203.09	200	200	200	430.86	200	200	400	914.95
32	100	200	200	102.94	200	200	200	219.07	200	200	400	447.72	200	400	400	925.29
64	200	200	200	101.81	200	200	400	220.03	200	400	400	437.99	400	400	400	933.07
128	200	200	400	110.91	200	400	400	221.31	400	400	400	456.42	400	400	800	963.69
256	200	400	400	114.54	400	400	400	236.37	400	400	800	481.54	400	800	800	980.73
512	400	400	400	112.40	400	400	800	238.32	400	800	800	468.68	800	800	800	982.00
1024	400	400	800	126.38	400	800	800	249.90	800	800	800	494.18	800	800	1600	1048.29
2048	400	800	800	136.22	800	800	800	275.03	800	800	1600	593.02	800	1600	1600	1154.40
4096	800	800	800	170.64	800	800	1600	357.42	800	1600	1600	677.97	1600	1600	1600	1374.78
1	100	100	200	1822.27	100	200	200	3797.75	200	200	200	7715.32				
2	100	200	200	1813.75	200	200	200	3836.06	200	200	400	7660.58				
4	200	200	200	1865.64	200	200	400	3839.99	200	400	400	7749.17				
8	200	200	400	1867.87	200	400	400	3884.58	400	400	400	7870.17				
16	200	400	400	1862.39	400	400	400	3919.05	400	400	800	7810.51				
32	400	400	400	1904.62	400	400	800	3918.49	400	800	800	7874.35				
64	400	400	800	1907.69	400	800	800	3957.46	800	800	800	8006.56				
128	400	800	800	1961.68	800	800	800	4062.99	800	800	1600	8088.98				
256	800	800	800	1988.93	800	800	1600	4096.10	800	1600	1600	8177.80				
512	800	800	1600	1997.28	800	1600	1600	4119.41	1600	1600	1600	8269.49				
1024	800	1600	1600	2122.42	1600	1600	1600	4242.13	1600	1600	3200	8422.26				
2048	1600	1600	1600	2266.55	1600	1600	3200	4645.32								
4096	1600	1600	3200	2663.84												

TABLE V  
SPEED-UP ON IBM BLUE GENE/P.

$n_x$	$n_y$	$n_z$	$c$											
			2	4	8	16	32	64	128	256	512	1024	2048	4096
100	100	100	2.12	4.18	9.06	17.30	33.61	64.88	117.22	206.55	357.78	454.72	635.06	872.79
100	100	200	2.05	4.20	8.57	18.05	34.65	68.43	123.19	213.54	408.99	535.78	716.23	1156.77
100	200	200	2.09	4.21	8.94	18.70	36.89	72.23	130.94	240.37	427.99	620.19	966.68	1535.38
200	200	200	2.01	4.14	8.46	17.91	35.22	75.78	136.65	254.69	451.79	793.12	1263.89	1800.66

TABLE VI  
PARALLEL EFFICIENCY ON IBM BLUE GENE/P.

$n_x$	$n_y$	$n_z$	$c$											
			2	4	8	16	32	64	128	256	512	1024	2048	4096
100	100	100	1.061	1.046	1.133	1.081	1.050	1.014	0.916	0.807	0.699	0.444	0.310	0.213
100	100	200	1.024	1.049	1.072	1.128	1.083	1.069	0.962	0.834	0.799	0.523	0.350	0.282
100	200	200	1.047	1.053	1.118	1.169	1.153	1.129	1.023	0.939	0.836	0.606	0.472	0.375
200	200	200	1.006	1.034	1.058	1.119	1.101	1.184	1.068	0.995	0.882	0.775	0.617	0.440

by inexperienced user, who does not have multiple machines to run tests of her/his code on.

## V. CONCLUSIONS AND FUTURE WORK

We have studied parallel performance of the recently developed parallel algorithm based on a new direction splitting approach for solving of the 3D time dependent Stokes equation on a finite time interval and on a uniform rectangular mesh. The performance was evaluated on two different parallel architectures. Satisfactory parallel efficiency was obtained on both parallel systems, on up to 1024 processors. Out of the two machines, the faster CPUs on the Sooner lead to shorter run-time, on the same number of processors.

In the near future, it is our intention to consider and compare the performance of this algorithm to other efficient methods

for solving of the time dependent Stokes equation. In order to get better parallel performance using four cores per processor on the IBM Blue Gene/P (and future multi-core computers) we plan to develop mixed MPI/OpenMP code. Furthermore, we plan to synchronize the decomposition of the computational domain into sub-domains with the topology of the compute nodes in the Blue Gene connectivity network. In such way we will minimize the communication time in the parallel algorithm.

## ACKNOWLEDGMENTS

Computer time grants from the Oklahoma Supercomputing Center (OSCC) and the Bulgarian Supercomputing Center (BGSC) are kindly acknowledged. This research was partially supported by grants DO02-147 and DPRP7RP-02/13 from

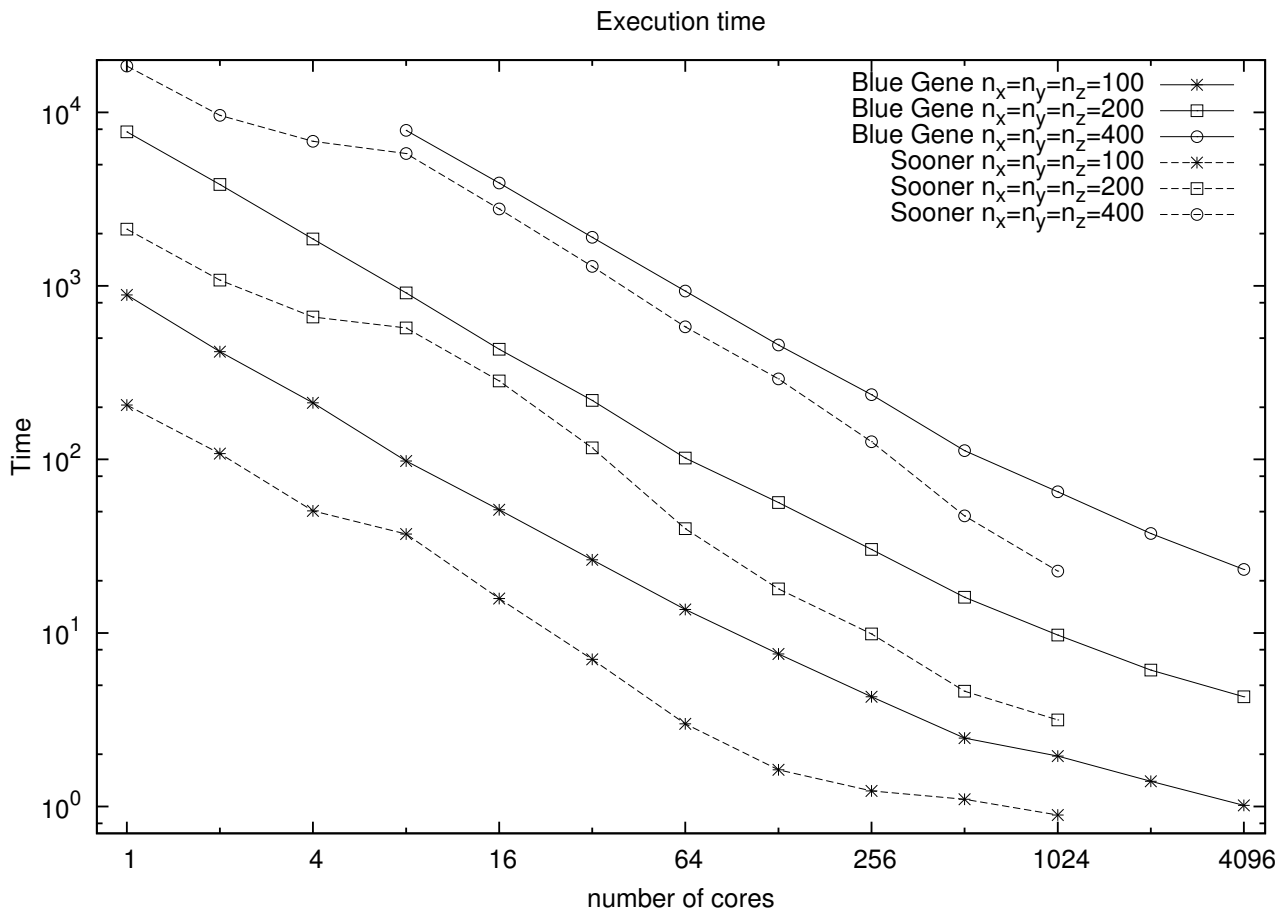


Fig. 1. Execution times for  $n_x = n_y = n_z = 100, 200, 400$ ; both computers

the Bulgarian NSF. Work presented here is a part of the Poland-Bulgaria collaborative grant “Parallel and distributed computing practices”.

#### REFERENCES

- [1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, D. Sorensen, LAPACK Users' Guide, Third Edition, SIAM, 1999.
- [2] C. E. Baumann, J. T. Oden, A discontinuous hp finite element method for the solution of the Euler and Navier-Stokes equations, *Int. J. Numer. Meth. Fluids*, **31**, 1999, 79–95.
- [3] M. Benzi, M. A. Olshanskii, An augmented Lagrangian-based approach to the Oseen problem, *SIAM Journal on Scientific Computing*, **28** (6), 2006, 2095–2113.
- [4] R. Biswas, K. D. Devine, J. E. Flaherty, Parallel, adaptive finite element methods for conservation laws, *Appl. Numer. Math.*, **14** (1–3), 1994, 255–283.
- [5] A. J. Chorin, Numerical solution of the Navier-Stokes equations, *Math. Comp.*, **22**, 1968, 745–762.
- [6] V. Dolejší, M. Feistauer, Ch. Schwab, On discontinuous Galerkin methods for nonlinear convection-diffusion problems and compressible flow, *Mathematica Bohemica*, **127** (2), 2002, 163–179.
- [7] H. C. Elman, D. Silvester, A. J. Wathen, *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluids Dynamics*, Oxford University Press, Oxford, 2005.
- [8] H. Elman, V. E. Howle, J. Shadid, D. Silvester, R. Tuminaro, Least squares preconditioners for stabilized discretizations of the Navier-Stokes equations, *SIAM Journal on Scientific Computing*, **30**, 2007, 290–311.
- [9] R. Eymard, R. Herbin, J. C. Latche, On a stabilized collocated Finite Volume scheme for the Stokes problem, *ESAIM: Math. Model. Numer. Anal.*, **40** (3), 2006, 501–527.
- [10] N. T. Frink, Recent progress toward a three dimensional unstructured Navier-Stokes flow solver. AIAA Paper No. 94-0061, 1994.
- [11] V. Girault, P.-A. Raviart, *Finite element methods for the Navier-Stokes equations: Theory and algorithms*, Springer, 1986.
- [12] R. Glowinski, *Numerical Methods for fluids (Part 3)*, *Handbook of Numerical Analysis*, Vol. IX, P.G. Ciarlet, J.L. Lions eds., North Holland, 2003.
- [13] J.-L. Guermond, P. Mineev, A new class of fractional step techniques for the incompressible Navier-Stokes equations using direction splitting, *Comptes Rendus Mathematique*, **348** (9–10), 2010, 581–585.
- [14] J.-L. Guermond, P. Mineev, J. Shen, An overview of projection methods for incompressible flows, *Comput. Methods Appl. Mech. Engrg.*, **195**, 2006, 6011–6054.
- [15] J.-L. Guermond, A. Salgado, A splitting method for incompressible flows with variable density based on a pressure Poisson equation, *Journal of Computational Physics*, **228** (8), 2009, 2834–2846.
- [16] J.-L. Guermond, A. Salgado, A fractional step method based on a pressure Poisson equation for incompressible flows with variable density, *Comptes Rendus Mathematique*, **346** (15–16), 2008, 913–918.
- [17] J.-L. Guermond, J. Shen, On the error estimates for the rotational pressure-correction projection methods, *Math. Comp.*, **73** (248), 2004, 1719–1737.

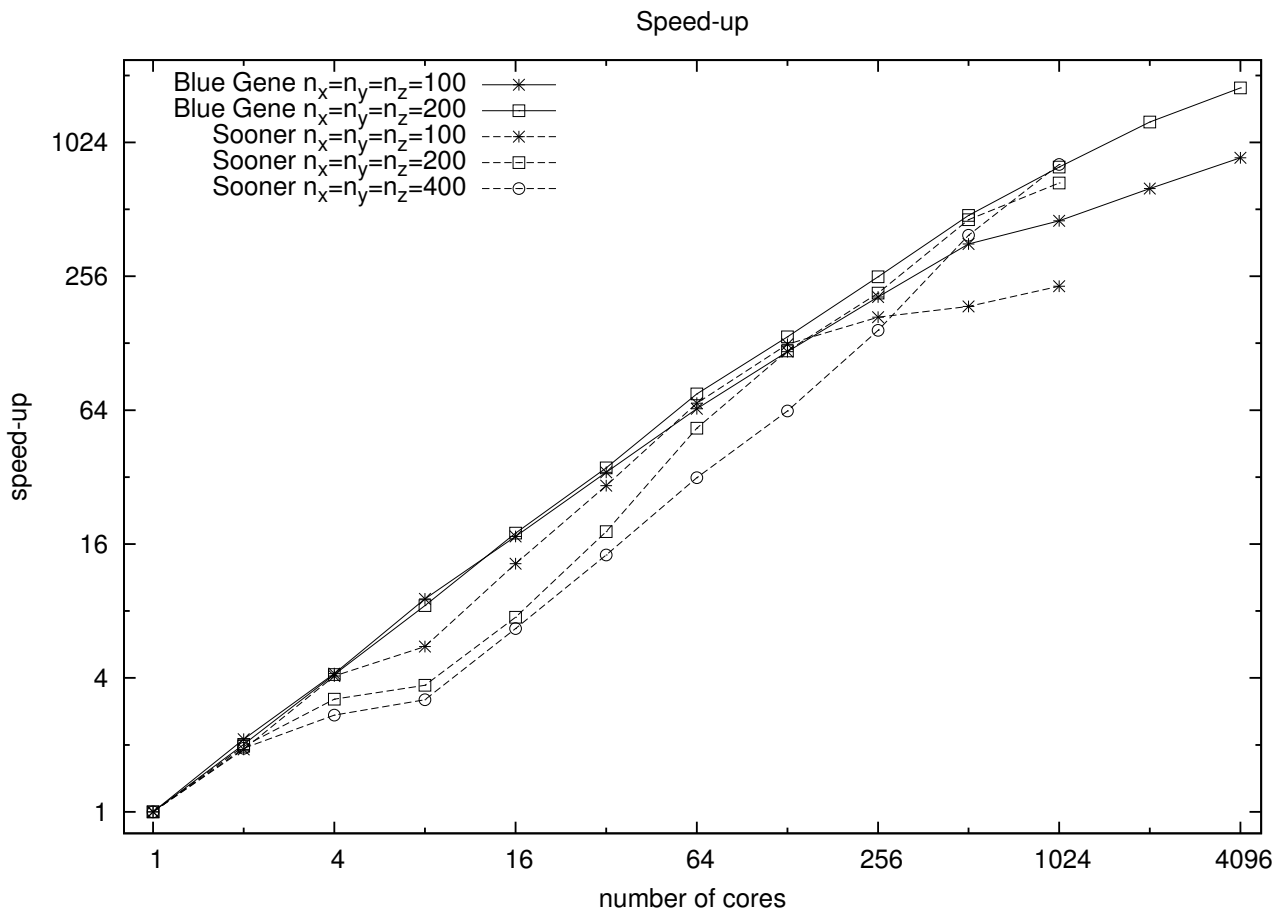


Fig. 2. Speed-up for  $n_x = n_y = n_z = 100, 200, 400$ ; both computers

- [18] M. D. Gunzburger, Finite element methods for viscous incompressible flows — A guide to theory, practice, and algorithms, Computer Science and Scientific Computing (Academic Press), 1989.
- [19] R. Hartmann, P. Houston, Symmetric Interior Penalty DG Methods for the Compressible Navier-Stokes Equations I: Method formulation, *Int. J. Num. Anal. Model.*, **3** (1), 2006, 1–20.
- [20] O. Hassan, K. Morgan, J. Peraire, An implicit finite element method for high speed flows. AIAA Paper No. 90–0402, 1990.
- [21] D. Kay, D. Loghin, A. Wathen, A preconditioner for the steady-state Navier-Stokes equations, *SIAM Journal on Scientific Computing*, **24** (1), 2002, 237–256.
- [22] D. A. Knoll, D. E. Keyes, Jacobian-free Newton-Krylov methods: a survey of approaches and applications, *Journal of Computational Physics*, **193**, 2004, 357–397.
- [23] R. J. LeVeque, Finite volume methods for hyperbolic problems. Cambridge University Press, 2002.
- [24] I. Lirkov, Y. Vutov, M. Paprzycki, M. Ganzha, Parallel Performance Evaluation of MIC(0) Preconditioning Algorithm for Voxel  $\mu$ FE Simulation, *Parallel processing and applied mathematics*, Part II, R. Wyrzykowski, J. Dongarra, K. Karczewski, J. Wańniewski ed., *Lecture notes in computer science*, **6068**, Springer, 2010, 135–144.
- [25] S. V. Patankar, Numerical Heat Transfer and Fluid Flow, *Series in Computational Methods in Mechanics and Thermal Sciences*, Mc Graw Hill, 1980.
- [26] S. V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, New York, 1980.
- [27] S. V. Patankar, D. A. Spalding, A calculation procedure for heat, mass and momentum transfer in three dimensional parabolic flows, *International Journal on Heat and Mass Transfer*, **15**, 1972, 1787–1806.
- [28] M. ur Rehman, C. Vuik, G. Segal, Preconditioners for the steady incompressible Navier-Stokes problem, *International Journal of Applied Mathematics*, **38**, 2008, 223–232.
- [29] M. ur Rehman, C. Vuik, G. Segal, SIMPLE-type preconditioners for the Oseen problem, *International Journal for Numerical Methods in Fluids*, **61**(4), 2009, 432–452.
- [30] Y. Saad, *Iterative Methods for Sparse Linear Systems* (2nd edn), SIAM, Philadelphia, PA, 2003.
- [31] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM Journal on Scientific and Statistical Computing*, **7**, 1986, 856–869.
- [32] M. Snir, St. Otto, St. Huss-Lederman, D. Walker, J. Dongarra, *MPI: the complete reference*, Scientific and engineering computation series. The MIT Press, Cambridge, Massachusetts, 1997, Second printing.
- [33] J. L. Steger, R. F. Warming, Flux vector splitting of the inviscid gas dynamics equations with application to finite difference methods. *J. Comput. Phys.*, **40** (2), 1981, 263–293.
- [34] R. Temam, Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires, *Arch. Rat. Mech. Anal.*, **33**, 1969, 377–385.
- [35] U. Trottenberg, C. Oosterlee, A. Schuller, *Multigrid*, Academic Press, San Diego, 2001.
- [36] S. Turek, *Efficient Solvers for Incompressible Flow Problems*, Springer, Berlin, 1999.
- [37] D. Walker, J. Dongarra, MPI: a standard Message Passing Interface, *Supercomputer*, **63**, 1996, 56–68.
- [38] P. Wesseling, *Principles of Computational Fluid Dynamics*, Springer Series in Computational Mathematics, **29**, Springer, New York, 2001.

# GPGPU calculations of gas thermodynamic quantities

Igor Mračka, Peter Somora and Tibor Žáčik

Department of Applied Mathematics

Mathematical Institute

Slovak Academy of Sciences

Štefánikova 49, 814 73 Bratislava, Slovakia

mracka@mat.savba.sk, somora@mat.savba.sk, zacik@mat.savba.sk

**Abstract**—Computational processors NVIDIA Tesla GPU based on the new Fermi generation of CUDA architecture are intended to perform massively parallel calculations applicable to various parts of the scientific and technical research, including the area of fluid dynamics modeling, in particular the simulation of real gas flow. In this paper we show that a significant acceleration of simulation calculations can be achieved even without the parallelization of the solution of involved differential equations by parallel pre-calculation of thermodynamic quantities using GPGPU.

**Index Terms**—gas state equation, AGA8, gas thermodynamic quantities, CFD, GPGPU, CUDA, FERMI, NVIDIA Tesla.

## I. INTRODUCTION

IT IS well-known that the computing power of processors keeps increasing by the (slightly modified) Moore's law [1], even though the focus of processor development has shifted from boosting the frequency to increasing the number of cores in processors. Main stream desktop processors contain four to eight cores. This new line of multi-core processors has changed the course of programming from serial towards parallel algorithms. Hand in hand with the increase of computer performance the demand arises for the increase of computation precision especially in scientific or technical applications [2]. However, in a large part of scientific problems the dependency between computational difficulty and calculation precision is "stronger" than linear and the computation time increases quadratically (or even faster) with increasing precision. In such cases the computer performance of common desktop multi-core CPUs is not sufficient. Fortunately, devices with a new architecture of General Purpose computation on Graphics Processing Units (GPGPU), usually termed as GPGPU processors, focused on massive parallel computations and capable of performing fast calculation in double precision is becoming available right now.

The new class of GPGPU massively parallel processors containing hundreds of cores is able to simultaneously process thousands of computational threads (a survey of general-purpose computations on graphics hardware is presented in [3]). Although this brings obvious advantages to scientific and technical computations, the developers must also deal with limitations of the new architecture (e.g., new memory classification, thread distribution between streaming multiprocessors,

etc.). An important aspect of the utilization of the GPGPU processors is the existence of development tools for the optimization and debugging of massively parallel algorithms. Nowadays, two probably biggest producers of GPGPU processors are AMD and NVIDIA. The new Radeon GPU processor series of AMD is based on the FireStream architecture [4] and contains more cores than the NVIDIA Tesla GPU processor based on the CUDA architecture codenamed FERMI ([5], [6]). Since the raw computational power and capabilities in double precision calculations of both class of processors are comparable, the main decision parameter for utilization could be the support of programming languages and development tools.

The NVIDIA CUDA architecture supports C, C++ and FORTRAN programming languages and several development environments (APIs) – CUDA C/C++, OpenCL, Direct Compute (and recently announced Microsoft C++ AMP) – thus ensuring a rising popularity among the scientific and technical community (see [7] and [8]). The spread of NVIDIA CUDA popularity is documented by a long series of papers and reports of scientific teams and institutions [2], [9], and has also reached the area of fluid dynamics modeling [10], [11], [12], [13], as proven by the increase of interest of scientific research centers in applying the NVIDIA Research Center program.

This paper is focused on the acceleration of (both steady-state and transient) simulation calculations of gas flow using approximations of values of thermodynamic quantities involved in the calculations. Several methods exist for the evaluation of values of thermodynamic quantities, each with its own range of application, accuracy and degree of computational difficulty. We show that, by pre-calculating the values of approximation matrices, it is possible to maintain a constant access time of the values of thermodynamic quantities during simulation independently of the method used. The short access time implies significant acceleration of simulation. Applying a more complex state equation of gas results in the increase of the time of calculation of each element of the approximation matrix. Increasing the approximation precision results in the increase of the matrix dimension and hence in a quadratic increase of the number of matrix elements to be evaluated. With this in mind, the utilization of massively parallel computations on GPGPU processors appears to be the ideal solution



for the evaluation of approximation matrices. Moreover, the parallelization itself is rather straightforward (regardless of the limitations caused by the transfer to GPU) without the need for any major changes in the implementation of the original algorithms. For the comparison of approximation matrices evaluation times the ratio between the time of GPGPU parallel evaluation and the time of one CPU core serial evaluation was taken. When considering parallel evaluation on multicore CPUs instead of serial evaluation, the resulting acceleration ratio will be correspondingly smaller, but not quite objective, since apart from the number of CPU cores the result also depends on the individual hardware configuration (activation/deactivation of hyper-threading technology, etc.) and the actual CPU usage of background running processes.

## II. THERMODYNAMIC QUANTITIES IN SIMULATION CALCULATIONS

The following system of partial differential equations (representing the conservation laws of mass, momentum and energy, respectively, [14]) describes the one-dimensional model of the turbulent flow of a mixture of gas with constant composition through a pipeline of constant inner diameter.

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho v}{\partial x} = 0,$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial}{\partial x} (\rho v^2 + P) + \frac{\lambda}{2D} \rho v |v| + \rho g \frac{\partial z}{\partial x} = 0,$$

$$\frac{\partial}{\partial t} \left[ \rho \left( \frac{v^2}{2} + h - \frac{P}{\rho} + gz \right) \right] + \frac{\partial}{\partial x} \left[ \rho v \left( \frac{v^2}{2} + h + gz \right) \right] + \alpha(T - T_w) \frac{\pi D}{S} = 0,$$

where

- $x$  – space coordinate,
- $t$  – time coordinate,
- $P$  – gas pressure,
- $T$  – gas temperature,
- $\rho$  – gas density,
- $v$  – gas velocity,
- $h$  – specific enthalpy,
- $\lambda$  – resistance coefficient,
- $D$  – pipeline diameter,
- $S$  – pipeline cross-section area,
- $z$  – pipeline altitude at given point,
- $g$  – standard gravity acceleration,
- $T_w$  – pipeline wall temperature.

Since in practical situations only the measurements of pressure  $P$ , temperature  $T$  and mass flow  $G$  are usually available, the gas velocity  $v$  is calculated using the following formula

$$v = \frac{G}{\rho S}.$$

The density is expressed using the state equation for the real gas

$$\rho = \frac{P}{ZRT},$$

where

- $R$  – gas constant for given gas composition,
- $Z$  – compressibility factor.

The change of enthalpy  $h$  occurring in the energy conservation law is given by the thermodynamic equation

$$dh = c_p dT - c_p \mu dP,$$

where

- $c_p$  – specific heat capacity at constant pressure,
- $\mu$  – Joule-Thomson coefficient.

In general, the state equation expresses the relation between the quantities  $P$ ,  $T$  and  $\rho$ , and is usually represented by a semi-empirical equation with the coefficients to be determined experimentally for some gas mixture, pressure and temperature ranges. Such approximated formulas are usually given the name of its authors, for example: Van der Waals, Redlich-Kwong, Peng-Robinson, Lee-Kesler, BWR (Benedict-Webb-Rubin) state equations. There are also various modifications of the original equations: Soave modification of the Redlich-Kwong equation or Elliott-Suresh-Donohue modification of the Peng-Robinson equation, etc. (see [15]). In some cases the equations get names after its institutions of origin, such as AGA (American Gas Association), see [16], or GERG (European Gas Research Group), see [17].

Specific heat capacity  $c_p$ , enthalpy  $h$ , and the Joule-Thomson coefficient  $\mu$  are thermodynamic quantities depending on the gas state at a given point in space and time and are functions of pressure and temperature (for given gas composition):  $c_p = c_p(P, T)$ ,  $h = h(P, T)$ ,  $\mu = \mu(P, T)$ , and  $Z = Z(P, T)$ . All thermodynamic quantities mentioned above (as functions of pressure and temperature) can be derived from the actual state equation.

By solving the above equations for a steady state – assuming the time-derivations equal zero – after some simplifications we obtain the following formula representing steady-state hydrodynamic conditions in one pipeline:

$$P_1^2 - aP_0^2 = bG|G|,$$

where  $P_0$  and  $P_1$  denote the pressures at the beginning and the end of pipeline, respectively, and the constants  $a$  and  $b$  are determined by

$$a = \exp\left(\frac{2g}{Z_{av}RT_{av}}\right),$$

and

$$b = \frac{\lambda L}{DS^2} \cdot Z_{av}RT_{av} \cdot \frac{a-1}{\ln a},$$

where  $Z_{av}$  and  $T_{av}$  are average values of compressibility and temperature along the pipeline, respectively. For steady state simulations, both the average temperature calculation and the distribution of temperature along the pipeline network involve

the use of specific heat capacity at constant pressure and the Joule-Thomson coefficient.

For both transient and steady state simulations, if the pipeline network contains compressor stations, the model of compressor involves the use of compressibility and the Poisson coefficient.

It follows that regardless of the type of simulation (transient or steady-state) the evaluation of thermodynamic quantities (compressibility, specific heat capacities, enthalpy, Joule-Thomson coefficient, Poisson coefficient) occurs quite frequently. For example, in the case of steady-state optimization simulations the time needed to acquire compressibility and other thermodynamic quantities from state equations of Redlich-Kwong or Peng-Robinson occupies approximately half the time of the overall calculation. If, in particular, simulating the flow of natural gas, then instead of general state equations one can use the AGA8 version of state equation the coefficients of which have been derived directly for gasses with components and concentrations typical for natural gas [16], [18]. By using AGA8 the resulting gas properties, e.g., compressibility, can be determined much more accurately, but the compressibility calculation algorithm is much more calculation-demanding than in the case of Redlich-Kwong or Peng-Robinson. In fact, the evaluation of thermodynamic quantities in steady-state simulations has been consuming more than 98 % of the overall calculation time.

Moreover, the overall calculation time has increased to such an extent, that it has started to pose a problem in practical use.

### III. APROXIMATION MATRICES

The solution of the problem could be the pre-calculation of thermodynamic quantities for selected values of pressure and temperature. The values for other pressures and temperatures can be obtained by interpolating the pre-calculated values.

Let  $(P_{\min}, P_{\max})$  and  $(T_{\min}, T_{\max})$  be intervals of pressures and temperatures respectively covering the values of pressure and temperature occurring in the calculations. For any concrete class of problems such intervals can be found.

Divide the intervals  $(P_{\min}, P_{\max})$  and  $(T_{\min}, T_{\max})$  into a given number of subintervals

$$P_{\min} = P_1 < P_2 < \dots < P_M = P_{\max}, \quad M \gg 1,$$

$$T_{\min} = T_1 < T_2 < \dots < T_N = T_{\max}, \quad N \gg 1.$$

Denote

$$\Delta P = P_2 - P_1, \quad \Delta T = T_2 - T_1.$$

By calculating the values of an arbitrary thermodynamic quantity in the points  $(P_i, T_j)$  one can obtain a matrix of numbers, which shall be called the approximation matrix of the selected quantity. The  $i$ -th row of the approximation matrix contains the values for the fixed pressure  $P_i$  and similarly, the  $j$ -th column contains the values for fixed temperature  $T_j$ .

The approximation matrix for compressibility

$$A_Z = \begin{pmatrix} Z(P_1, T_1) & Z(P_1, T_2) & \dots & Z(P_1, T_N) \\ Z(P_2, T_1) & Z(P_2, T_2) & & \\ \vdots & & \ddots & \vdots \\ Z(P_M, T_1) & Z(P_M, T_2) & \dots & Z(P_M, T_N) \end{pmatrix}$$

shall be called the compressibility matrix (and similarly for matrices of other thermodynamic quantities).

The request for a not pre-calculated value of compressibility is realized by means of bilinear interpolation (using the four closest grid points of the matrix) which is continuous and not computationally demanding.

That means, for the value of compressibility at  $(P, T)$  one has to find integers  $i, j$  and real numbers  $x, y$ , ( $0 \leq x < 1$ ,  $0 \leq y < 1$ ) such that

$$P = i \cdot \Delta P + x \quad \text{and} \quad T = j \cdot \Delta T + y.$$

Then for the  $(2 \times 2)$  submatrix

$$A_{ij} = \begin{pmatrix} Z(P_i, T_j) & Z(P_i, T_{j+1}) \\ Z(P_{i+1}, T_j) & Z(P_{i+1}, T_{j+1}) \end{pmatrix},$$

the bilinear interpolation reads

$$Z(P, T) \doteq (1-x \quad x) \cdot A_{ij} \cdot \begin{pmatrix} 1-y \\ y \end{pmatrix}$$

(see Fig. 1).

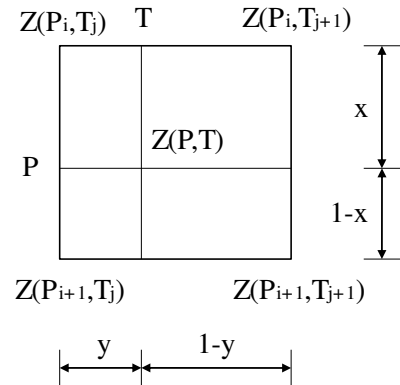


Fig. 1. Bilinear interpolation of  $Z(P, T)$  from the four closest points of compressibility matrix.

It is obvious, that the increase of the number of grid points,  $M \times N$ , results in the increase of accuracy of the interpolated compressibility values. In the case of optimization calculations it is usually sufficient for the neighboring pressure grid points to be ca. 5 kPa apart and the neighboring temperature points to be ca. 0.1 °C apart. Thus, when covering the pressure interval from 0 to 10 MPa and temperature interval from -20 to 80 °C, the resulting matrix is of dimension  $2,000 \times 1,000$  hence requiring 2,000,000 calculations of compressibility (see Fig. 2). A serial calculation of the compressibility matrix according to AGA8 can take tens of seconds on a common CPU. When solving an assignment with fixed gas composition

the matrix can be pre-calculated and the calculation obtains the required compressibility values by interpolation.

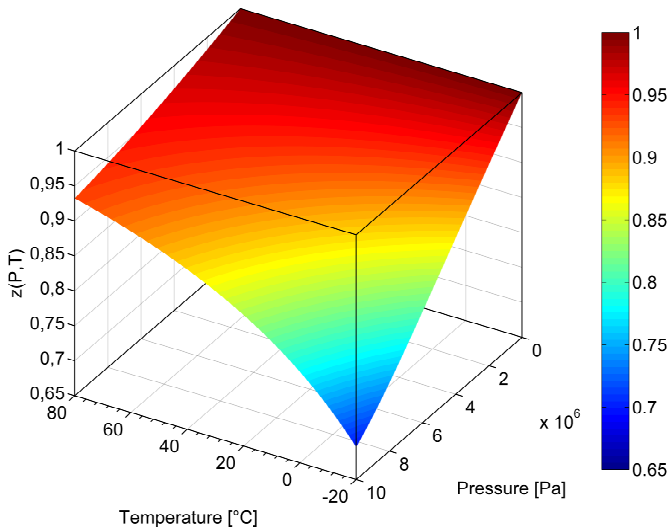


Fig. 2. Visualization of compressibility calculated from AGA8 state equation as function of pressure and temperature.

By using the approximation matrices we have achieved (even if slightly compromising the accuracy) a constant access time to the values of thermodynamic quantities. The time needed to calculate compressibility by bilinear interpolation is smaller than the time needed to directly evaluate compressibility from the more simple equations of Redlich-Kwong and Peng-Robinson. Hence the approximation matrix approach is suitable for any algorithm of thermodynamic values acquisition. From the time-consumption point of view, the use of approximation matrices is the same regardless of the state equation used in the simulation calculations.

However, the use of approximation matrices is relevant up to the point of gas composition change. Every change of gas composition requires the values of the matrices to be re-calculated. For example, when solving a number of optimization tasks with each task having its own gas composition, the continuous re-calculation of approximation matrices is from the time-consumption point of view unacceptable.

#### IV. GPU-BASED CALCULATIONS

The evaluation of matrices of thermodynamic quantities consists of a large number (of order  $10^6$ ) of mutually independent calculations. Among the inputs for the algorithm of evaluation are the following: a vector of pressures and temperatures and a matrix of constants derived from the properties of each component of the overall mixture of gas (the matrix of thermodynamic quantities is evaluated for a given or constant gas composition). The output of the algorithm is a matrix of values of a selected thermodynamic quantity (compressibility, enthalpy, thermal capacity, etc.). Even though

the algorithm itself is relatively small (less than 8 KB after compilation of AGA8 algorithm), it contains a large number of standard operations—multiplications, divisions and evaluations of analytic functions (powers, square roots, logarithms) – that have to be performed in double precision. For example one evaluation of compressibility of a mixture of natural gas with twelve components based on the AGA8 state equation requires approximately five hundred calculations of analytic functions (200 powers, 300 square roots) and approximately ten thousand standard operations (+, −, \*, /).

One can clearly see that the problem of evaluation of state matrices is a typical case for massive parallelization within processors supporting the parallel processing of a large number (more than a thousand) of computation threads. Current processors in desktop PCs now usually contain four cores and allow the simultaneous processing of four to eight threads (for example, Intel Core i7 950, see [19]).

This was our main stimulus for deciding to use the massive parallel processors Tesla GPU based on the Fermi architecture (see [5]) that are supporting the CUDA C/C++ interface environment (see [7]) stemming from C++ and hence allowing a smooth transition of C++ algorithms onto the GPGPU platform while at the same time satisfying the necessary condition of performing mathematical operations (and analytic functions evaluation) in double precision. Compared to ordinary CPUs this hardware allows the parallel processing of a far larger number of computation threads simultaneously. More precisely, the Fermi architecture Tesla GPU processors consist of fourteen to sixteen streaming multiprocessors (SM) each containing thirty two cores, hence capable of processing at least 448 computation threads parallelly. Each SM unit has its own exclusive L1 cache and a shared memory (see [20], [6]) enabling it to optimize parallel processing by reorganizing the calculations into blocks such that the computation threads running on the same SM unit are sharing partial results. Another important part of the Tesla GPU processors is the Giga Thread planner distributing the blocks of threads to subordinate SM planners thus ensuring a uniform utilization of all SM units in the GPU processor. For full utilization of SM units in thread processing it is possible to define blocks containing up to 1024 computation threads. The overall number of blocks that can be processed in one request of parallel evaluation on a GPGPU processor (i.e., CUDA kernel) is “limited” by the size of the so called three-dimensional grid of blocks that is,  $65535^3$ , allowing the evaluation of the whole state matrix in a single request.

Main advantages of Tesla GPU processors [5] (compared to standard Intel Core i7 950 3.07 GHz PC CPU [19]):

- 14–16 SM units with 32 cores each = 448–512 total cores (PC CPU: 4–8 cores)
- massive raw power in floating point calculations – up to 1 TFLOPS (PC CPU: 70 GFLOP)
- large ECC internal memory of 3–6 GB unconstrained by operating system requests
- large throughput between SM units and global memory – up to 150 GB/s (PC CPU: 20 GB/s)

- large number of parallel computation threads possible to process – in so called data parallel algorithms (PC CPU: 4–8 threads)
- several parallel programming languages / environments support – CUDA C/C++, OpenCL, FORTRAN, Direct-Compute
- development tools for debugging and optimization of parallel algorithms – NVIDIA Nsight, NVIDIA Compute Visual Profiler

A notable disadvantage of the GPGPU processors compared to standard PC processors is a limited amount of resources such as constant memory, relatively small cache and a shared memory in SM units. The primary bottleneck in GPU calculations appears to be the throughput of the PCI-Express bus. For example, with SM unit frequency 1.147 GHz and fourteen SM units per processor the system is able to process up to  $512 \times 10^9$  standard operations in double precision per second. If every evaluation of a state quantity would require the transfer of the complete task assignment (i.e., pressure, temperature and gas composition determined constants) occupying about two hundred bytes of memory, then such transfer through the PCI-Express bus (with speed ca. 4 GB/s) would take approximately 1/20 of a microsecond, nevertheless within that time the GPGPU processor could perform up to 25,000 standard operations in double precision! It follows that it is extremely important to minimize the amount of data transferred between the host PC and the GPGPU device.

Main disadvantages of Tesla GPU processors:

- limited memory resources of SM units – 64 KB for L1 cache and shared memory,
- low throughput of the PCI-Express bus (4–6 GB/s) – unsuitable for simple calculations where the ratio of the number of operations or against the number of bytes of task assignment and task output for one calculation is less than 100 : 1.

The problem of thermodynamic quantities matrix evaluation consists of a sufficient amount of independent calculations and is suitable for massive parallelization. Computational algorithm (AGA8) of state quantities is not particularly demanding with respect to the operational and constant memory so the resources of GPGPU processor are sufficient and no special algorithm tuning for GPGPU is needed. For example, the result matrix containing  $2,000 \times 1,000$  values occupies 16 MB of memory which takes only ca 2 milliseconds to transfer from GPU to host PC. So the problem is an ideal candidate for maximal potential utilization of the massively parallel Tesla GPU processor, with the possible task runtime acceleration factor reaching 60 compared to serial evaluation of the matrix on a standard reference PC (Intel Core i7 950 3.07 GHz CPU).

Compared to the serial evaluation, the parallelization of the calculations on standard four core CPUs accelerates the task runtime approximately three to five times – the exact number being heavily dependent on the CPU configuration (activation/deactivation of hyper-threading technology, etc.) and the actual CPU usage of background running processes.

Such parallel CPU-based acceleration of the approximation matrices calculations is in no way sufficient for our objectives.

The transformation of approximation matrices calculation from CPU to GPU only requires the replacement of the matrix grid point evaluation cycle with the CUDA api function (CUDA kernel [21]) providing the distribution of individual evaluations to the GPU cores.

## V. RESULTS

The simulation calculations were performed using a model of the Slovak transit gas pipeline network consisting of approximately two thousand three hundred kilometers of pipelines, four compressor stations each containing more than twenty compressors of various types, fourteen pressure regulator valves and circa four hundred block valves. As the state equation AGA8 was used, with the pressure range 8 to 10 MPa and temperature range  $-10$  to  $+60$  °C.

Without the use of approximation matrices the steady-state simulation runtime was 22.50 s. With the use of approximation matrices the same simulation runtime was 0.45 s (not including the time for the pre-calculation of approximation matrices) which represents a 50-fold acceleration. The simulation runtime is not affected by the change of state matrix dimensions. For each new gas composition the pre-calculation of six approximation matrices dimensioned  $2,000 \times 1,000$  was required (each for one quantity: compressibility, enthalpy, Joule-Thomson coefficient, viscosity, Poisson coefficient, specific heat capacities at constant pressure). This takes 21.45 s when calculating on CPU. Most of the time, 17.50 s, is consumed by the compressibility matrix calculation.

The overall calculation time combining the steady-state simulation time with the approximation matrices pre-calculation time was  $0.45 + 21.45 = 21.90$  s representing a 1.03-fold acceleration compared to the case not using the approximation matrices. Hence the approximation matrices evaluation consumes the majority of calculation time where the compressibility matrix evaluation takes about 80 % of the overall calculation time. It follows that for the sake of acceleration of calculations with varying gas composition the acceleration of approximation matrices evaluation is critical (especially for compressibility matrix).

The transient simulation calculation (four hours of transient gas transport) without the approximation matrices has taken 248.50 s, while with the use of approximation matrices the time taken was 38.90 s. The number of compressibility evaluations was approximately  $20 \cdot 10^6$ .

Fig. 3 depicts the area of the compressibility matrix used in the steady-state simulation optimization calculations with the assignment being the maximization of mass flow through the gas pipeline network. The color indicates the number of evaluations of gas properties for an area with dimensions  $\Delta P = 5 \text{ kPa}$  and  $\Delta T = 0.1$  °C uniformly covering the used cartesian product  $[0 \text{ MPa}, 8 \text{ MPa}] \times [-10 \text{ °C}, 60 \text{ °C}]$ . The overall task time was 387 s with more than  $2.27 \cdot 10^9$  evaluations of compressibility for given  $P$  and  $T$ .

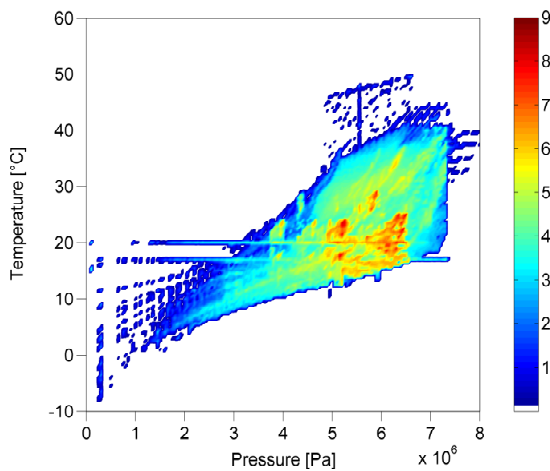


Fig. 3. Example of using of compressibility matrix in maximal gas pipeline network mass flow calculation. (Color scale indicates the exponent in the power of 10.)

It follows from Fig. 3 that the area used in the simulation occupies a significant part of the area covered by the approximation matrices.

The GPGPU acceleration of the approximation matrix evaluation was demonstrated using the calculation of the compressibility matrix based on the AGA8 algorithm, since the compressibility matrix evaluation consumes most of the overall calculation time.

The state matrix evaluation time depends on the matrix dimensions. Table 1 shows the comparison of compressibility matrix calculation times for CPU (serial calculation on Intel Corei7 950) and GPGPU processor (parallel calculation on NVIDIA Tesla C2050 GPU) and Table 2 shows the comparison of their acceleration factors. For additional comparison the results achieved on NVIDIA GTX 570 (with more cores than Tesla C2050, but with limited FERMI architecture) [22] are also stated. The results achieved on CPU are not satisfactory in the case of varying gas composition. Evaluation of the approximation matrix consists of a large number of independent calculations parameterized by pressures and temperatures. Such task is typically suitable for massive parallelization and hence ideal for GPGPU processing. By using massively parallel calculations on GPGPU processor we were able to cut down the evaluation time of state matrices (of size  $2,000 \times 1,000$ ) from tens of seconds (18 s) to under one second (0.3 s), providing a satisfactory level of simulation precision.

In matrices with the number of compressibility evaluations under 2500 the calculation times on CPU are comparable to the selected GPGPU device. We have focused on cases where the compressibility evaluation count exceeded  $10^4$  (matrices with dimension  $100 \times 100$  and more), as illustrated by the calculation times (Fig. 4) and calculation acceleration factors (Fig. 5) for the compressibility matrix evaluation. As one can see from Fig. 5, in the range of evaluations from  $10^5$  to  $10^6$  the acceleration factor of the GPGPU processor over CPU increases sharply. This corresponds to the area of evaluation

TABLE I  
COMPARISON OF COMPRESSIBILITY MATRIX CALCULATION TIMES USING AGA8 ALGORITHM.

$M \times N$	Corei7 950 (serial)	Tesla C2050 (parallel)	GTX 570 (parallel)
	[s]	[s]	[s]
$500 \times 250$	1.14	0.05	0.08
$1,000 \times 500$	4.39	0.1	0.18
$2,000 \times 1,000$	17.88	0.32	0.55
$4,000 \times 2,000$	70.26	1.16	1.84

TABLE II  
COMPARISON OF ACCELERATION FACTORS OF COMPRESSIBILITY MATRIX CALCULATION USING AGA8 ALGORITHM.

$M \times N$	$T_{\text{Tesla}}/T_{\text{CPU}}$	$T_{\text{GTX}}/T_{\text{CPU}}$
$500 \times 250$	22×	14×
$1,000 \times 500$	42×	25×
$2,000 \times 1,000$	56×	33×
$4,000 \times 2,000$	60×	38×

counts where the parallel calculation is too small to utilize the full potential of the GPGPU processors. Beyond the value of  $5 \cdot 10^6$  evaluation counts one can see the slowing down (stabilization) of the acceleration factor increase corresponding to the area of evaluation counts where the potential of the GPGPU device is assumed to be fully utilized.

Concluding, the GPGPU parallelization is optimal beyond some value of evaluation counts. The upper bound for the evaluation counts is provided by the maximal allowed calculation time or by memory limits allocated for results. In the case of the evaluation of compressibility matrix with  $2 \cdot 10^6$  elements (regarded as sufficiently large to provide satisfyingly precise results) the potential of the GPGPU device is still not fully used.

An increase of the matrix dimension to  $4,000 \times 2,000$  (acceleration factor increases) would cause a slight increase of the overall calculation time (to 1.16 s), but would quadratically increase the amount of memory for pre-calculated matrices (to 64 MB for each) which is not desirable in our case. If one would assume a two-time increase in precision and four-time increase in memory consumption compared to the current “optimal” case, then the achieved full utilization of the potential of the GPGPU device is represented by an acceleration factor of 60 (for NVIDIA Tesla C2050 GPU).

## VI. CONCLUSION

The focus of the paper rests on the problem of performance optimization in gas transit simulations, i.e., the calculation of thermodynamic quantities based on a given state equation of real gas. We solved this problem by pre-calculating the thermodynamic quantities matrices which provides independence from computational difficulty of the used gas state equation. The evaluation of a thermodynamic quantity in simulation calculations for given gas composition is performed as the bilinear interpolation from nearest grid points of the respective

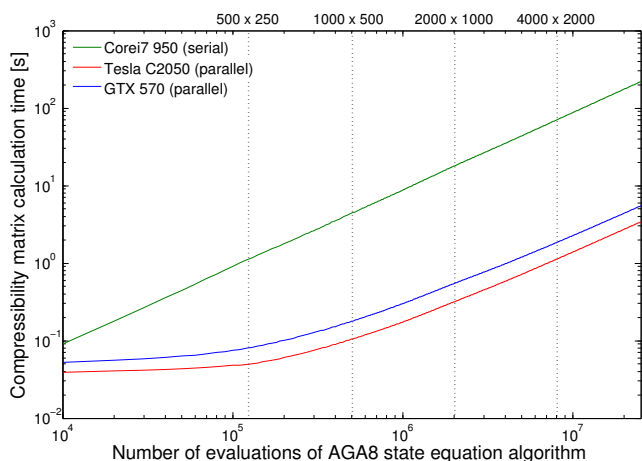


Fig. 4. Compressibility matrix calculation times on CPU and GPU in correspondence with compressibility evaluation counts through direct AGA8 algorithm calculation. (lower value is better)

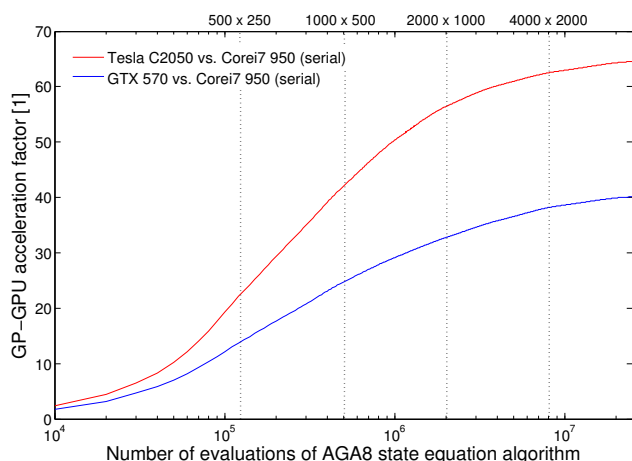


Fig. 5. Acceleration factor of GPU over CPU for compressibility matrix calculation in correspondence with compressibility evaluation counts through direct AGA8 algorithm calculation. (higher value is better)

pre-calculated matrices in much faster time than in the case of precise calculation of the given quantity, but with some degree of inaccuracy caused by small dimensions of the thermodynamic quantity matrix. In many cases the inaccuracy of the used real gas state equation is larger than the inaccuracy caused by bilinear interpolation even for small thermodynamic quantity matrices (e.g.,  $200 \times 200$ ). However, when considering more precise (and more computationally demanding) state equations, e.g., AGA8, GERG, the inaccuracy caused by the small dimensions of the approximation matrices becomes the limiting factor. Hence, choosing a more accurate state equation is not sufficient in order to achieve more precise simulations, but one also has to increase the dimensions of the approximation matrices. Since the computational difficulty increases quadratically with the size of the matrix, one has

to find a way to shorten the time required for the pre-calculation of the approximation matrices. A very suitable way appears to be the use of the new massively parallel GPGPU processors (NVIDIA Tesla C2050 GPU). We were able to shorten the time of pre-calculation of the compressibility matrix ( $4,000 \times 2,000$ ) from 70 s in serial evaluation to 1.16 s in parallel evaluation (i.e., accelerated more than 60 times). Besides the evaluation of the thermodynamic quantities matrices there are other interesting problems in fluid dynamics (e.g., gas transport optimization, leak detection, etc.) that can benefit from the enormous potential of GPGPU. Additional promising predictions of the GPGPU hardware manufacturers state that the GPGPU potential shall increase several times compared to its current status (NVIDIA CEO on its GPU Technology Conference in 2010 presented a roadmap with the new CUDA architecture codenamed Maxwell, to be introduced in 2013, with performance about 10 times better than the current FERMI architecture [23]). From our point of view, all these facts forecast a bright future for the utilization of GPGPU technology in scientific and technical applications.

#### ACKNOWLEDGMENT

This work was supported by VEGA grant 2/0124/10.

#### REFERENCES

- [1] Moore's Law Made real by Intel Innovations <http://www.intel.com/technology/mooreslaw/>
- [2] High Performance Computing – Supercomputing with Tesla GPUs, [www.nvidia.com/tesla](http://www.nvidia.com/tesla)
- [3] J. D. Owens, D. Luebke, N. Govindaraju, M. Harris, J. Krger, A. E. Lefohn, T. J. Purcell, "A Survey of General-Purpose Computation on Graphics Hardware," *Computer Graphics Forum*, Vol. 26, No. 1, 2007, pp. 80–113.
- [4] GPU Computing: Past, Present and Future with ATI Stream Technology, [http://developer.amd.com/gpu\\_assets/GPU Computing - Past Present And Future with ATI Stream Technology.pdf](http://developer.amd.com/gpu_assets/GPU%20Computing%20-%20Past%20Present%20And%20Future%20with%20ATI%20Stream%20Technology.pdf)
- [5] Tesla C2050/C2070 GPU computing processor, [http://www.nvidia.com/docs/IO/43395/NV\\_DS\\_Tesla\\_C2050\\_C2070\\_jul10\\_lores.pdf](http://www.nvidia.com/docs/IO/43395/NV_DS_Tesla_C2050_C2070_jul10_lores.pdf)
- [6] Next-Generation GPU Architecture – 'Fermi', [http://www.lunarc.lu.se/Documents/nvidia-workshop/files/presentation/45\\_Fermi.pdf](http://www.lunarc.lu.se/Documents/nvidia-workshop/files/presentation/45_Fermi.pdf)
- [7] NVIDIA CUDA Compute Unified Device Architecture 2.0 Programming Guide, 2008.
- [8] Khronos Group, The OpenCL Specification, Version 1.0, 2009.
- [9] W. W. Hwu, "GPU Computing Gems," Morgan Kaufmann Publishers (is an imprint of Elsevier), Burlington, MA, USA, 2010.
- [10] N. Goodnight, CUDA/OpenGL Fluid Simulation, NVIDIA Corporation, 2007.
- [11] J. M. Cohen, M. J. Molemaker, "A Fast Double Precision CFD Code Using CUDA," *Proceedings of Parallel CFD*, 2009.
- [12] E. Phillips, Y. Zhang, R. Davis, J. Owens, "CUDA Implementation of a Navier-Stokes Solver on Multi-GPU Desktop Platforms for Incompressible Flows," 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, No. AIAA 2009-565, Orlando, FL, USA, January 2009.
- [13] J. C. Thibault, I. Senocak, "CUDA Implementation of a Navier-Stokes Solver on Multi-GPU Desktop Platforms for Incompressible Flows," 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, No. AIAA 2009-758, Orlando, FL, USA, January 2009.
- [14] L. D. Landau, E. M. Lifshitz, "Fluid Mechanics (Volume 6 of Course of Theoretical Physics Second English Edition), Pergamon Press, Maxwell House, Fairview Prak, Elmsford, NY, USA, 1987.

- [15] Y. S. Wei and R. J. Sadus, "Equation of State for the Calculation of Fluid-Phase Equilibria," *AICHE Journal Review*, vol. 46, No. 1, January 2000, p. 169–196.
- [16] M. Farzaneh-Gord, A. Khamforoush, S. Hashemi, H. P. Namin, "Computing Thermal Properties of Natural Gas by Utilizing AGA8 Equation of State," *International Journal of Chemical Engineering and Applications*, vol. 1, No. 1, June 2010.
- [17] O. Kunz, R. Klimeck, W. Wagner, M. Jaeschke, "The GERG-2004 Wide-Range Equation of State for Natural Gases and Other Mixtures," GERG technical monograph, Publishing House of the Association of German Engineers, Germany, 2007.
- [18] K. E. Starling, J. L. Savidge, "Compressibility Factor of Natural Gas and Other Related Hydrocarbon Gases," *Report No. 8 Software of American Gas Association(AGA)*, 3rd printing, November 2003.
- [19] Intel Core i7-900 Desktop Processor Extreme Edition Series and Intel Core i7-900 Desktop Processor Seriesm, <http://download.intel.com/design/processor/datashts/320834.pdf>
- [20] D. B. Kirk, W. W. Hwu, "Programming Masively Parallel Processors," Morgan Kaufmann Publishers (is an imprint of Elsevier), Burlington, MA, USA, 2010, pp. 8.
- [21] CUDA Programming Model Overview, 2008. <http://www.sdsc.edu/us/training/assets/docs/NVIDIA-02-BasicsOfCUDA.pdf>
- [22] NVIDIA Geforce GTX 570 GPU Datasheet, <http://www.nvidia.com/docs/IO/102043/GTX-570-Web-Datasheet-Final.pdf>
- [23] NVIDIA GPU Technology Conference 2010 Jen-Hsun Huang Keynote <http://blogs.nvidia.com/2010/09/gpu-technology-conference-liveblog-jen-hsun-huang-keynote/>



# The influence of a matrix condition number on iterative methods' convergence

Anna Pyzara    Beata Bylina    Jarosław Bylina

Institute of Mathematics

Marie Curie-Skłodowska University

Pl. M. Curie-Skłodowskiej 5, 20-031 Lublin, Poland

Email: anna.pyzara@gmail.com, beata.bylina@umcs.pl, jaroslaw.bylina@umcs.pl

**Abstract**—We investigate condition numbers of matrices that appear during solving systems of linear equations. We consider iterative methods to solve the equations, namely Jacobi and Gauss-Seidel methods. We examine the influence of the condition number on convergence of these iterative methods. We study numerical aspects of relations between the condition number and the size of the matrix and the number of iterations experimentally. We analyze random matrices, the Hilbert matrix and a strictly diagonally dominant matrix.

## I. INTRODUCTION

THE CONDITION number plays an important role in the numerical linear algebra. The condition number measures the sensitivity of the solution of a problem to perturbations in the data. It provides an approximate upper bound on the error in a computed solution. The condition number depends on the norm used. The condition number can also be used to predict the convergence of iterative methods.

A lot of articles [1], [2], [3], [7] deal with problems associated with the condition number in terms of mathematical theory. In those papers authors did not examine experimental and numerical aspects of the condition number. This paper investigates numerical aspects of relations between the condition number and the size of the matrix and the number of iterations experimentally.

The convergence of iterative methods depends on the condition number of the coefficient matrix describing the system. That is the subject of this work. In order to determine its impact on the convergence of an iterative process, the accuracy of the Jacobi and Gauss-Seidel method will be tested. We examine some kinds of special matrices, namely random matrices, the Hilbert matrix, a strictly diagonally dominant matrix and relations between the size of the matrix, the condition number and the convergence of iterative methods.

The paper is set up as follows. In Section II the condition number of a matrix have been presented [4], [5], [6]. Section III is a report from an experimental research about the condition number. It contains the results of tests on the condition number of random matrices, the Hilbert matrix [2], a strictly diagonally dominant matrix. It shows the results of

testing the accuracy of iterative methods on the basis of which the convergence of these methods is determined. Section IV concludes the paper.

## II. THE CONDITION NUMBER

The convergence of an iteration process and the existence of an equation linear system solution depends on the form of the matrix  $A$ . If the square matrix of the system  $Ax = b$  is singular, then this system does not have a solution. On the contrary, when  $\det A \neq 0$ , it is the condition number of a matrix that decides about the convergence of the approximate solution, obtained in the iteration process, to the correct solution of the equation system. The condition number is defined by a formula:

$$\kappa(A) = \|A\| \cdot \|A^{-1}\|.$$

The value of condition number is dependent on the choice of a matrix norm, and indirectly on the choice of a vector norm. When it is necessary to show this choice we use the following symbols:  $\kappa_\infty(A)$  (the chosen norm is infinity norm — of  $l_\infty$ ),  $\kappa_1(A)$  (for the matrix norm of  $l_1$ ); and so on.

The condition number is strongly connected to the accuracy of the approximate solution of the equation system. In order to calculate the accuracy, we must appoint a residual vector, with the aid of a formula:

$$r = b - Ax^{(k)}$$

where  $A$  is a square matrix of the system  $Ax = b$ ,  $b$  is a free elements' vector, and  $x^{(k)}$  is the  $k$ -th approximation solution of the system, which is calculated with the use of numerical methods. Taking the norm of the residual vector, we get a number which defines the accuracy with which the obtained solution approximates the correct value of the solution. This accuracy can be calculated with the use of:

$$\varepsilon = \|r\| = \|b - Ax^{(k)}\|.$$

A condition number of a matrix is always bigger than or equal to 1. Equality is reached when the obtained solution is the correct solution; for example, a condition number of the unit matrix is equal to 1. If a condition number is not “too big”, then the matrix  $A$  is said to be well-conditioned. It means that the result was obtained with a good accuracy. A matrix

This work was partially supported within the project N N516 479640 of the Ministry of Science and Higher Education of the Polish Republic (MNiSW) *Modele dynamiki transmisji, sterowania zatkanieniem i jakością usług w Internecie*.

with a big condition number  $\kappa(A)$  (called an ill-conditioned matrix) can generate approximations with a huge error.

It is not precisely defined what means “too big” in the above context. This paper hopes to help understand it.

An example of an ill-conditioned matrix is the matrix of the following system:

$$\begin{cases} 10^{-18}x_1 + x_2 = 1, \\ x_1 + 2x_2 = 4. \end{cases} \quad (1)$$

Its correct solution is

$$\begin{aligned} x_1 &= 2.000000000000000004, \\ x_2 &= 0.99999999999999997999999999999999996. \end{aligned}$$

On the other hand, with the use of the iterative Jacobi method, after 10 iteration steps, we get

$$\begin{aligned} x_1 &= -6.21875000000000 \cdot 10^{88}, \\ x_2 &= -3.09375000000000 \cdot 10^{88}. \end{aligned}$$

We can easily notice that these approximations differ considerably from the real solutions.

### III. NUMERICAL EXPERIMENTS

The main goal of the tests was to check the behavior of the condition number for some matrices and its influence on the accuracy of the iterative Jacobi and Gauss-Seidel methods. The tests were carried out for some matrices generated by the authors. In the following subsection we show the test matrices.

#### A. The studied matrices

##### Matrix 1

$$\begin{bmatrix} 2 & -1 & 0 \\ 1 & 6 & -2 \\ 4 & -3 & 8 \end{bmatrix}$$

In order to examine the condition number of this matrix, we examine the system:

$$\begin{bmatrix} 2 & -1 & 0 \\ 1 & 6 & -2 \\ 4 & -3 & 8 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \\ 5 \end{bmatrix}.$$

##### Matrix 2

The diagonally dominant matrix is generated by a computer program which random numbers from the range  $\langle 0; 1 \rangle$  assigns to the system’s coefficients. Additionally, to make the matrix diagonally dominant, the elements on diagonal are equal to the sum of the elements in the given row, increased by 30. The size of the matrix is  $50 \times 50$ .

##### Matrix 3

$$\begin{bmatrix} 4 & -1 & 3 \\ -7 & 8 & 1.5 \\ -8 & 2 & -6 \end{bmatrix}$$

We try to solve the system:

$$\begin{bmatrix} 4 & -1 & 3 \\ -7 & 8 & 1.5 \\ -8 & 2 & -6 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7.5 \\ -4 \end{bmatrix}.$$

TABLE I

THE MATRICES’ CONDITION NUMBER VALUES CALCULATED WITH THE USE OF TWO DIFFERENT NORMS

#	singular?	$\kappa_1$	$\kappa_\infty$
1	No	8.50E+00	7.80E+00
2	No	2.30E+00	2.26E+00
3	Yes	—	—
4	No	2.84E+04	2.84E+04
5	No	6.11E+19	7.06E+19
6	No	9.00E+00	9.00E+00

This matrix is a singular matrix, so this system does not have a normal solution.

##### Matrix 4

$$\begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{bmatrix}$$

Matrix 4 is a  $4 \times 4$  Hilbert matrix. Its elements are calculated like this:

$$h_{ij} = \frac{1}{i + j - 1}.$$

##### Matrix 5

Matrix 5 is a  $50 \times 50$  Hilbert matrix.

##### Matrix 6

$$\begin{bmatrix} 10^{-18} & 1 \\ 1 & 2 \end{bmatrix}$$

Here we solve the system (1):

$$\begin{bmatrix} 10^{-18} & 1 \\ 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix}.$$

#### B. The condition number study

Table I presents the researched matrices’ condition number values calculated with the use of two norms.

We can see that there is some correlation between the kind of the matrix and its condition number’s value. Among the researched ones, the only matrix with the determinant equal to zero is Matrix 3, and that is why this linear system does not have any solution, and we cannot compute its condition number.

We should notice considerably large condition numbers of the Hilbert matrices (Matrices 4 and 5). They are much bigger than the other ones what suggests a hypothesis that for Hilbert matrices iterative methods are not convergent. On the other hand, system 2 is certainly convergent, whose condition number in any norm does not exceed 2.5, while the condition number of the Hilbert matrix with the size  $4 \times 4$  is as large as ten thousand.

Quite interesting are solutions given for Matrices 1 and 6. These matrices have very small sizes and their condition numbers in the same norms are very similar. However, Matrix 1 is convergent, and Matrix 6 is divergent in terms of iterative methods. In order to explain this situation more precisely, we carried out next tests, results of which are given in the next part of this article.

TABLE II

THE RELATIONSHIP OF THE NUMERICAL METHODS' ACCURACY WITH THE CONDITION NUMBER OF MATRICES

#	Jacobi method		Gauss-Seidel method		$\approx_1$
	10 iter.	100 iter.	10 iter.	100 iter.	
1	5.264E-03	0.00E+00	1.47E-05	0.00E+00	8.50E+00
2	1.93E-02	1.00E-17	4.15E-09	8.24E-18	2.30E+00
3	—	—	—	—	—
4	4.29E+05	5.12E+42	7.67E+00	1.23E+00	2.84E+04
5	1.67E+19	8.71E+165	4.34E+02	4.34E+02	6.11E+19
6	1.28E+84	3.64E+835	9.67E+166	7.81E+1669	9.00E+00

C. The iterative methods' convergence study

We can easily notice the relationship between the condition number of matrices and the iterative methods' convergence when we analyze the accuracy with which the obtained solution approaches a correct solution of the system. Table II shows this well.

We know that the smaller the accuracy approximation value is, the better the obtained result approaches the correct solution's value. "The accuracy equal to zero" means that a given approximation is the correct solution of the equation system. Comparing obtained information, we can notice that for Matrices 1 and 2, the accuracy's values are very small and they decrease in the successive iterations, going to zero, meaning the convergence of the methods.

It is different in the case of Matrices 4, 5 and 6 where the error increases very fast with every iteration. It means divergence.

For the Hilbert matrices (4 and 5) and Matrix 2, the relationship between a condition number of the matrix and the accuracy, or the iterative methods' convergence is very visible. This relationship is not clear in the case of Matrices 1 and 6. The condition numbers of these matrices are very similar, but it is clearly visible that for Matrix 1 numerical methods are convergent and from some moment they give a correct solution of the linear equation system while Matrix 6 acquires immense errors after some iterations.

Therefore, in order to show the relationship between the condition number of matrices and iterative methods' convergence, we carry out next tests, the solutions of which can be found in the next part of this article. For Matrix 3, we can not set any values, because its determinant is equal to 0 and the calculations are impossible, therefore in the next part of this article singular matrices are not be considered.

D. The condition number of Hilbert matrices and of diagonally dominant matrices study

In order to better present the relationship of the condition number and the kind of the matrix, we made a more detailed examination, the results of which can be found in Tables III and IV. We made tests for Hilbert matrices and for matrices generated in an analogous way to Matrix 2. The calculations were made in norms of  $l_1$  and  $l_\infty$  for matrices of different sizes.

When we analyze the data in Tables III and IV, we can easily notice that the condition number of Hilbert matrices

TABLE III

THE CONDITION NUMBER VALUES OF HILBERT MATRICES AND OF DIAGONALLY DOMINANT MATRICES IN THE NORM OF  $l_1$

size of matrix	$\approx_1$	
	Hilbert matrix	matrix of type 2
1	1.00E+00	1.00E+00
2	2.70E+01	1.04E+00
3	7.48E+02	1.10E+00
4	2.84E+04	1.15E+00
5	9.44E+05	1.19E+00
6	2.91E+07	1.22E+00
7	9.85E+08	1.31E+00
8	3.39E+10	1.31E+00
9	1.10E+12	1.35E+00
10	3.54E+13	1.40E+00
15	2.81E+21	1.62E+00
20	1.30E+22	1.69E+00
25	5.88E+21	1.86E+00
30	5.71E+22	1.92E+00
35	1.73E+23	2.00E+00
40	1.06E+23	2.09E+00
45	1.95E+23	2.27E+00
50	1.35E+23	2.36E+00

TABLE IV

THE CONDITION NUMBER VALUES OF HILBERT MATRICES AND OF DIAGONALLY DOMINANT MATRICES IN THE NORM OF  $l_\infty$

size of matrix	$\approx_\infty$	
	Hilbert matrix	matrix of type 2
1	1.00E+00	1.00E+00
2	2.70E+01	1.04E+00
3	7.48E+02	1.09E+00
4	2.84E+04	1.13E+00
5	9.44E+05	1.19E+00
6	2.91E+07	1.24E+00
7	9.85E+08	1.28E+00
8	3.39E+10	1.29E+00
9	1.10E+12	1.30E+00
10	3.54E+13	1.35E+00
15	2.81E+21	1.66E+00
20	1.27E+22	1.66E+00
25	5.84E+21	1.79E+00
30	5.92E+22	1.89E+00
35	1.21E+23	2.00E+00
40	1.22E+23	2.06E+00
45	1.27E+23	2.09E+00
50	1.03E+23	2.26E+00

is much bigger than the condition number of diagonally dominant matrices (Fig. 1).

However, a condition number's value does not depend only on the kind of the matrix but also on the size of the system that is examined. This relationship is direct, but for a randomly generated diagonally dominant matrices the condition number increases very slowly, while for Hilbert matrices it grows very rapidly. As early as for the size  $8 \times 8$ , the condition number of the Hilbert matrix reaches value  $3.39E+10$ , while for the diagonally dominant matrix of the same size, the condition number is only 1.3121 in the norm of  $l_1$  and 1.3166 in the norm of  $l_\infty$ . We can see that for Hilbert matrices, beginning from the size  $25 \times 15$ , the condition number, though huge, does not grow so fast but rather oscillates around some very high value.

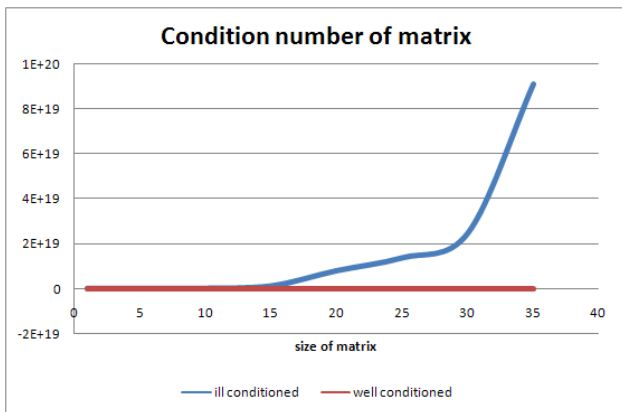


Fig. 1. The graph of the condition number of Hilbert matrices and diagonally dominant matrices

TABLE V  
THE RELATIONSHIP OF THE ITERATIVE METHODS' ACCURACY AND HILBERT MATRICES' CONDITION NUMBER

size of matrix	number of iterations	Jacobi method	Gauss-Seidel method	$\approx_1$
5	10	7.61E+06	8.16E+00	9.442E+05
	25	8.66E+14	3.02E+00	
	50	2.32E+28	1.98E+00	
	75	6.19E+41	2.02E+00	
	100	1.65E+55	1.93E+00	
10	10	6.94E+10	2.52E+01	3.54E+13
	25	1.61E+24	1.99E+01	
	50	3.02E+46	1.40E+01	
	75	5.68E+68	9.54E+00	
20	10	3.28E+14	7.69E+01	1.30E+22
	25	5.95E+32	4.66E+01	
	50	1.61E+63	2.63E+01	
	75	4.35E+93	2.65E+01	
40	10	1.22E+18	2.97E+02	1.06E+2
	25	1.12E+41	1.44E+02	
	50	2.08E+79	1.18E+02	
	75	3.88E+117	7.70E+01	
	100	7.22E+155	6.25E+01	

The obtained results also show that the condition number's values calculated in different norms for the same matrices, which are of the same size, are very close to each other. We can say that the kind of the norm has little influence on the value of the condition number of a matrix.

*E. The examination of the iterative methods' convergence for Hilbert matrices and diagonally dominant matrices*

The results of examination presented in the previous subsection show how strongly a condition number depends on the kind of the matrix. The tests of the iterative methods' accuracy given in Tables V and VI show, that there exists very strong relationship between the condition number's value and the iterative methods' convergence.

Iterative methods for this system of equations are divergent what results clearly from calculation's accuracy analysis for the Hilbert matrices (Table V). The Jacobi method just after some iterations gives solutions with a serious error. The Gauss-

TABLE VI  
THE RELATIONSHIP OF THE ITERATIVE METHODS' ACCURACY AND DIAGONALLY DOMINANT MATRICES' CONDITION NUMBER

size of matrix	number of iterations	Jacobi method	Gauss-Seidel method	$\approx_1$
5	5	3.13E-05	3.37E-08	1.13E+00
	10	4.66E-11	8.67E-19	
	15	6.91E-17	0.00E+00	
	25	0.00E+00	0.00E+00	
	50	0.00E+00	0.00E+00	
10	5	1.79E-03	1.32E-06	1.32E+00
	10	9.95E-08	8.46E-15	
	15	5.53E-12	2.49E-18	
	25	1.62E-18	2.49E-18	
	50	1.62E-18	2.49E-18	
20	5	2.95E-02	1.70E-05	1.76E+00
	10	2.04E-05	2.68E-12	
	15	1.41E-08	4.13E-18	
	25	6.69E-15	4.13E-18	
	50	5.25E-18	4.13E-18	
40	5	4.98E-01	4.02E-04	2.20E+00
	10	4.84E-03	1.89E-09	
	15	4.70E-05	8.37E-15	
	25	4.44E-09	6.02E-18	
	50	5.30E-18	6.02E-18	

Seidel method can create a impression of the convergence because he accuracy's value does not grow as visibly as in the Jacobi method. It can be even supposed that the solutions are better and better approximated. However, after more detailed analysis, we can see that after a lot of iterations, the accuracy's value oscillates around the number range of tenth or units. This number is definitely too big to recognize the obtained information as the correct solution's approximation.

We can see that a big condition number of a matrix can show the iterative methods' divergence. Additionally, because the size of the system has an influence on the condition number's value, the bigger the size of a matrix is, the faster the iterative methods' divergence is.

The research of the iterative methods' accuracy for randomly generated diagonally dominant matrices shows that these methods are convergent (Table VI). These methods, after a few iterations, give the correct solution or a solution of the equation system with a minute error, which is after some iterations very close to zero.

The analysis of the obtained information clearly shows that the Gauss-Seidel method approaches the correct solution much faster than the Jacobi method. The convergence speed of the iterative methods depends also on the size of a matrix (the bigger the matrix, the slower the method converges), and also on the condition number of a matrix. This relationship is inverse because the smaller the condition number is, the faster the iterative methods approach the correct linear equation system's solution.

The data in Table V and VI shows that Hilbert matrices, the condition number of which is very high, are ill-conditioned, so the iterative methods are not convergent for them. On the contrary, diagonally dominant matrices have small condition numbers, which means that they are well-conditioned matrices for which the iterative methods are convergent.

TABLE VII

THE CONDITION NUMBERS IN NORM OF  $l_1$  OF SMALL HILBERT MATRICES AND SMALL DIAGONALLY DOMINANT MATRICES

size of matrix	Hilbert matrix	Matrix 2a	Matrix 2b
1	1.00E+00	1.00E+00	1.00E+00
2	2.70E+01	1.53E+00	1.04E+00
3	7.48E+02	1.92E+00	1.10E+00
4	2.84E+04	2.74E+00	1.15E+00
5	9.44E+05	3.08E+00	1.19E+00
6	2.91E+07	4.24E+00	1.22E+00
7	9.85E+08	4.45E+00	1.31E+00
8	3.39E+10	4.79E+00	1.31E+00
9	1.10E+12	4.92E+00	1.35E+00

### F. The examination of condition numbers of small matrices

The previous tests do not explain why the condition numbers of Matrices 1 and 6 are so similar although the first matrix is well-conditioned and the second one is ill-conditioned. In order to explain this problem, we have to analyze the results in Table VII.

The first matrix is a Hilbert matrix. Matrices 2a and 2b are generated in the same way as Matrix 2. Matrix 2a differs from matrix 2b in such a way that its elements are random numbers from the range  $(0; 10)$  — not from the range  $(0; 1)$ . Such a generation of matrices causes that elements of the Hilbert matrices and of Matrices 2b are much smaller than the elements of Matrices 2a. We can see from Table VII, that the condition numbers of Matrices 2a are much bigger than the condition numbers of Matrices 2b, and nevertheless very small in relation to the Hilbert matrices. The accuracy tests' results of the Matrices 2a presented in the Table VIII show that iterative methods for this matrix are convergent, and therefore the matrix is well-conditioned.

Comparing the information from Tables VI and VIII, we can see that the iterative methods for a matrix of bigger elements (and thereby of a bigger condition number) converge slower than for a matrix of a small condition number, and also approach the correct solution of the linear equation system faster.

The condition number of a matrix do not always define whether the matrix is well-conditioned or ill-conditioned. That happens when the matrix is of a small size. For every system whose size is equal to 1, a condition number of the matrix is equal to 1, and the iterative methods give the correct solution just after the first iteration. Analyzing Tables VII and IX, we can notice that for a matrix of a small size (2 or 3) but of big elements' values the condition number is similar and we cannot decide on its basis if the iterative method for this matrix is convergent or divergent. This fact is evidently confirmed by the results shown in Table X. It is clear that Matrix 1 is well-conditioned and Matrix 6 is not convergent in the iteration process although the condition number of these matrices calculated in the norm of  $l_1$  differs very little from each other.

A problem with recognizing, with the use of a condition number, if an iteration process at a given matrix is convergent or not, refers only to a matrix of a very small size because for

TABLE VIII

THE RELATIONSHIP OF THE ITERATIVE METHODS' ACCURACY AND THE CONDITION NUMBER OF DIAGONALLY DOMINANT MATRICES WITH BIG ELEMENTS (2A)

size of matrix	number of iterations	Jacobi method	Gauss-Seidel method	$\kappa_1$
3	5	4.64E-02	6.65E-06	1.92E+00
	10	1.45E-04	1.63E-12	
	15	4.58E-07	1.79E-18	
	25	4.57E-12	1.73E-18	
5	50	1.73E-18	1.73E-18	3.08E+00
	5	1.73E+00	1.06E-02	
	10	1.41E-01	2.73E-06	
	15	1.15E-02	1.11E-09	
7	25	7.62E-05	9.44E-17	4.45E+00
	50	2.73E-10	1.23E-18	
	5	1.46E+01	1.80E-02	
	10	6.35E+00	8.32E-06	
9	15	2.77E+00	3.66E-09	4.92E+00
	25	5.27E-01	5.71E-16	
	50	8.30E-03	1.11E-18	
	5	1.76E+01	9.60E-03	
9	10	1.29E+01	3.94E-06	4.92E+00
	15	9.50E+00	1.86E-09	
	25	5.12E+00	2.75E-16	
	50	1.09E+00	1.06E-18	

TABLE IX

THE VALUES OF THE CONDITION NUMBER OF THE MATRICES WHICH HAVE SMALL SIZES, CALCULATED WITH THE USE OF TWO DIFFERENT NORMS

#	size	$\kappa_1$	$\kappa_\infty$
1	3	8.50E+00	7.80E+00
6	2	9.00E+00	9.00E+00

TABLE X

THE RELATIONSHIP OF THE ITERATIVE METHODS ACCURACY AND THE CONDITION NUMBER OF THE MATRICES WHICH HAVE SMALL SIZES

#	Jacobi method		Gauss-Seidel method		$\kappa_1$
	10 iter.	50 iter.	10 iter.	50 iter.	
1	5.26E-03	0.00E+00	1.47E-05	0.00E+00	8.50E+00
6	1.28E+84	1.22E+443	9.67E+166	8.80E+884	9.00E+00

ill-conditioned matrices of the size  $4 \times 4$  the condition number is incomparably bigger than a condition of a well-conditioned matrix of 4th degree, which has big elements' values. This happens, because even for big elements a condition number of a well-conditioned matrix grows incomparably slower than a condition of an ill-conditioned matrix, which is confirmed by the information in Table VI.

## IV. CONCLUSIONS

These experiments show that there exists a strong relationship between the size of a system and a condition number of a matrix as well as between the condition number's value and the iterative methods' convergence.

Iterative methods are convergent for a matrix of a small condition number (well-conditioned matrices) and divergent for matrices which have a big condition number (ill-conditioned matrices). The condition number of a matrix is directly connected to its size, but for well-conditioned matrices it grows incomparably slower than for ill-conditioned matrices. Fig. 1 visibly shows the difference between the condition number of

a well-condition matrix and the condition number of an ill-conditioned matrix.

The size of a system also indirectly influences the iterative process's convergence. In the case of well-conditioned matrices, the bigger size of a matrix is, the slower the methods approach the correct solution's value. On the contrary, for ill-conditioned matrices, the bigger size of a matrix is (a bigger condition number thereby), the faster the iterative process diverges.

For well-conditioned matrices the condition number can be influenced by the size of elements of a matrix, although the condition number is incomparably smaller than the condition number of ill-conditioned matrices.

It is also easily seen (Fig. 1), that a value of a condition number is growing with the growth of the size of the matrix.

So we can see, that the condition number of a well-conditioned matrix is always small and that the big condition number proves the matrix to be ill-conditioned. However, we

can find some ill-conditioned matrices with small condition numbers.

#### REFERENCES

- [1] M. Arioli, F. Romani, Relations between condition numbers and the condition convergence of the Jacobi method for real positive definite matrices, *Numerische Mathematik* 46, 31 - 42 (1985)
- [2] B. Beckermann, The condition number of real Vandermonde, Krylov and positive definite Hankel matrices, *Numerische Mathematik*. 85(4), 553-577, 2000.
- [3] E. Brakkee, P. Wilders, The influence of interface conditions on convergence of Krylov-Schwarz domain decomposition for the advection-diffusion equation, *Journal of Scientific Computing*, Vol 12, No 1, 11 - 30(20), March 1997
- [4] G. Dahlquist, A. Björck, *Numerical methods*, Englewood Cliffs, Prentice-Hall, 1974
- [5] D. Kincaid, W. Cheney, *Numerical Analysis. Mathematics of Scientific Computing*, Third Edition, 2008
- [6] J. Stoer, R. Bulirsch, *Introduction to numerical analysis*, New York, Springer 1980
- [7] T. Zolezzi, Condition numbers and Ritz type methods in unconstrained optimization, *Control and Cybernetics*, vol 36, No.3 (2007)

# Solving Linear Recurrences on Hybrid GPU Accelerated Manycore Systems

Przemysław Stpicznyński

Institute of Mathematics, Maria Curie-Skłodowska University, Lublin, Poland

Institute of Theoretical and Applied Informatics of the Polish Academy of Sciences, Gliwice, Poland

Email: przem@hektor.umcs.lublin.pl

**Abstract**—The aim of this paper is to show that linear recurrence systems with constant coefficients can be efficiently solved on hybrid GPU accelerated manycore systems with modern Fermi GPU cards. The main idea is to use the recently developed *divide-and-conquer* algorithm which can be expressed in terms of Level 2 and 3 BLAS operations. The results of experiments performed on hybrid system with Intel Core i7 and NVIDIA Tesla C2050 are also presented and discussed.

## I. INTRODUCTION

GRAPHICAL processing units (GPUs [10]) have recently been widely used for scientific computing due to their large number of parallel processors which can be exploited using the Compute Unified Device Architecture (CUDA) programming language [9]. GPUs offer very high performance at low costs for data-parallel computational tasks. Thus it is a good idea to develop algorithms for hybrid (heterogeneous) computer architectures where large parallelizable tasks are scheduled for execution on GPUs, while small non-parallelizable tasks should be run on CPUs [25]. In 2010 NVIDIA introduced a new GPU (CUDA) architecture with an internal name of Fermi [14]. The new architecture offers new features and much better performance than older CUDA devices when computations are carried out in double precision.

Linear recurrence systems with constant coefficients are central parts of many numerical algorithms for solving important problems like evaluation of orthogonal polynomials [2], symmetric tridiagonal eigenvalue problem [17], trigonometric interpolation [17], [18], numerical inverse of the Laplace Transform [8], [22] or general polynomial evaluation [19]. Also, they are commonly used in signal processing [16], [24]. There are several algorithms for solving linear recurrences on parallel computers [2], [3], [5], [6], [15], [26]. They are usually devoted to the problem of solving first or second order recurrence systems. In [21] we have proposed a new *divide-and-conquer* approach for solving  $m$ -th order linear recurrences with constant coefficients which can fully utilize the underlying hardware of modern computer systems (i.e. memory hierarchies and multiple processors). Our algorithm based on this approach achieves excellent speedup on various parallel computers [20], [23], [24].

It is clear that there is a real need to implement efficient solvers for linear recurrences and related problems on modern GPU accelerated manycore systems. The first attempt has been made by Nistor et al. [11]. They present a compile-time

optimization technique for minimizing memory usage in the parallel computation of linear recurrences on GPUs. The aim of this paper is to show that the use of recently developed *divide-and-conquer* algorithm which can be expressed in terms of Level 2 and 3 BLAS operations [1] is crucial for achieving reasonable performance for linear recurrence computations on GPU-based systems.

## II. THE *Divide & Conquer* METHOD

Let us consider the following problem. For given real numbers  $f_k$ ,  $1 \leq k \leq n$ , and  $a_j$ ,  $1 \leq j \leq m$ , where  $n \gg m$ , find  $n$  numbers  $x_k$ ,  $1 \leq k \leq n$ , such that

$$x_k = \begin{cases} 0 & \text{for } k \leq 0 \\ f_k + \sum_{j=1}^m a_j x_{k-j} & \text{for } 1 \leq k \leq n. \end{cases} \quad (1)$$

It is clear that a simple algorithm based on (1), which requires  $2mn$  flops, cannot utilize the underlying hardware, i.e. memory hierarchies, vector extensions and multiple processors, what is essential in case of modern manycore computer architectures. Thus, let us consider the following *divide-and-conquer* approach which allows to describe the process of solving (1) in terms of matrix operations [21]. First, let us choose positive integers  $r$  and  $s$  such that  $rs \leq n$  and  $s > m$ . It is clear that the numbers  $x_1, \dots, x_{rs}$  satisfy the following block system of linear equations

$$\begin{bmatrix} L & & & & \\ U & L & & & \\ & & \ddots & & \\ & & & U & L \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_r \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \vdots \\ \mathbf{f}_r \end{bmatrix}, \quad (2)$$

where the vectors  $\mathbf{x}_j$  and  $\mathbf{z}_j$  are given by

$$\mathbf{x}_j = [x_{(j-1)s+1}, \dots, x_{js}]^T \in \mathbb{R}^s, \quad (3)$$

and

$$\mathbf{f}_j = [f_{(j-1)s+1}, \dots, f_{js}]^T \in \mathbb{R}^s. \quad (4)$$





computers and clusters of workstations [20], [23] achieving reasonable performance. The performance of the algorithm (in Gflops) grows when the problem sizes ( $n$ ,  $m$ ) and the number of processors grow.

---

**Algorithm 2** Parallel BLAS-based algorithm for (1)

---

**Require:** given  $a_1, \dots, a_m$  and  $\mathbf{f}_1, \dots, \mathbf{f}_r$  given by (4), where  $r > 1$ ,  $s > 1$ ,  $rs = n$  and  $p$  is the number of parallel processors

**Ensure:**  $X_{1:s,1:r} = (\mathbf{x}_1, \dots, \mathbf{x}_r)$ .

```

1:  $X \leftarrow (\mathbf{f}_1, \dots, \mathbf{f}_r, \mathbf{e}_1)$ 
2:  $C \leftarrow \begin{pmatrix} a_m & \cdots & a_1 \\ & \ddots & \vdots \\ 0 & & a_m \end{pmatrix}$ 
3:  $t \leftarrow \lfloor (r+1)/p \rfloor$ 
4: parallel for  $i = 0$  to  $p-1$  do
5:    $t_1 \leftarrow it + 1$ ;  $t_2 \leftarrow (i+1)t$ 
6:   if  $i = p-1$  then
7:      $t_2 \leftarrow r+1$ 
8:   end if
9:   for  $k = 2$  to  $s$  do
10:     $X_{k,t_1:t_2} \leftarrow X_{k,t_1:t_2} + C_{1,\max\{1,m-k+2\}:m} X_{\max\{1,k-m\}:k-1,t_1:t_2}$  {xGEMV}
11:   end for
12: end parallel for {  $X_{*,r+1} = \mathbf{y}_1$  }
13:  $Y \leftarrow (\mathbf{y}_1, \dots, \mathbf{y}_m)$ 
14: for  $j = 2$  to  $r$  do
15:    $A_{*,j-1} \leftarrow CX_{s-m+1:s,j-1}$  {xTRMV}
16:    $X_{s-m+1:s,j} \leftarrow X_{s-m+1:s,j} + Y_{s-m+1:s,*} A_{*,j-1}$  {xGEMV}
17: end for
18:  $t \leftarrow \lfloor r/p \rfloor$ 
19: parallel for  $i = 0$  to  $p-1$  do
20:    $t_1 \leftarrow it + 1$ ;  $t_2 \leftarrow (i+1)t$ 
21:   if  $i = 0$  then
22:      $t_1 \leftarrow t_1 + 1$ 
23:   end if
24:   if  $i = p-1$  then
25:      $t_2 \leftarrow r$ 
26:   end if
27:    $X_{1:s-m,t_1:t_2} \leftarrow X_{1:s-m,t_1:t_2} + Y_{1:s-m,*} A$  {xGEMM}
28: end parallel for

```

---

### III. GPU ARCHITECTURE AND IMPLEMENTATION

The detailed description of NVIDIA CUDA and Fermi architectures can be found in [14] and [13]. A *computing device* (usually GPU) comprises a number of streaming multi-processors (SM). Each SM consists of 32 (8 for older CUDA devices) scalar cores (streaming processors, SP). SMs are responsible for executing blocks of threads. Threads within a block are grouped into so-called *warps*, each consisting of 32 threads managed and executed together.

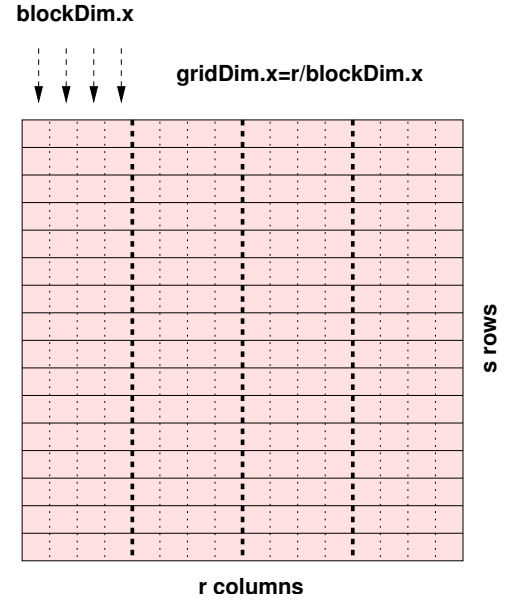


Fig. 1. Storage for the *divide and conquer* algorithm on GPUs

A device has its own memory system including the *global memory* (large but low latency), *constant* and *texture* read-only memories providing reduction of memory latency. Each SM has also a 48 kB (16 kB for older CUDA devices) of fast *shared memory* that can be used for sharing data among threads within a block. The global memory access can be improved by coalesced access of all threads in a half-warp. Threads must access either 4-byte words in one 64-byte memory transaction, or 8-byte words in one 128-byte memory transaction. All 16 words must lie in the same memory segment [13].

Fermi (CUDA) programs consist of a number of C functions called *kernels* that are to be executed on devices as threads. Kernels are called from programs executed on CPUs. Host programs are also responsible for allocation of variables in the device global memory. There are also some CUDA API-functions used to copy data between host and device global memories.

The basic idea of the implementation of the *divide-and-conquer* algorithm on a hybrid CPU + GPU system is to run large parallel tasks (**Step1** and **Step3**) on GPU, while small sequential tasks (namely finding the vector  $\mathbf{y}_1$  and **Step2**) are executed on CPU. To allow coalesced memory access, elements of the  $s \times r$  array  $X$  should be stored row-wise in the global memory of a device. Thus we cannot use CUBLAS (i.e. the implementation of the BLAS for CUDA [12]) because routines from that library assume column-wise storage of matrices. Each thread is responsible for computing one column of the array. The ID of its own column can be computed using  $\text{idx} = \text{blockIdx.x} * \text{blockDim.x} + \text{threadIdx.x}$ . Each block of threads is responsible for computing one *panel* – a group of adjacent columns (Figure 1). For simplicity, we assume that  $r = (\#blocks) \times (\#threads \text{ in block})$ .

The numbers  $a_j$ ,  $1 \leq j \leq m$ , are stored in the constant

```

__global__ void step1(int m, int s, int r,
                    double *x){
    __shared__ double xc[16][MAXBSIZE];
    int idx=blockDim.x*blockIdx.x+
            threadIdx.x;

    int myIdx=threadIdx.x;

    xc[0][myIdx]=x[idx];
    double t;

    // rows 0..m-1
    for(int i=1; i<m; i++){
        t=x[i*r+idx]; // read from global mem.
        for(int j=1; j<=i; j++){
            t+=xc[i-j][myIdx]*a_d[j-1];
            x[i*r+idx]=t; // write from global mem.
            xc[i][myIdx]=t;
        }

        // rows m..s-1
        for(int i=m; i<s; i++){
            t=x[i*r+idx];
            for(int k=1; k<=m; k++){
                t+=xc[m-k][myIdx]*a_d[k-1];
            }
            for(int k=0; k<m-1; k++){
                xc[k][myIdx]=xc[k+1][myIdx];
            }
            xc[m-1][myIdx]=t;
            x[i*r+idx]=t; // write to global mem.
        }
    }
}

```

Fig. 2. Kernel for **Step1**

memory of GPU. The algorithm starts with the kernel `step1` (Figure 2) on GPU. To reduce the number of references to the global memory, we use shared memory to store  $m$  recently computed entries of the thread's column. At the same time CPU solves the system  $Ly_1 = e_1$ . Next,  $m$  last rows of the array  $X$  are copied from the device memory and CPU computes (12). Then the vector  $y_1$  and  $m$  last rows of  $X$  are copied to the device memory. Finally, the kernel `step3` (Figure 3) is scheduled for execution on GPU.

#### IV. RESULTS OF EXPERIMENTS

The performance results in this section use the hybrid computer system with Intel Core i7 950 Extreme Edition(3.06 GHz, 24GB RAM) and NVIDIA Tesla C2050 (448 cores, 3GB GDDR5 RAM), running under Linux with Intel `icc` (version 12.0) and NVIDIA `nvcc` (release 3.2) compilers and Intel MKL Library (version 10.3). We have tested the following algorithms:

- the simple sequential algorithm based on (1) executed on CPU,
- Algorithm 1 executed on CPU,

```

__global__ void step3(int m, int s, int r,
                    double *x, double *y){
    __shared__ double xc[16][MAXBSIZE];
    int idx=blockDim.x*blockIdx.x+
            threadIdx.x;

    int myIdx=threadIdx.x;

    for(int k=0; k<m; k++){
        xc[k][myIdx]=x[(s-m+k)*r+idx];
    }

    if(idx==0){
        for(int k=0; k<m; k++){
            xc[k][0]=0;
        }
        y=y+m-1;
    }

    for(int i=0; i<s-m; i++){
        double t=0.0;
        for(int k=m-1; k>=0; k--){
            t+=xc[k][myIdx]*y[i-k];
            x[i*r+idx]+=t;
        }
    }
}

```

Fig. 3. Kernel for **Step3**

- the BLAS-based *divide-and-conquer* algorithm executed on GPU+CPU.

The performance of the simple sequential algorithm is very poor (up to 660 Mflops), thus the exemplary results shown in Figure 5 and 6 describe the performance of two considered implementations of the BLAS-based algorithm. They have been tested for various values of  $n$  from  $n = 1024^2 = 1048576$  to  $n = 16384^2 = 268435456$ ,  $m = 1, 2, 4, 6, 8, 16$  and various values of  $r$  and  $s$ , where  $rs = n$ . It is clear that greater values of the parameter  $r$  let us keep a GPU busy, but in such a case the sequential part of the algorithm (namely **Step2**) is longer, thus the GPU+CPU implementation achieves the best performance for  $r = s$ . The CPU implementation achieves the best performance for  $s = 4r$ . The performance of the GPU+CPU implementation grows when both  $m$  and  $n$  grow, while the performance of the CPU implementation decreases when  $n$  grows. The use of GPU+CPU is profitable for  $n \geq 4194304 = 2048^2$ . The GPU+CPU implementation is up to 58 (single precision) and 47 (double precision) times faster than CPU for  $m = 1$ . For greater values of  $m$ , GPU+CPU also outperforms CPU. One can observe that the speedup decreases with growing of  $m$ , because on CPU, for greater values of  $m$ , the BLAS routines `xGEMV` and `xGEMM` are much more efficient. Finally it should be noticed that our GPU+CPU implementation achieves much better speedup than the algorithm presented in [11].

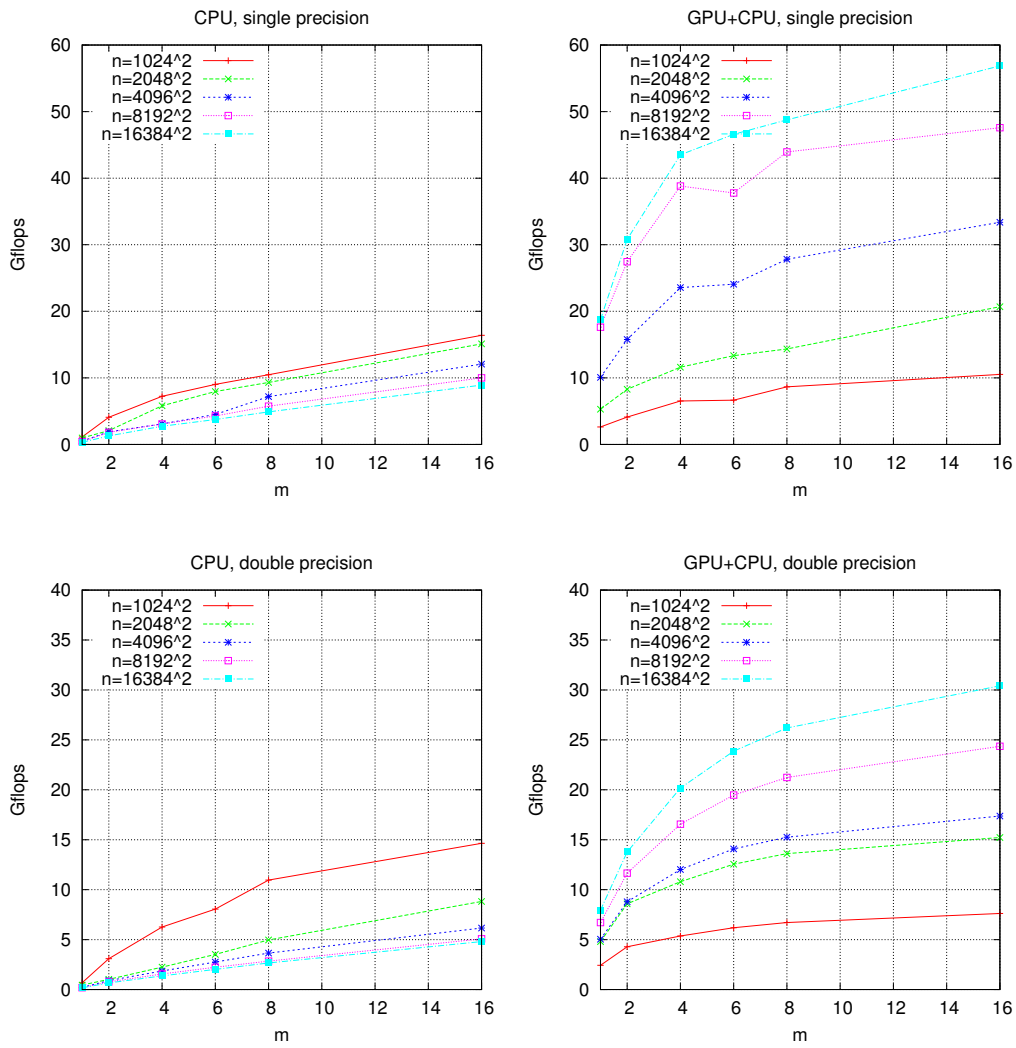


Fig. 5. Performance of the BLAS-based CPU implementation and the hybrid GPU+CPU implementation for single and double precision

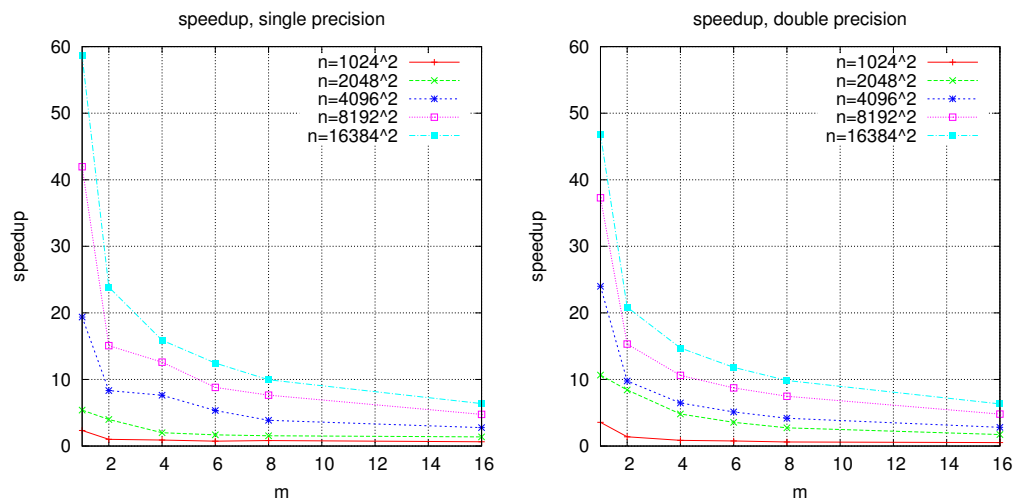


Fig. 6. Speedup of the hybrid GPU+CPU implementation over the BLAS-based CPU implementation

```

int bsize=256;

step1 <<<r/bsize , bsize >>>(m, s, r, x_d);

recl y(m, s, a_h, y_h+m-1);
for (int k=0; k<m-1; k++){
    y_h[k]=0;
}

cudaMemcpy(t_h, &x_d[(s-m)*r],
           r*m*sizeof(*t_h),
           cudaMemcpyDeviceToHost);

for (int j=1; j<r; j++){ // Step 2
    for (int i=0; i<m; i++){
        int ig=s-m+i;
        for (int k=0; k<m-i; k++){
            x_h[ig*r+j]+=
            a_h[m-k]*x_h[(s-m+k)*r+j-1];
        }
    }

    cudaMemcpy(&x_d[(s-m)*r],
              t_h, s*sizeof(*t_h),
              cudaMemcpyHostToDevice);
    cudaMemcpy(y_d, y_h,
              (m-1+s)*sizeof(*y_d),
              cudaMemcpyHostToDevice);
step3 <<<r/bsize , bsize >>>(m, s, r, x_d, y_d);

cudaThreadSynchronize();

```

Fig. 4. Host program

## V. CONCLUSIONS

We have shown that linear recurrence systems with constant coefficients can be efficiently solved on hybrid GPU accelerated manycore systems with modern Fermi GPU cards using the *divide and conquer* BLAS-based algorithm. The reasonable performance can be achieved by using some optimization techniques for GPUs like coalesced global memory access and the use of shared memory for caching elements of global memory arrays during computations. It should be noticed the algorithm can be easily implemented for CPU + multiple GPU systems using OpenCL [7].

## ACKNOWLEDGEMENTS

The author would like to thank the anonymous referees for valuable discussions and suggestions.

## REFERENCES

- [1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostruchov, and D. Sorensen, *LAPACK User's Guide*. Philadelphia: SIAM, 1992.
- [2] R. Bario, B. Melendo, and S. Serrano, "On the numerical evaluation of linear recurrences," *J. Comput. Appl. Math.*, vol. 150, pp. 71–86, 2003.
- [3] G. Blelloch, S. Chatterjee, and M. Zgha, "Solving linear recurrences with loop raking," *Journal of Parallel and Distributed Computing*, vol. 25, pp. 91–97, 1995.
- [4] R. Chandra, L. Dagum, D. Kohr, D. Maydan, J. McDonald, and R. Menon, *Parallel Programming in OpenMP*. San Francisco: Morgan Kaufmann Publishers, 2001.
- [5] H. Hafner and W. Shonauer, "Investigation of different algorithms for the first order recurrence," *Supercomputer*, vol. 40, pp. 34–41, 1990.
- [6] J.-L. Larriba-Pey, J. J. Navarro, A. Jorba, and O. Roig, "Review of general and Toeplitz vector bidiagonal solvers," *Parallel Comput.*, vol. 22, no. 8, pp. 1091–1125, 1996.
- [7] A. Munshi, *The OpenCL Specification v. 1.0*. Khronos OpenCL Working Group, 2009.
- [8] A. Murli and M. Rizzardi, "Algorithm 682: Talbot's method for the Laplace inversion problem," *ACM Trans. Math. Soft.*, vol. 16, pp. 158–168, 1990.
- [9] J. Nickolls, I. Buck, M. Garland, and K. Skadron, "Scalable parallel programming with CUDA," *ACM Queue*, vol. 6, pp. 40–53, 2008.
- [10] J. Nickolls and W. J. Dally, "The gpu computing era," *IEEE Micro*, vol. 30, pp. 56–69, 2010.
- [11] A. Nistor, W.-N. Chin, T.-S. Tan, and N. Tapus, "Optimizing the parallel computation of linear recurrences using compact matrix representations," *J. Parallel Distrib. Comput.*, vol. 69, pp. 373–381, 2009.
- [12] NVIDIA Corporation, *CUBLAS Library*. NVIDIA Corporation, 2009, available at <http://www.nvidia.com/>.
- [13] —, *CUDA Programming Guide*. NVIDIA Corporation, 2009, available at <http://www.nvidia.com/>.
- [14] —, "NVIDIA next generation CUDA compute architecture: Fermi," <http://www.nvidia.com/>, 2009.
- [15] K. Shimizu and Y. Kanada, "Solving linear recurrence problems on supercomputers," *Supercomputer*, vol. 8, no. 1, pp. 30–37, Jan. 1991.
- [16] S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. San Diego, CA: California Technical Publishing, 1997.
- [17] J. Stoer and R. Bulirsh, *Introduction to Numerical Analysis*, 2nd ed. New York: Springer, 1993.
- [18] P. Stpiczyński, "Fast parallel algorithms for computing trigonometric sums," in *Proceedings of PARELEC 2002, International Conference on Parallel Computing in Electrical Engineering*, M. Tudruj and A. Jordan, Eds. IEEE Computer Society Press, 2002, pp. 299–304. [Online]. Available: <http://computer.org/proceedings/parelec/1730/17300299abs.htm>
- [19] —, "Fast parallel algorithm for polynomial evaluation," *Parallel Algorithms and Applications*, vol. 18, no. 4, pp. 209–216, 2003.
- [20] —, "Numerical evaluation of linear recurrences on various parallel computers," in *Proceedings of Aplimat 2004, 3rd International Conference, Bratislava, Slovakia, February 4–6, 2004*, M. Kovacova, Ed. Technical University of Bratislava, 2004, pp. 889–894.
- [21] —, "Solving linear recurrence systems using level 2 and 3 BLAS routines," *Lecture Notes in Computer Science*, vol. 3019, pp. 1059–1066, 2004.
- [22] —, "A note on the numerical inversion of the laplace transform," *Lecture Notes in Computer Science*, vol. 3911, pp. 551–558, 2006.
- [23] —, "Numerical evaluation of linear recurrences on high performance computers and clusters of workstations," in *Proceedings of PARELEC 2004, International Conference on Parallel Computing in Electrical Engineering*. IEEE Computer Society Press, 2004, pp. 200–205.
- [24] —, "Evaluating recursive filters on distributed memory parallel computers," *Comm. Numer. Meth. Engng.*, vol. 22, pp. 1087–1095, 2006.
- [25] S. Tomov, J. Dongarra, and M. Baboulin, "Towards dense linear algebra for hybrid GPU accelerated manycore systems," *Parallel Computing*, vol. 36, pp. 232–240, 2010.
- [26] H. van der Vorst and K. Dekker, "Vectorization of linear recurrence relations," *SIAM J. Sci. Stat. Comput.*, vol. 10, no. 1, pp. 27–35, 1989.

# A multipoint shooting feasible-SQP method for optimal control of state-constrained parabolic DAE systems

Krystyn Styczeń

Institute of Computer Engineering  
 Wrocław University of Technology  
 Janiszewskiego 11/17, 50-372 Wrocław, Poland  
 Email: krystyn.styczen@pwr.wroc.pl

Wojciech Rafajłowicz

Institute of Computer Engineering  
 Wrocław University of Technology  
 Janiszewskiego 11/17, 50-372 Wrocław, Poland  
 Email: wojciech.rafajlowicz@pwr.wroc.pl

**Abstract**—Optimal control problem for parabolic differential-algebraic equations (PDAE) systems with spatially sensitive state-constraints and technological constraints is considered. Multipoint shooting approach is proposed to attack such problems. It is well suited to deal with unstable and ill-conditioned PDAE systems. This approach consists in the partitioning of the time-space domain into shorter layers, which allows us to fully parallelize the computations and to employ the reliable PDAE solvers. A new modified method of this kind is developed. It converts the multipoint shooting problem having mixed equality and inequality constraints into the purely inequality constrained problem. The results of the consecutive layer shots are exploited to determine a feasible shooting solution of the converted problem. The knowledge of such a solution is crucial for the use of highly efficient feasible-SQP methods avoiding the incompatibility of the constraints of the QP subproblems (versus the infeasible path SQP methods). The applications of the method proposed to the optimization of some heat transfer processes as well as chemical production processes performed in tubular reactors are discussed.

## I. INTRODUCTION

ADVANCED technological processes are often performed in distributed-parameter systems. A wide class of such processes is described by parabolic differential-algebraic equations including heat transfer processes and chemical production processes performed in tubular or multizone reactors [3], [9],[11],[2]. Many of them exhibit unstable modes and high sensitivity to parameter changes. In a consequence we can deal with unbounded state profiles and ill-conditioned optimization problems if the single shooting method is applied to optimize the system performance [2]. The multipoint shooting approach, known from the modelling of lumped parameter systems [10], can also be useful for distributed parameter systems to resolve the difficulties with their unstable and highly sensitive modes [5],[6],[13]. Moreover, it ensures the full parallelization of the computations necessary to employ advanced superlinearly convergent optimization methods such as the sequential quadratic programming (SQP) method or the interior point (IP) method [1]. However, the main elaborations on this subject deal with infeasible path approach, where all the constraints of the optimization problem may be violated on the current iteration.

This may lead to the incompatibility of the constraints of the QP subproblems, which complicates the SQP algorithm. Its regularization may be required with the use, for example, the homotopy approach [2].

In the present paper the feasible-SQP method, known in many variants as the nonlinear programming methods [4],[7],[15], is developed as a method specialized in the multipoint shooting approach to the optimal control of PDAE systems. The exploitation of the specific structure of the method proposed enables us to resolve the difficult problem of obtaining of a feasible initial solution guaranteeing the compatibility of QP subproblems exploited in the process optimization.

Notation:  $L_\infty(\Omega, R^n)$ ,  $W_\infty^{\alpha,\beta}(\Omega, R^n)$ , and  $PC(\Omega, R^n)$  the spaces of  $n$ -dimensional essentially bounded, essentially bounded derivatives and piecewise continuous functions defined on  $\Omega$ , and  $S_r([t_0, t_f], R^n)$  the space of  $n$ -dimensional  $r$ -step functions defined on the interval  $[t_0, t_f]$ .

## II. OPTIMAL CONTROL PROBLEM FOR PARABOLIC DAE SYSTEMS

Consider the following optimal control for parabolic DAE systems (the PD problem): minimize the objective function

$$\mathcal{J}(x, z, u, u_0, p) \doteq \int_0^\tau \int_0^1 h_0(x(t, s), z(t, s), u(t, s), p) dt ds + \int_0^\tau h_1(x(t, 1), z(t, 1), u_0(t), p) dt \quad (1)$$

subject to a system of parabolic differential-algebraic equations (PDAE) of index one

$$x_t(t, s) = f(x(t, s), z(t, s), x_s(t, s), x_{ss}(t, s), u(t, s), p), \quad (2)$$

$$0 = g(x(t, s), z(t, s), u(t, s), p), \quad (t, s) \in \Omega_\tau, \quad (3)$$

to the boundary conditions

$$b_i(x(t, i), z(t, i), u_0(t), p) = 0, \quad t \in I_\tau \quad (i = 0, 1), \quad (4)$$

to the technological constraint

$$\int_0^\tau h(x(t, 1), z(t, 1), p) dt = 0, \quad (5)$$

to the box constraints

$$\begin{aligned} x(t, s) \in X(s), \quad z(t, s) \in Z(s), \quad u(t, s) \in U, \quad (t, s) \in \Omega_\tau, \\ (t, s) \in \Omega_\tau, \quad u_0(t) \in U_0, \quad t \in I_\tau, \quad p \in P, \end{aligned} \quad (6)$$

and to the physical realizability conditions for the distributed and boundary control

$$u \in PC(\Omega_\tau, R^{n_u}), \quad u_0 \in PC(I_\tau, R^{n_{u_0}}), \quad (7)$$

where  $I_\tau \doteq [0, \tau]$  is the time horizon,  $\Omega_\tau \doteq I_\tau \times I$  is the time-space domain,  $x \in W_\infty^{1,2}(\Omega_\tau, R^{n_x})$  is the differential state trajectory of the PDAE system,  $z \in L_\infty(\Omega_\tau, R^{n_z})$  is its algebraic state trajectory,  $u \in L_\infty(\Omega_\tau, R^{n_u})$  is its distributed control,  $u_0 \in L_\infty(I_\tau, R^{n_{u_0}})$  is its boundary control,  $p \in R^p$  is its global parameter, and  $X(s) \doteq [x_-(s), x_+(s)]$ ,  $Z(s) \doteq [z_-(s), z_+(s)]$ ,  $U \doteq [u_-, u_+]$ ,  $U_0 \doteq [u_{0-}, u_{0+}]$ , and  $P \doteq [p_-, p_+]$  are boxes with the spatially sensitive bounds  $x_\pm(s) \in R^{n_x}$ ,  $z_\pm(s) \in R^{n_z}$ ,  $u_\pm \in R^{n_u}$ ,  $u_{0\pm} \in R^{n_{u_0}}$  and  $p_\pm \in R^p$ , and the functions

$$h_i : R^{n_x} \times R^{n_z} \times R^{n_{u_i}} \times R^{n_p} \rightarrow R \quad (i = 0, 1) \quad u^0 \doteq u, \quad u^1 \doteq u_0,$$

$$h : R^{n_x} \times R^{n_z} \times R^{n_p} \rightarrow R^{n_h},$$

$$f : R^{n_x} \times R^{n_z} \times R^{n_u} \times R^p \rightarrow R^{n_f},$$

$$g : R^{n_x} \times R^{n_z} \times R^{n_u} \times R^p \rightarrow R^{n_g},$$

$$b_i : R^{n_x} \times R^{n_z} \times R^{n_{u_0}} \times R^{n_p} \rightarrow R^{n_{b_i}} \quad (i = 0, 1)$$

are twice continuously differentiable in all their arguments, while  $X(s)$  and  $Z(s)$  are bounded point to set mappings.

We use the time scaling  $t := t/\tau$  to normalize the time horizon to the unit interval  $t \in I$ , and to include the variable process duration  $\tau$  to the global parameter  $p$ , which may also encompass the technological and design variables (such as the level of the fixed bed catalyst or the reactor volume in chemical production processes), and the slack variables converting the technological inequality constraints into the equality ones. We reduce the general control and state inequality path constraints  $q(x(t, s), z(t, s), u(t, s), p) \leq 0$  into the equality form (3) by means of the slack control  $\tilde{u}(t, s)$  fulfilling the condition  $q(x(t, s), z(t, s), u(t, s), p) + \tilde{u}(t, s) = 0$ . Thus the formulation (1)-(7) encompasses a wide class of optimal control problems for PDAE systems with the boundary and distributed controls.

### III. TIME-SPACE MULTIPOINT SHOOTING FEASIBLE-SQP METHOD

We discretize the time coordinate  $t_k = k/l$  ( $k = 0, 1, \dots, l$ ) and the space coordinate  $s_i = i/j$  ( $i = 0, 1, \dots, j$ ). We connect with the time-space layer  $[t_k, t_{k+1}] \times [s_i, s_{i+1}]$  ( $i = 0, 1, \dots, j$ ) its shooting differential states  $x_{ik} \in R^{n_x}$ , its shooting algebraic states  $z_{ik} \in R^{n_z}$ , its shooting control variables  $u_{ik} \doteq (u_{ik1}^T, u_{ik2}^T, \dots, u_{ikr_{ik}}^T)^T \in R^{n_{u_{ik}}}$  ( $n_{u_{ik}} \doteq n_{ur_{ik}}$ ,  $r_{il} \doteq 0$ ,  $u_{i,l-1} \doteq u_{i,l-1,r_{i,l-1}}$ ), its shooting global

parameters  $p_{ik} \in R^{n_p}$ , its shooting optimization variables  $w_{ik} \doteq (x_{ik}^T, z_{ik}^T, u_{ik}^T, p_{ik}^T)^T \in R^{n_{w_{ik}}}$  ( $n_{w_{ik}} \doteq n_x + n_z + n_{u_{ik}} + n_p$ ), its layer shooting optimization variable  $w_k \doteq (w_{0k}^T, w_{1k}^T, \dots, w_{jk}^T)^T \in R^{n_{w_k}}$  ( $n_{w_k} \doteq \sum_{i=0}^j n_{w_{ik}}$ ), its differential state trajectories  $\tilde{x}_{ik} \in W_\infty^1([t_k, t_{k+1}], R^{n_x})$  determined by  $w_k$ , its algebraic state trajectories  $\tilde{z}_{ik} \in L_\infty([t_k, t_{k+1}], R^{n_z})$  determined by  $w_k$ , and its controls  $\tilde{u}_{ik} \in S_{r_{ik}}([t_k, t_{k+1}], R^{n_u})$  determined by  $u_{ik}$ . We include the boundary control shooting variables into the control variable  $u_{0k}$ , i.e.  $u_{0k} \doteq (u_{0k1}^T, u_{0k2}^T, \dots, u_{0kr_k}^T, u_{0k,r_k+1}^T, \dots, u_{0kr_{0k}}^T)^T$ , where  $u_{0kr} \in R^{n_{u_0}}$  ( $r = 1, 2, \dots, r_k$ ) are the coefficient of the boundary control, while  $u_{0kr} \in R^{n_u}$  ( $r = r_k + 1, r_k + 2, \dots, r_{ik}$ ) are the coefficients of the distributed control variable connected with the 0th spatial element. We reformulate the problem discussed as the multipoint shooting PDAE optimal control problem (the MSPD problem): minimize the objective function

$$\begin{aligned} J(w) \doteq \sum_{i=0}^{j-1} \sum_{k=0}^{l-1} \delta_j \int_{t_k}^{t_{k+1}} h_0(\tilde{x}_{ik}(t), \tilde{z}_{ik}(t), \tilde{u}_{ik}(t), p_{ik}) \\ + \sum_{k=0}^{l-1} \int_{t_k}^{t_{k+1}} h_1(\tilde{x}_{jk}(t), \tilde{z}_{jk}(t), \tilde{u}_{0k}(t), p_{jk}) \end{aligned} \quad (8)$$

subject to the continuity conditions for the differential state trajectory and the shooting parameters

$$\begin{aligned} \tilde{x}_{ik}(t_{k+1}) = x_{i,k+1}, \quad p_{ik} = p_{i,k+1} \quad (i = 0, 1, \dots, j; \\ k = 0, 1, \dots, l-1), \end{aligned} \quad (9)$$

to the consistency equations for the algebraic states and the boundary conditions

$$g_{ik}(x_{ik}, z_{ik}, u_{ik}, p_{ik}) = 0 \quad (i = 0, 1, \dots, j; \quad k = 0, 1, \dots, l), \quad (10)$$

to the technological constraint

$$\sum_{k=0}^{l-1} \int_{t_k}^{t_{k+1}} h(\tilde{x}_{jk}(t), \tilde{z}_{jk}(t), \tilde{u}_{0k}(t), p_{jk}) = 0, \quad (11)$$

and to the bound constraints

$$\begin{aligned} x_{ik} \in X_i, \quad z_{ik} \in Z_i, \quad u_{ik} \in U_{ik}, \quad p_{ik} \in P \\ (i = 0, 1, \dots, j; \quad k = 0, 1, \dots, l), \end{aligned} \quad (12)$$

where  $\delta_j \doteq 1/j$ , and the boundary conditions are incorporated into the consistency equations

$$\begin{aligned} g_{ik}(x_{ik}, z_{ik}, u_{ik}, p_{ik}) \doteq \\ (\tilde{g}^T(x_{0k}, z_{0k}, u_{0k}, p_{0k}), \tilde{b}_0^T(x_{0k}, z_{0k}, u_{0k}, p_{0k}))^T \quad (i = 0), \end{aligned}$$

and

$$g_{ik}(x_{ik}, z_{ik}, u_{ik}, p_{ik}) \doteq g(x_{ik}, z_{ik}, u_{ik}, p_{ik}) \quad (0 < i < j),$$

and

$$\begin{aligned} g_{ik}(x_{ik}, z_{ik}, u_{ik}, p_{ik}) \doteq \\ (\tilde{g}^T(x_{jk}, z_{jk}, u_{jk}, p_{jk}), \tilde{b}_1^T(x_{jk}, z_{jk}, u_{0k}, p_{jk}))^T \quad (i = j), \end{aligned}$$



and  $w \doteq (w_0^T, w_1^T, \dots, w_l^T)^T \in R^{n_w}$  ( $n_w \doteq \sum_{k=0}^l n_{w_k}$ ) is the shooting optimization variable, and  $X_i \doteq [x_{i-}, x_{i+}]$ ,  $x_{i\pm} \doteq x_{\pm}(s_i)$ ,  $Z_i \doteq [z_{i-}, z_{i+}]$ ,  $z_{i\pm} \doteq z_{\pm}(s_i)$ ,  $U_{ik} \doteq [u_{ik-}, u_{ik+}]$ ,  $u_{ik\pm} \doteq \underbrace{(u_{\pm}^T, u_{\pm}^T, \dots, u_{\pm}^T)^T}_{r_{ik+1} \text{ times}}$ , and  $b_i$  ( $i = 0, 1$ )

are the modifications of the functions  $b_i$  connected with the redefinition of the controls  $u_{0k}$ .

Let  $X_{i\epsilon_i} \doteq [\epsilon_i + x_{i-}, -\epsilon_i + x_{i+}]$  ( $\epsilon_i \in R_+^{n_x}$ ) and  $Z_{i\epsilon_i} \doteq [\epsilon_i + z_{i-}, -\epsilon_i + z_{i+}]$  ( $\epsilon_i \in R_+^{n_z}$ ) be the restricted box sets for the differential and algebraic states, and let

$$\epsilon_{in}, \epsilon_{in}, \tilde{x}_{ikn}, x_{ikn}, p_{ikn}, x_{in-}, x_{in+}, z_{in-}, z_{in+}, g_{ikn}, h_n$$

be the  $n$ th coordinates of the quantities

$$\epsilon_i, \epsilon_i, \tilde{x}_{ik}, x_{ik}, p_{ik}, x_{i-}, x_{i+}, z_{i-}, z_{i+}, g_{ik}, h.$$

*Algorithm 1:* The conversion of the time-space shooting problem (8)-(12) for the PDAE system (the MSPD problem) to the parametric MSPD problem (the MSPD<sub>c</sub> problem) with a known feasible initial solution.

*Step 0:* Choose  $\epsilon_i = 0.05(x_{i+} - x_{i-})$ ,  $\epsilon_i = 0.05(z_{i+} - z_{i-})$ ,  $\tilde{x}_{i0} \in X_{i\epsilon_i}$ ,  $\tilde{z}_{i0} \in Z_{i\epsilon_i}$ ,  $\tilde{u}_{i0} \in U_{i0}$  and  $\tilde{p}_{i0} \in P$ . Set  $\tilde{w}_{i0} \doteq (\tilde{x}_{i0}^T, \tilde{z}_{i0}^T, \tilde{u}_{i0}^T, \tilde{p}_{i0}^T)^T$ , and  $\tilde{w}_0 = (\tilde{w}_{00}^T, \tilde{w}_{10}^T, \dots, \tilde{w}_{j0}^T)^T$  and  $k = 0$ .

*Step 1:* If  $k = l$  go to *Step 4*. Else determine the differential and algebraic state trajectories  $\tilde{x}_{ik}$  and  $\tilde{z}_{ik}$  by the shot in the  $k$ th time interval for a given layer shooting optimization variable  $w_k$ .

*Step 2:* Using the results of the current shot determine

- the consecutive shooting differential states  $\tilde{x}_{i,k+1,n} = \epsilon_{in} + x_{in-}$  if  $\tilde{x}_{ikn}(t_{k+1}) \leq \epsilon_{in} + x_{in-}$ , and  $\tilde{x}_{i,k+1,n} = \tilde{x}_{ikn}(t_{k+1})$  if  $\tilde{x}_{ikn}(t_{k+1}) \in (\epsilon_{in} + x_{in-}, -\epsilon_{in} + x_{in+})$ , and  $\tilde{x}_{i,k+1,n} = -\epsilon_{in} + x_{in+}$  if  $\tilde{x}_{ikn}(t_{k+1}) \geq -\epsilon_{in} + x_{in+}$  ( $i = 0, 1, \dots, j$ ;  $n = 1, 2, \dots, n_x$ ),

- the consecutive shooting algebraic states  $\tilde{z}_{i,k+1,n} = \epsilon_{in} + z_{in-}$  if  $\tilde{z}_{ikn}(t_{k+1}) \leq \epsilon_{in} + z_{in-}$ , and  $\tilde{z}_{i,k+1,n} = \tilde{z}_{ikn}(t_{k+1})$  if  $\tilde{z}_{ikn}(t_{k+1}) \in (\epsilon_{in} + z_{in-}, -\epsilon_{in} + z_{in+})$ , and  $\tilde{z}_{i,k+1,n} = -\epsilon_{in} + z_{in+}$  if  $\tilde{z}_{ikn}(t_{k+1}) \geq -\epsilon_{in} + z_{in+}$  ( $i = 0, 1, \dots, j$ ;  $n = 1, 2, \dots, n_z$ ), and choose the consecutive shooting controls  $\tilde{u}_{i,k+1} \in U_{i,k+1}$  ( $i = 0, 1, \dots, j$ ), and the consecutive shooting parameters  $\tilde{p}_{i,k+1} = \tilde{p}_{ik}$ , and denote the solution found for the consecutive interval as  $\tilde{w}_{k+1} \doteq (\tilde{x}_{k+1}^T, \tilde{z}_{k+1}^T, \tilde{u}_{k+1}^T, \tilde{p}_{k+1}^T)^T$ .

*Step 3:* Determine

- the defect functions of the shooting differential states  $G_{1ikn}(w) \doteq \tilde{x}_{ikn}(t_{k+1}) - x_{i,k+1,n}$  if  $\tilde{x}_{ikn}(t_{k+1}) \leq \epsilon_{in} + x_{in-}$  or  $\tilde{x}_{ikn}(t_{k+1}) \in (\epsilon_{in} + x_{in-}, -\epsilon_{in} + x_{in+})$ , and  $G_{1ikn}(w) \doteq -\tilde{x}_{ikn}(t_{k+1}) + x_{i,k+1,n}$  if  $\tilde{x}_{ikn}(t_{k+1}) \geq -\epsilon_{in} + x_{in+}$  ( $i = 0, 1, \dots, j$ ;  $n = 1, 2, \dots, n_x$ ),

- the defect functions of the consistency for the algebraic states and the boundary conditions  $G_{2ikn}(w) \doteq g_{ikn}(x_{ik}, z_{ik}, u_{ik}, p_{ik})$  if  $g_{ikn}(x_{ik}, z_{ik}, u_{ik}, p_{ik}) \leq 0$ , and  $G_{2ikn}(w) \doteq -g_{ikn}(x_{ik}, z_{ik}, u_{ik}, p_{ik})$  in the opposite case ( $i = 0, 1, \dots, j$ ;  $n = 1, 2, \dots, n_z$ ),

- the defect functions for the discretized parameter  $G_{3ikn}(w) \doteq p_{ik} - p_{i,k+1}$ . Set  $k = k + 1$ .

*Step 4:* Determine the defect functions for the technological constraints

$$G_{jln}(w) \doteq \sum_{k=0}^{l-1} \int_{t_k}^{t_{k+1}} \tilde{h}(\tilde{x}_{jk}(t), \tilde{z}_{jk}(t), \tilde{u}_{0k}(t), p_{jk})$$

if

$$\sum_{k=0}^{l-1} \int_{t_k}^{t_{k+1}} \tilde{h}(\tilde{x}_{jk}(t), \tilde{z}_{jk}(t), \tilde{u}_{0k}(t), p_{jk}) \leq 0,$$

and

$$G_{jln}(w) \doteq - \sum_{k=0}^{l-1} \int_{t_k}^{t_{k+1}} \tilde{h}(\tilde{x}_{jk}(t), \tilde{z}_{jk}(t), \tilde{u}_{0k}(t), p_{jk})$$

in the opposite case ( $n = 1, 2, \dots, n_{\tilde{h}}$ ). Save the solution found as  $\tilde{w} \doteq (\tilde{w}_1^T, \tilde{w}_2^T, \dots, \tilde{w}_l^T)^T$ .

*Step 5:* Set up the functions required for the formulation of the MSD<sub>c</sub> problem:

$$G_{1ik}(w) \doteq (G_{1ikn}(w))_{n=1}^{n_x}, \quad G_{2ik}(w) \doteq (G_{2ikn}(w))_{n=1}^{n_z},$$

$$G_{3ik}(w) \doteq (G_{3ikn}(w))_{n=1}^{n_p},$$

$$G_{ik}(w) \doteq (G_{1ik}^T(w), G_{2ik}^T(w), G_{3ik}^T(w))^T,$$

$$G_{jl}(w) \doteq (G_{jln}(w))_{n=1}^{n_{\tilde{h}}},$$

$$G_{j+1+i, l+1+k}(w) \doteq (-w_{ik}^T + w_{ik-}^T, w_{ik}^T - w_{ik+}^T)^T,$$

$$w_{ik\pm} \doteq (x_{i\pm}^T, z_{i\pm}^T, u_{k\pm}^T, p_{\pm}^T)^T \quad (i = 0, 1, \dots, j; k = 0, 1, \dots, l).$$

*Step 6:* State the converted problem (the c-problem): minimize the objective function

$$J_c(w) \doteq J(w) - c \sum_{i=0}^j \sum_{k=0}^l G_{ik}(w) \quad (13)$$

subject to the constraints

$$G_{ik}(w) \leq 0 \quad (i = 0, 1, \dots, 2j + 1; k = 0, 1, \dots, 2l + 1). \quad (14)$$

where  $c \in R_+$  is the cost coefficient of the problem conversion.

If the coefficient  $c$  is sufficiently large the discretized problem and the c-problem have the same KKT points [8],[7],[15]. *Algorithm 1* yields by its formulation a feasible solution  $\tilde{w}$  of the c-problem, which can be further assumed as an initial solution  $w^0 \doteq \tilde{w}$  for efficient feasible-SQP type algorithms solving this problem. The issues connected with a suitable choice of the coefficient  $c$ , and with the verification of the feasibility of the MSPD problem (and eventually of the PD problem) by an optimal solution of the MSPD<sub>c</sub> problem are taken up by

*Algorithm 2:* The search for a locally optimal solution  $w^*$  of the MSPD problem and for a locally suboptimal layer solution  $(x_i^*, z_i^*, u_i^*, w_{0i}^*, p_i^*)$  of the PD problem by the time-space multipoint shooting feasible-SQP (TMSFSQP) method.

*Step 0:* Input the initial solution  $w^0$  found by *Algorithm 1*, a symmetric positive definite matrix  $H \in R^{n_w \times n_w}$ , and positive constants  $c_0, \bar{c}, \rho > 1$  and  $\bar{\rho} \in (0, 1)$ .

*Step 1:* To find a locally optimal solution  $w^*$  of the  $\text{MSD}_c$  problem use the Matlab R2010b feasible-SQP active set procedure solving the compatible equality constrained QP subproblems of the form

$$J'_{c,w}(w)d + \frac{1}{2}d^T H(w)d$$

s.t.

$$G'_{i_k,w}(w)d = b(w) \quad (i, k) \in \mathcal{A}_\kappa(w),$$

where  $\mathcal{A}_\kappa(w)$  is an estimate of the active set. Determine the Lagrange multipliers  $\lambda_{i_k}(c)$  associated with the constraints  $G_{i_k}(w)$  ( $i = 0, 1, \dots, j; k = 0, 1, \dots, l$ ).

*Step 2:* If  $c < \lambda_+ \doteq \max\{\lambda_{i_k}(w(c))|_\infty, i = 0, 1, \dots, j; k = 0, 1, \dots, l\}$  set  $c := \lambda_+ + \bar{c}$  and return to *Step 1*.

*Step 3:* If  $\sum_{i=0}^j \sum_{k=0}^l |G_{i_k}(w(c))|_\infty = 0$  set  $w^* = w(c)$ . Else set  $c := c + \bar{c}$ .

*Step 4:* If the bound constraints (6) for the differential states  $\tilde{x}_{i_k}(t, w^*)$  and for the algebraic states  $\tilde{z}_{i_k}(t, w^*)$ ,  $t \in [t_k, t_{k+1}]$ , ( $i = 0, 1, \dots, j; k = 0, 1, \dots, l - 1$ ) are satisfied determine a locally suboptimal layer solution of the PD problem as  $x_{i_k}^*(t) = \tilde{x}_{i_k}(t, w^*)$ ,  $z_{i_k}^*(t) = \tilde{z}_{i_k}(t, w^*)$ ,  $u_{i_k}^*(t) = \tilde{u}_{i_k}(t, w^*)$ ,  $p_{i_k}^*(t) = \tilde{p}_{i_k}(t, w^*)$ ,  $t \in [t_k, t_{k+1}]$ , ( $i = 0, 1, \dots, j; k = 0, 1, \dots, l - 1$ ). Else set  $\varepsilon := \varrho\varepsilon$  and  $\epsilon := \varrho\epsilon$  and go to *Step 0*.

The algorithm exploits the equivalence of the KKT points of the  $\text{MSPD}_c$  and  $\text{MSPD}$  problems for sufficiently large  $c$ , which should exceed the maximum modulus of the Lagrange multipliers for the converted constraints  $G_{i_k}(w)$  ( $k = 0, 1, \dots, l$ ) (the  $c$ -condition). If this condition is violated the coefficient  $c$  is increased (*Step 2*), and the optimization process is repeated. Else the fulfilling of the equality constraints of the  $\text{MSPD}$  problem is verified. It can be violated even if the  $c$ -condition is satisfied for numerical errors propagation in large-scale PDAE systems. Then some further increase of the coefficient  $c$  may be helpful (*Step 3*). The violation of the bound constraints for the differential and algebraic states can be removed by the manipulation of the parameters  $\varepsilon$  and  $\epsilon$  in view of the calmness of the layer DAE systems under discussion [12]. This leads to a locally suboptimal layer solution of the basic PD problem (*Step 4*).

The computational experience showed that the coefficient  $c$  may be overestimated leading to the ill-conditioning of the optimization problem (see fig.1). Thus the practical approach for the finding of the proper value of  $c$  may require its decrease and the repeated application of Algorithm 2 as for complex problems such as the optimization of PDAE systems.

We illustrate the theoretical considerations by the heat transfer problem for the probe heating the object with difficult accessibility, and by the performance optimization problem for chemical production processes.

*Example 1:* Consider the heat transfer process for the probe heating the object with difficult accessibility. The desired temperature profile at the right boundary of the probe should be reached avoiding overheating the zone at its left boundary, where the controlled heat source is exploited. We search for a

boundary control  $u_0$  minimizing the objective function

$$\int_0^\tau (x(t, 1) - \xi(t))^2 dt + \rho \int_0^\tau u_0^2(t) dt \quad (15)$$

subject to the heat transfer diffusion state equation

$$C(x(t, s))x_t(t, s) = Dx_{ss}(t, s), \quad (t, s) \in \Omega_\tau$$

to the initial temperature distribution

$$x(0, s) = x_0(s), \quad s \in I,$$

to the boundary conditions

$$Dx_s(t, 0) = \gamma(x(t, 0) - u_0(t)), \quad Dx_s(t, 1) = 0, \quad t \in I_\tau,$$

and to the spatially sensitive state constraints

$$x(t, s) \leq x_+(s), \quad (t, s) \in \Omega_\tau,$$

where  $x(t, s)$  is the temperature along the probe,  $u_0(t)$  is the boundary heating control,  $\xi(t)$  is the desired temperature profile,  $\rho$  is the heating cost coefficient,  $C(x(t, s)) \doteq \alpha + \beta x(t, s)$  is the temperature dependent heat capacity of the probe,  $D$  is the diffusion coefficient,  $x_+(s) \doteq a + bs$ , and  $\alpha, \beta, \gamma, a$  and  $b$  are positive constants. The problem is normalized to the form: minimize

$$\int_0^1 (x(t, 1) - \xi(t))^2 dt + \rho \int_0^1 u_0^2(t) dt$$

s.t.

$$x_t(t, s) = Dx_{ss}(t, s)/(\alpha + \beta x(t, s)), \quad (t, s) \in \Omega$$

$$x(0, s) = x_0(s), \quad s \in I,$$

$$x_s(t, 0) = \tilde{\gamma}(x(t, 0) - u_0(t)), \quad x_s(t, 1) = 0, \quad t \in I,$$

$$x(t, s) \leq a + bs, \quad (t, s) \in \Omega.$$

The distributed parameter model is approximated in each time interval  $[t_k, t_{k+1}]$  by the system of ordinary differential equations for  $i = 1, 2, \dots, j$

$$\dot{\tilde{x}}_{i_k}(t) = D_j(\tilde{x}_{i+1,k}(t) - 2\tilde{x}_{i_k}(t) + \tilde{x}_{i-1,k}(t))/(\alpha + \beta\tilde{x}_{i_k}(t)),$$

and by algebraic equation following from the approximation of the left boundary condition

$$\tilde{x}_{1k}(t) = \tilde{\gamma}(\tilde{x}_{0k}(t) - \tilde{u}_{0k}(t)),$$

where  $D_j \doteq D/\delta_j^2$ ,  $\delta_j \doteq 1/j$ . The right boundary condition determines the variable  $\tilde{x}_{j+1,k}(t)$  as  $\tilde{x}_{j+1,k}(t) \doteq \tilde{x}_{jk}(t)$ . The initial conditions are determined by the shooting variables as follows

$$\tilde{x}_{i_k}(t_k) = x_{i_k} \quad (i = 0, 1, 2, \dots, j).$$

Thus the shot in each interval  $[t_k, t_{k+1}]$  is reduced to the solving of a large-scale DAE system approximating the basic PDAE system. The distinctive features of the above example encompass the spatially dependent state constraints, the potential instabilities in the process nonlinear model caused by high sensitivity of the heat capacity to the large variations of the temperature in the probe, and the sparse structure of

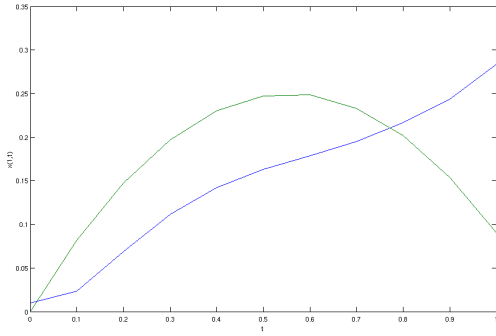


Figure 1. Too high  $c$

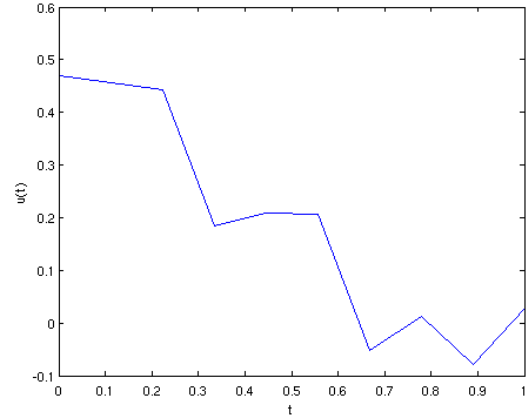


Figure 3. Optimal boundary control profile

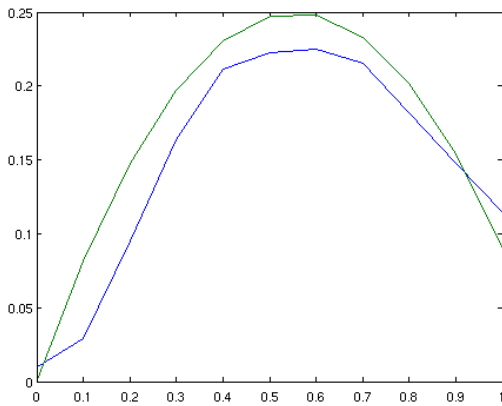


Figure 2. Resulting temperature profile

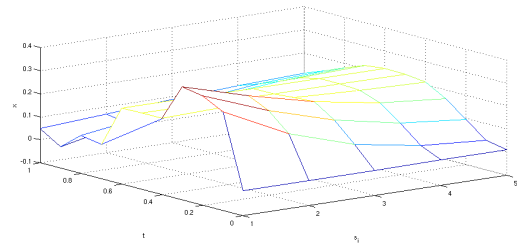


Figure 4. Time-space optimal state trajectory

the approximating DAE system, which can be advantageously exploited in the solution procedure.

In *Example 1* let's consider  $\xi(t) = 0.9t - 0.81t^2$ . For simplicity we use small number of equations  $j = 5$  and small number of shots  $l = 5$ . We use the initial condition  $x_{ik} = (0.01 \ 0.01 \ 0.01 \ 0.01 \ 0.01)$ .

The results of calculations are good but not astonishing. First of all the results depends on the coefficient  $c$ . When it is too small, the equality constrains aren't satisfied. When it is too high, the constrains are satisfied, but high ratio of penalty to square error of right boundary gives disappointing results for the desired temperature profile shown on fig. 1. We have chosen  $\rho = 0.3$ ,  $c = 0.1$ . It is giving fair result, shown on fig. 2. Optimal boundary control and optimal state surface are shown on fig.3 and fig.4.

*Example 2:* Consider the chemical production process performed in a tubular reactor. The averaged gain from the conversion of the raw material A into the desired product B at the outlet of the reactor should be maximized avoiding excessive concentrations of A in the initial zone of the reactor, which may lead to the undesirable catalyst poisoning. We search for a boundary control  $u_0(t)$ , for a distributed control  $u(t, s)$ , and for a global parameter  $p$  minimizing the objective

function

$$\int_0^1 (x(t, 1) + \rho u_0(t)) dt$$

subject to the mass transfer diffusion-convexion state equation

$$x_t(t, s) = D x_{ss}(t, s) - u(t, s) x_s(t, s) - \kappa(p) x^\alpha(t, s), \quad (t, s) \in \Omega,$$

to the initial distribution of A

$$x(0, s) = x_0(s), \quad s \in I,$$

to the boundary conditions

$$D x_s(t, 0) = v(t, 0)(x(t, 0) - u_0(t)), \quad D x_s(t, 1) = 0, \quad t \in I,$$

and to the state and control box constraints

$$\begin{aligned} x(t, s) &\in [0, x_+(s)], \quad u(t, s) \in [0, u_+], \quad (t, s) \in \Omega, \\ u_0(t) &\in [0, u_{0+}], \quad t \in I, \end{aligned}$$

where  $x(t, s)$  is the concentration of A in the reactor,  $u_0(t)$  is the inlet concentration of A,  $u(t, s)$  is the time-space variable flow intensity of the reacting mixture,  $p$  is the process temperature,  $\kappa(p) \doteq \kappa_0 e^{-\beta/p}$  is the Arrhenius function, and  $\alpha$  is the reaction order.

The distributed parameter model is approximated in each time interval  $[t_k, t_{k+1}]$  by the system of ordinary differential equations for  $i = 1, 2, \dots, j$

$$\dot{\tilde{x}}_{ik}(t) = D_j(\tilde{x}_{i+1,k}(t) - 2\tilde{x}_{ik}(t) + \tilde{x}_{i-1,k}(t))$$

$$(-\tilde{u}_{ik}(t)(\tilde{x}_{i+1,k}(t) - \tilde{x}_{ik}(t)) + \kappa(p)\tilde{x}_{ik}^\alpha(t))/\delta_j,$$

and by algebraic equation following from the approximation of the left boundary condition

$$\tilde{x}_{1k}(t) = \tilde{\gamma}\tilde{u}_{0k2}(t)(\tilde{x}_{0k}(t) - \tilde{u}_{0k1}(t)),$$

where  $D_j \doteq D/\delta_j^2$ ,  $\delta_j \doteq 1/j$ . The right boundary condition determines the variable  $\tilde{x}_{j+1,k}(t)$  as  $\tilde{x}_{j+1,k}(t) \doteq \tilde{x}_{jk}(t)$ . The initial conditions are determined by the shooting variables as follows

$$\tilde{x}_{ik}(t_k) = x_{ik} \quad (i = 0, 1, 2, \dots, j).$$

Thus the shot in each interval  $[t_k, t_{k+1}]$  is reduced as in *Example 1* to the solving of a large-scale DAE system approximating the basic PDAE system. Denoting by  $z(t, s)$  the concentration of the desired product as the algebraic state variable satisfying the algebraic equations

$$x(t, s) + z(t, s) = 1, \quad (t, s) \in \Omega,$$

we can formulate the technological constraint as the demand of the prescribed amount  $\bar{z}$  of the desired product at the outlet of the reactor in the interval  $(0, 1)$

$$\int_0^1 z(t, 1)dt = \bar{z}.$$

The distinctive features of the above example encompass the spatially dependent state constraints, the potential instabilities in the process nonlinear model caused by high sensitivity of the reaction rate to the large variations of the temperature and the concentration of A, and the sparse structure of the approximating DAE system, which can be advantageously exploited in the solution procedure.

#### IV. CONCLUSION

We presented multipoint shooting algorithms specialized to the optimization of parabolic DAE systems. They use the feasible-SQP method, which aims at the finding of a suboptimal solution of a high practical applicability. We described a procedure of obtaining an initial feasible solution for the multipoint shooting problem exploiting the results of consecutive shots of the system trajectory. We emphasized the specific features of the proposed approach such as the

spatially sensitive state constraints, the potential instabilities in the discussed heat and mass transfer processes, and the sparse structure of the large-scale DAE systems approximating the basic partial differential equations.

#### REFERENCES

- [1] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Philadelphia: SIAM, 2010.
- [2] L. T. Biegler, *Nonlinear Programming. Concepts, Algorithms, and Chemical Processes*. Philadelphia: SIAM, 2010.
- [3] K. E. Brenan, S.L. Campbell, and L.R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Philadelphia: SIAM, 1996.
- [4] L. Chen, Y. Wang, and G. He, "A feasible active set QP-free method for nonlinear programming," *SIAM Journal on Optimization*, vol. 17, pp. 401-429, 2006.
- [5] P. E. Gill, L. O. Jay, M. W. Leonard, L. R. Petzold, and V. Sharma, "An SQP method for the optimal control of large-scale dynamical systems," vol. 120, pp. 197-213, 2000.
- [6] H. K. Hesse, and G. Kanschat, "Mesh adaptive multiple shooting for partial differential equations. Part I: linear quadratic optimal control problems," *J. Numer. Math.*, vol. 17, pp. 195-217, 2009.
- [7] J.-B. Jian, C.-M. Tang, Q.-J. Hu, H.-Y. Zheng, "A feasible descent SQP algorithm for general constrained optimization without strict complementarity," *Journal of Computational and Applied Mathematics*, vol. 180, pp. 391-412, 2005. pp.1531-1542, 1986.
- [8] C. T. Lawrence and A. L. Tits, "Nonlinear equality constraints in feasible sequential quadratic programming," *Optimization Methods and Software*, vol. 6, pp. 265-282, 1996.
- [9] F. Leibfritz and E.W. Sachs, "Inexact SQP interior point methods and large scale optimal control problems," *SIAM J. Control Optim.*, vol.38, pp. 272-293, 1999.
- [10] D. B. Leineweber, I. Bauer, H. G. Bock, J. P. Schlöder, "An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part 1: theoretical aspect," *Computers and Chemical Engineering*, vol. 27, pp. 157-166, 2003.
- [11] H. D. Mittelmann and H. Maurer, "Solving elliptic control problems with interior point and SQP methods: control and state constraints," *J. of Computat. and Applied Mathematics*, vol. 120, pp. 175-195, 2000.
- [12] R. Pytlak, "Optimal control of differential-algebraic equations of higher index, Part 1: First-order approximations," *J. Optimiz. Theory Appl.*, vol. 134, pp. 61-75, 2007.
- [13] R. Serban, S. Li, and L. R. Petzold, "Adaptive algorithms for optimal control of time-dependent partial differential-algebraic equation systems," *Intern. Journal for Numerical Methods in Engineering*, vol. 57, pp. 1457-1469, 2003.
- [14] M. J. Śmietański, "Inexact quasi-Newton global convergent method for solving constrained nonsmooth equations," *International Journal of Computer Mathematics*, vol. 84, pp.1757-1770, 2007.
- [15] Z. Zhu, W. Zhang, and Z. Geng, "A feasible SQP method for nonlinear programming," *Applied Mathematics and Computation*, vol. 215, pp. 3956-3969, 2010.

# A modified multipoint shooting feasible-SQP method for optimal control of DAE systems

Krystyn Styczeń

Institute of Computer Engineering  
Wrocław University of Technology  
Janiszewskiego 11/17, 50-372 Wrocław, Poland  
Email: krystyn.styczen@pwr.wroc.pl

Paweł Draj

Institute of Computer Engineering  
Wrocław University of Technology  
Janiszewskiego 11/17, 50-372 Wrocław, Poland  
Email: pawel.drag@pwr.wroc.pl

**Abstract**—Optimal control problem for state-constrained differential-algebraic (DAE) systems is considered. Such problems can be attacked by the multiple shooting approach well suited to unstable and ill-conditioned dynamic systems. According to this approach the control interval is partitioned into shorter intervals allowing the parallelization of computations with the reliable using of DAE solvers. A new modified method of this kind is proposed, which converts the partitioned problem with mixed equality and inequality constraints into the purely inequality constrained problem. An algorithm for obtaining a feasible initial solution of the converted problem is described. A feasible-SQP algorithm based on an active set strategy is applied to the converted problem. It avoids the inconsistency of the constraints of the QP subproblems (versus the infeasible path SQP methods) and delivers a locally optimal solution of the basic problem preserving all its constraints (including the equality ones), which is of a high practical meaning. Some further developments concerning the regularization of suboptimal solutions for large-scale DAE optimal control problems and multilevel versions of the method proposed are also discussed. The theoretical considerations are illustrated by a numerical example of optimization of a complex DAE chemical engineering system.

## I. INTRODUCTION

THE DEVELOPMENT of advanced technologies is often connected with the design of complex systems described by large-scale differential-algebraic equations (DAE) subject to the control and state path constraints, and to the terminal state constraints [3],[4]. Many of such systems possess unstable dynamic modes and high sensitivity to parameter changes. This may lead to unbounded state profiles and to ill-conditioning of the Jacobian and the Hessian matrices if the single shooting method is used to optimize the system performance. This also complicates the optimization of dynamic systems with boundary value conditions. The multiple shooting (MS) method has been proposed to resolve the above difficulties [11]. It discretizes the control interval into shorter subintervals within which the dynamic model is integrated independently and linked by the continuity constraints in the discretized model. Such an approach is well suited to deal with unstable and highly sensitive modes of nonlinear dynamic systems. Moreover, it guarantees the full parallelization of the data computations necessary for the application of advanced second-order optimization methods such as the sequential quadratic programming (SQP) method or the interior point

(IP) method [4]. Many further advantages of the MS method concerning the exploitation of the problem sparse structure, the computation of the state sensitivities by the specialized algorithms, and the employment of the reduced or partially reduced SQP algorithms are depicted in the literature [11]. However, the main elaborations on this subject are connected with the iterative infeasible path approach, where all the constraints may be violated on the current iteration, and reached solely at the limit of the subsequence of convergent iterations. This may complicate the SQP algorithm because of the inconsistency of the constraints for the QP subproblem, and the need of its regularization, for example, by the homotopy approach [4]. Thus the current solution obtained at the moderate computation time may be impractical for its infeasibility.

In the present paper the feasible-SQP approach, known in many variants as the nonlinear programming methods guaranteeing the feasibility of the current solution at all the optimization iterations [6],[10],[15], is developed as a method specialized in the multiple shooting approach to the optimal control of DAE systems [5]. One of the main difficulties in the use of the feasible-SQP algorithms is the requirement of the knowledge of an initial feasible solution satisfying all the constraints. The problem of the determination of such an solution may be as difficult as the basic optimization problem. We show, however, that the exploitation of the specific structure of the MS method enables us to conveniently resolve this problem. To this end we apply the c-conversion [10] of the partitioned shooting problem with mixed equality and inequality constraints into the purely inequality problem. We describe an algorithm, which determines a feasible initial solution of the converted problem exploiting the results of the consecutive shots. Next we present a modified algorithm for the choice of the conversion coefficient, which employs the MATLAB R2010b feasible-SQP active set procedure. It avoids the inconsistency of the constraints of the QP subproblems and allows the full parallelization of the optimization computations. We verify the algorithms developed by a numerical example of optimization of a complex DAE chemical engineering system. Finally we discuss the regularization of suboptimal solutions for large-scale DAE optimal control problems with the help of bound constrained Newton

method [2], [14], and multilevel versions of the method proposed [7].

Notation:  $L_\infty^n(t_0, t_f)$ ,  $W_\infty^{1,n}(t_0, t_f)$  and  $PC^n(t_0, t_f)$  the spaces of  $n$ -dimensional essentially bounded, essentially bounded derivative, and piecewise continuous functions defined on the interval  $[t_0, t_f]$ , respectively;  $S_r^n(t_0, t_f)$  the space of  $n$ -dimensional  $r$ -step functions defined on the interval  $[t_0, t_f]$ ,  $n_x$  the dimension of a vector  $x$ ,  $|x|_\infty \doteq \max\{|x_n| : n = 1, 2, \dots, n_x\}$ .

## II. OPTIMAL CONTROL PROBLEM FOR DAE SYSTEMS

Consider the following optimal control problem for DAE systems (the D problem): minimize the objective function

$$\mathcal{J}(x, z, u, p) \doteq h(x(1), z(1), p) \quad (1)$$

subject to a system of differential-algebraic equations of index one

$$\dot{x}(t) = f(x(t), z(t), u(t), p, t), \quad t \in [0, 1], \quad (2)$$

$$0 = g(x(t), z(t), u(t), p, t), \quad t \in [0, 1], \quad (3)$$

to the terminal constraint

$$\tilde{h}(x(1), z(1), p) = 0, \quad (4)$$

to the bound constraints

$$x(t) \in X, \quad z(t) \in Z, \quad u(t) \in U, \quad t \in [0, 1], \quad p \in P, \quad (5)$$

and to the physical realizability condition for the control

$$u \in PC^{n_u}(0, 1), \quad (6)$$

where  $x \in W_\infty^{1,n_x}(0, 1)$  is the differential state trajectory of the DAE system,  $z \in L_\infty^{n_z}(0, 1)$  is its algebraic state trajectory,  $u \in L_\infty^{n_u}(0, 1)$  is its control,  $p \in R^{n_p}$  is its global parameter, and  $X \doteq [x_-, x_+]$ ,  $Z \doteq [z_-, z_+]$ ,  $U \doteq [u_-, u_+]$ , and  $P \doteq [p_-, p_+]$  are parallelepipeds with the bounds  $x_\pm \in R^{n_x}$ ,  $z_\pm \in R^{n_z}$ ,  $u_\pm \in R^{n_u}$  and  $p_\pm \in R^{n_p}$ , and the functions

$$h : R^{n_x} \times R^{n_z} \times R^{n_p} \rightarrow R, \quad \tilde{h} : R^{n_x} \times R^{n_z} \times R^{n_p} \rightarrow R^{n_h},$$

$$f : R^{n_x} \times R^{n_z} \times R^{n_u} \times R^p \times R \rightarrow R^{n_x},$$

$$g : R^{n_x} \times R^{n_z} \times R^{n_u} \times R^{n_p} \times R \rightarrow R^{n_g}$$

are twice continuously differentiable in all their arguments.

We normalize the nonunit control interval  $[0, \tau]$ ,  $\tau \neq 1$  by the time scaling  $t := t/\tau$ . We include the variable process duration  $\tau$  into the global parameter  $p$ . The latter parameter may also concern the design variables (such as the level of the fixed bed catalyst or the reactor volume) and the slack variables converting the terminal inequality constraints into the equality ones. We reduce the general control and state inequality path constraints  $q(x(t), z(t), u(t), t) \leq 0$  to the equality form (3) with the help of the slack control  $\tilde{u}(t) \geq 0$  satisfying the condition  $q(x(t), z(t), u(t), t) + \tilde{u}(t) = 0$ ,  $t \in [0, \tau]$ . Thus the formulation (1)-(6) encompasses a wide class of optimal control problems for DAE systems.

## III. MODIFIED MULTIPOINT SHOOTING FEASIBLE-SQP METHOD

We use the discretized time  $t_k = k/l$  ( $k = 0, 1, \dots, l$ ). We connect with the time interval  $[t_k, t_{k+1}]$  its shooting initial differential state  $x_k \in R^{n_x}$ , its shooting initial algebraic state  $z_k \in R^{n_z}$ , its shooting control parameters  $u_k \doteq (u_{k1}^T, u_{k2}^T, \dots, u_{kr_k}^T)^T \in R^{n_{u_k}}$  ( $n_{u_k} \doteq n_u r_k$ ,  $r_l = 0$ ,  $u_{l1} \doteq u_{l-1, r_{l-1}}$ ), its shooting global parameter  $p_k \in R^{n_p}$ , and its shooting solution  $w_k \doteq (x_k^T, z_k^T, u_k^T, p_k^T)^T \in R^{w_k}$  ( $n_{w_k} \doteq n_x + n_z + n_{u_k} + n_p$ ), its differential state trajectory  $\tilde{x}_k \in W_\infty^{1, n_x}(t_k, t_{k+1})$  determined by  $w_k$ , its algebraic state trajectory  $\tilde{z}_k \in L_\infty^{n_z}(t_k, t_{k+1})$  determined by  $w_k$ , and its control  $\tilde{u}_k \in S_{r_k}^{n_u}(t_k, t_{k+1})$  determined by  $u_k$ . We reformulate the D problem as the multipoint shooting problem for DAE systems (the MSD problem): minimize the objective function

$$J(w) \doteq h(x_l, z_l, p_l) \quad (7)$$

subject to the continuity conditions for the differential state trajectory and the discretized parameters

$$\tilde{x}_k(t_{k+1}, w_k) - x_{k+1} = 0, \quad p_k - p_{k+1} = 0 \quad (k = 0, 1, \dots, l-1), \quad (8)$$

to the consistency conditions for the algebraic states

$$g(x_k, z_k, u_{k1}, p_k, t_k) = 0 \quad (k = 0, 1, \dots, l), \quad (9)$$

to the terminal equality constraints

$$\tilde{h}(x_l, z_l, p_l) = 0, \quad (10)$$

and to the bound constraints

$$x_k \in X, \quad z_k \in Z, \quad u_k \in U_k, \quad p_k \in P \quad (k = 0, 1, \dots, l), \quad (11)$$

where  $w \doteq (w_1^T, w_2^T, \dots, w_l^T)^T \in R^{n_w}$  ( $n_w \doteq n_{w_1} + n_{w_2} + \dots + n_{w_l}$ ) is the solution of the MSD problem, and  $U_k \doteq [u_{k-}, u_{k+}]$ ,  $u_{k\pm} \doteq \underbrace{(u_{\pm 1}^T, u_{\pm 2}^T, \dots, u_{\pm r_k}^T)^T}_{r_k \text{ times}}$ .

Let  $X_\varepsilon \doteq [\varepsilon + x_-, -\varepsilon + x_+]$  ( $\varepsilon \in R_+^{n_x}$ ) and  $Z_\varepsilon \doteq [\varepsilon + z_-, -\varepsilon + z_+]$  ( $\varepsilon \in R_+^{n_z}$ ) be the restricted bound sets for the differential and algebraic states, and let  $\varepsilon_n, \varepsilon_n, \tilde{x}_{kn}, x_{kn}, p_{kn}, x_{n-}, x_{n+}, z_{n-}, z_{n+}, g_n$  and  $\tilde{h}_n$  be the  $n$ th coordinates of the quantities  $\varepsilon, \varepsilon, \tilde{x}_k, x_k, p_k, x_-, x_+, z_-, z_+, g$  and  $\tilde{h}$ .

*Algorithm 1:* The conversion of the MSD problem (7)-(11) to the parametric MSD<sub>c</sub> problem with a known feasible initial solution.

*Step 0:* Choose  $\varepsilon = 0.05(x_+ - x_-)$ ,  $\varepsilon = 0.05(z_+ - z_-)$ ,  $\check{x}_0 \in X_\varepsilon$ ,  $\check{z}_0 \in Z_\varepsilon$ ,  $\check{u}_0 \in U_0$  and  $\check{p}_0 \in P$ , set  $\check{w}_0 \doteq (\check{x}_0^T, \check{z}_0^T, \check{u}_0^T, \check{p}_0^T)^T$  and  $k = 0$ .

*Step 1:* If  $k = l$  go to *Step 4*. Else determine the differential and algebraic state trajectories  $\tilde{x}_k$  and  $\tilde{z}_k$  by the shot in the  $k$ th time interval.

*Step 2:* Using the results of the current shot determine

- the consecutive shooting differential state  $\check{x}_{k+1, n} = \varepsilon_n + x_{n-}$  if  $\tilde{x}_{kn}(t_{k+1}) \leq \varepsilon_n + x_{n-}$ , and  $\check{x}_{k+1, n} = \tilde{x}_{kn}(t_{k+1})$  if  $\tilde{x}_{kn}(t_{k+1}) \in (\varepsilon_n + x_{n-}, -\varepsilon_n + x_{n+})$ , and  $\check{x}_{k+1, n} = -\varepsilon_n + x_{n+}$  if  $\tilde{x}_{kn}(t_{k+1}) \geq -\varepsilon_n + x_{n+}$  ( $n = 1, 2, \dots, n_x$ ),

• the consecutive shooting algebraic state  $\check{z}_{k+1,n} = \epsilon_n + z_{n-}$  if  $\check{z}_{kn}(t_{k+1}) \leq \epsilon_n + z_{n-}$ , and  $\check{z}_{k+1,n} = \check{z}_{kn}(t_{k+1})$  if  $\check{z}_{kn}(t_{k+1}) \in (\epsilon_n + z_{n-}, -\epsilon_n + z_{n+})$ , and  $\check{z}_{k+1,n} = -\epsilon_n + z_{n+}$  if  $\check{z}_{kn}(t_{k+1}) \geq -\epsilon_n + z_{n+}$  ( $n = 1, 2, \dots, n_z$ ), and choose the consecutive shooting control  $\check{u}_{k+1} \in U_{k+1}$ , and the consecutive shooting parameter  $\check{p}_{k+1} = \check{p}_k$ , and denote the solution found for the consecutive interval as  $\check{w}_{k+1} \doteq (\check{x}_{k+1}^T, \check{z}_{k+1}^T, \check{u}_{k+1}^T, \check{p}_{k+1}^T)^T$ .

*Step 3:* Determine

• the defect functions for the shooting differential states  $G_{1kn}(w) \doteq \check{x}_{kn}(t_{k+1}) - x_{k+1,n}$  if  $\check{x}_{kn}(t_{k+1}) \leq \epsilon_n + x_{n-}$  or  $\check{x}_{kn}(t_{k+1}) \in (\epsilon_n + x_{n-}, -\epsilon_n + x_{n+})$ , and  $G_{1kn}(w) \doteq -\check{x}_{kn}(t_{k+1}) + x_{k+1,n}$  if  $\check{x}_{kn}(t_{k+1}) \geq -\epsilon_n + x_{n+}$  ( $n = 1, 2, \dots, n_x$ ),

• the defect functions for the shooting algebraic states  $G_{2kn}(w) \doteq g_n(x_k, z_k, u_{k1}, p_k, t_k)$  if  $g_n(x_k, z_k, u_{k1}, p_k, t_k) \leq 0$ , and  $G_{2kn}(w) \doteq -g_n(x_k, z_k, u_{k1}, p_k, t_k)$  in the opposite case ( $n = 1, 2, \dots, n_z$ ),

• the defect functions for the shooting parameter  $G_{3kn}(w) \doteq p_k - p_{k+1}$ . Set  $k = k + 1$ .

*Step 4:* Determine the defect functions for the terminal constraints  $G_{ln}(w) \doteq \check{h}_n(x_l, z_l, p_l)$  if  $\check{h}_n(x_l, z_l, p_l) \leq 0$ , and  $G_{ln}(w) \doteq -\check{h}_n(x_l, z_l, p_l)$  in the opposite case ( $n = 1, 2, \dots, n_h$ ). Save the solution found as  $\check{w} \doteq (\check{w}_1^T, \check{w}_2^T, \dots, \check{w}_l^T)^T$ .

*Step 5:* Set up the functions required for the formulation of the MSD<sub>c</sub> problem:

$$G_{1k}(w) \doteq (G_{1kn}(w))_{n=1}^{n_x}, \quad G_{2k}(w) \doteq (G_{2kn}(w))_{n=1}^{n_z},$$

$$G_{3k}(w) \doteq (G_{3kn}(w))_{n=1}^{n_p},$$

$$G_k(w) \doteq (G_{1k}^T(w), G_{2k}^T(w), G_{3k}^T(w))^T \quad (k = 0, 1, \dots, l-1),$$

$$G_l(w) \doteq (G_{ln}(w))_{n=1}^{n_h},$$

$$G_{l+1+k}(w) \doteq (-w_k^T + w_{k-}^T, w_k^T - w_{k+}^T)^T,$$

$$w_{k\pm} \doteq (x_{\pm}^T, z_{\pm}^T, u_{k\pm}^T, p_{\pm}^T)^T \quad (k = 0, 1, \dots, l).$$

*Step 6:* State the MSD<sub>c</sub>: minimize the objective function

$$J_c(w) \doteq J(w) - c \sum_{k=0}^l G_k(w_k) \quad (12)$$

subject to the constraints

$$G_k(w) \leq 0 \quad (k = 0, 1, \dots, 2l + 1). \quad (13)$$

where  $c \in R_+$  is the cost coefficient of the problem conversion.

If the coefficient  $c$  is sufficiently large the MSD problem and the MSD<sub>c</sub> problem have the same KKT points [12],[8],[10],[15]. *Algorithm 1* yields by its formulation a feasible solution  $\check{w}$  of the MSD<sub>c</sub> problem, which can be further assumed as an initial solution  $w^0 \doteq \check{w}$  for efficient feasible-SQP type algorithms solving this problem. The issues concerning a suitable choice of the coefficient  $c$ , and the verification of the feasibility of the MSD problem (and eventually of the

D problem) by an optimal solution of the MSD<sub>c</sub> problem are taken up by

*Algorithm 2:* The search for a locally optimal solution  $w^*$  of the MSD problem and for a locally suboptimal solution  $(x^*, z^*, u^*, p^*)$  of the D problem by the multipoint shooting feasible-SQP (MSFSQP) method.

*Step 0:* Input the initial solution  $w^0$  found by *Algorithm 1*, a symmetric positive definite matrix  $H \in R^{n_w \times n_w}$ , and positive constants  $c, \bar{c}$  and  $\rho > 1$ .

*Step 1:* Use the Matlab R2010b feasible-SQP active set procedure to find a locally optimal solution  $w(c)$  of the MSD<sub>c</sub> problem, and the Lagrange multipliers  $\lambda_k(c)$  associated with the constraints  $G_k(w)$  ( $k = 0, 1, \dots, l$ ).

*Step 2:* If  $c < \lambda_+(c) \doteq \max\{|\lambda_k(c)|_\infty, k = 0, 1, \dots, l\}$  set  $c := \lambda_+(c) + \bar{c}$  and return to *Step 1*.

*Step 3:* If  $\sum_{k=0}^l |G_k(w(c))|_\infty = 0$  set  $w^* = w(c)$ . Else set  $c := c + \bar{c}$ .

*Step 4:* If the bound constraints (5) for the differential states  $\tilde{x}_k(t, w^*)$  and for the algebraic states  $\tilde{z}_k(t, w^*)$ ,  $t \in [t_k, t_{k+1}]$ , ( $k = 0, 1, \dots, l-1$ ) are satisfied determine a locally suboptimal feasible solution of the D problem as  $x^*(t) = \tilde{x}_k(t, w^*)$ ,  $z^*(t) = \tilde{z}_k(t, w^*)$ ,  $u^*(t) = \tilde{u}_k(t, w^*)$ ,  $p^*(t) = \tilde{p}_k(t, w^*)$ ,  $t \in [t_k, t_{k+1}]$ , ( $k = 0, 1, \dots, l-1$ ). Else set  $\epsilon := \rho\epsilon$  and  $\epsilon := \rho\epsilon$  and go to *Step 0*.

The algorithm exploits the equivalence of the KKT points of the MSD<sub>c</sub> and MSD problems for sufficiently large  $c$ , which should exceed the maximum modulus of the Lagrange multipliers for the converted constraints  $G_k(w)$  ( $k = 0, 1, \dots, l$ ) (the  $c$ -condition). If this condition is violated the coefficient  $c$  is increased (*Step 2*), and the optimization process is repeated. Else the fulfilling of the equality constraints of the MSD problem is verified. It can be violated even if the  $c$ -condition is satisfied for numerical errors propagation in large-scale DAE systems. Then some further increase of the coefficient  $c$  may be helpful (*Step 3*). The violation of the bound constraints for the differential and algebraic states can be removed by the manipulation of the parameters  $\epsilon$  and  $\epsilon$  in view of the calmness of the DAE systems under discussion [13]. This leads to a locally suboptimal feasible solution of the basic D problem (*Step 4*).

The regularization of the solution  $w(c)$  satisfying insufficiently accurately the equality constraints of the D problem may concern the application of the bound-constrained trust-region and inexact Newton method. In particular the consistent algebraic states for large-scale DAE systems can be found with the help of the superlinearly convergent trust-region approach [1],[2]: minimize in  $\delta z_k$  the quadratic model of the consistency equations

$$\frac{1}{2} |g(x_k, z_k, u_{k1}, p_k, t_k) + g'_{z_k}(x_k, z_k, u_{k1}, p_k, t_k) \delta z_k|^2$$

subject to the trust-region constraints

$$|\mathcal{D} \delta z_k| \leq \Delta,$$

where  $\mathcal{D}$  is the scaling matrix and  $\Delta$  is the trust-region matrix. This approach may be yet enhanced by the bound-constrained



inexact Newton method [14] applied to the consistency equations (9) in  $z_k$

$$\begin{aligned} z_k^+ &= z_k + \alpha \delta z_k, \quad z_k^+ \in Z_k, \\ |g(x_k, z_k^+, u_{k1}, p_k, t_k) + g'_{z_k}(x_k, z_k, u_{k1}, p_k, t_k) \delta z_k| \\ &< \beta |g(x_k, z_k, u_{k1}, p_k, t_k)|, \end{aligned}$$

where  $\alpha, \beta \in (0, 1)$ .

The feasibility of the solution obtained may facilitate the incorporation of the fixed dimension  $l$  multiple shooting method into the variable dimension  $l$  multiple shooting method exploiting the multilevel feasible approach based on the convergence of point-to-set mappings [7]. The imposition of the lower dimension variations of the shooting controls on the fine dimension shooting solution does not destroy the problem feasibility because the zero solution is feasible in the lower dimension problem.

A wide class of complex DAE systems is encountered in chemical engineering. This concerns, for example, processes of nonlinear chemical reactions performed in tank reactors or multizone reactors, and processes of heat exchange, distillation and separation [4]. A high practical meaning has the optimization of integrated processes of such a kind, which leads to the problems with complex DAE models requiring advanced optimization methods. The application of the above described algorithms to optimal control of some chemical engineering systems is proposed with the use of the Matlab toolbox of the parallel computations.

#### IV. NUMERICAL EXAMPLES

The new method, which we presented above, now we would like to use to solve a certain D problem [9]. Before we solve this task with the presented method and give results, we are going to introduce this problem and its details. Then we are going to be able to better understand the method presented in this paper.

Y. J. Huang et al (19) described a model decomposition based method for solving general dynamic optimization problems. The authors gave in their paper three interesting examples from chemical engineering. There were catalyst mixing problem, fed-batch penicillin fermentation and pressure-constrained batch reactor. From our point of view the most interesting is the  $3^{rd}$  problem.

In the reactor three reactions take place.  $A \rightarrow 2B$ , with reaction constant  $k_1$ . There is a reverse reaction  $2B \rightarrow A$ , with reaction constant  $k_2$ . The last reaction is  $A + B \rightarrow D$ , with reaction constant  $k_3$ . Description of the dynamic optimization problem is as follows

$$\min_F J = C_D(t_f),$$

subject to

$$\begin{aligned} \dot{C}_A &= -k_1 C_A + k_2 C_B C_B + \frac{F}{V} - k_3 C_A C_B, \\ \dot{C}_B &= k_1 C_A - k_2 C_B C_B - k_3 C_A C_B, \end{aligned}$$

$$\dot{C}_D = k_3 C_A C_B.$$

There are the algebraic and state constraints too

$$\begin{aligned} N &= V(C_A + C_B + C_D), \\ PV &= NRT, \\ P &\leq 340000[Pa], \\ 0 &\leq F \leq 8.5 \left[ \frac{mol}{h} \right], \end{aligned}$$

together with initial conditions

$$[C_A(0), C_B(0), C_D(0)] = [100, 0, 0].$$

We know, that  $t_f = 2$  hours.

To complete the description, there are values of another magnitudes used in equations:  $k_1 = 0.8 \left[ \frac{1}{h} \right]$ ,  $k_2 = 0.02 \left[ \frac{m^3}{mol \cdot h} \right]$ ,  $k_3 = \left[ \frac{m^3}{mol \cdot h} \right]$ , the volume  $V = 1.0 [m^3]$  and the temperature  $T = 400 [K]$ .

For purposes of our presentation, we can rewrite the constraints equations. Because

$$N = V(C_A + C_B + C_D)$$

and

$$PV = NRT,$$

so we have two equations with two unknowns  $N$  and  $P$ . Then we can write

$$PV = V(C_A + C_B + C_D)RT.$$

Now, because  $V = 1 \Rightarrow V \neq 0$ , in the next step we can state, that

$$P = (C_A + C_B + C_D)RT.$$

We know, that the gas constant equals to  $R = 8.314472 \left[ \frac{J}{mol \cdot K} \right]$  and in this situation the temperature is constant too, so

$$P \leq 340000 \Rightarrow (C_A + C_B + C_D)RT \leq 340000.$$

The last step is to compute the constraint explicite

$$(C_A + C_B + C_D) \leq \frac{340000}{RT} \Rightarrow (C_A + C_B + C_D) \leq 102.2314 \left[ \frac{mol}{m^3} \right].$$

We can check our calculations

$$\begin{aligned} \left[ \frac{Pa}{\frac{J}{mol \cdot K} \cdot K} \right] &= \left[ \frac{Pa \cdot mol \cdot K}{J \cdot K} \right] = \left[ \frac{Pa \cdot mol}{J} \right] = \\ &= \left[ \frac{\frac{N}{m^2} \cdot mol}{\frac{kg \cdot m^2}{s^2}} \right] = \left[ \frac{\frac{kg \cdot m}{s^2 \cdot m^2} \cdot mol}{\frac{kg \cdot m^2}{s^2}} \right] = \left[ \frac{kg \cdot mol \cdot s^2}{s^2 \cdot m \cdot kg \cdot m^2} \right] = \left[ \frac{mol}{m^3} \right]. \end{aligned}$$

Now this problem has another constraints, which have the same meaning: one state constraint and one constrained control variable.

To better understand this problem, we made some easy simulations for various constant control variable with using single shooting method. All simulation were made in Matlab R2010b, with settings  $RelTol = 10^{-7}$ ,  $AbsTol = 10^{-7}$  for DAE solver *ode15s*,  $TolFun = 10^{-7}$ ,  $TolX = 10^{-7}$  and

Table I  
RESULTS FOR SINGLE SHOOTING METHOD

$F$	$C_A(t_f)$	$C_B(t_f)$	$C_D(t_f)$	$\sum_{i=A,B,D} C_i(t_f)$
8.00	50.66	41.60	11.87	104.13
6.93	49.35	41.21	11.64	102.20
6.00	48.23	40.88	11.45	100.56
5.00	47.01	40.51	11.24	98.76
4.00	45.80	40.15	11.03	96.98
0.00	40.93	38.66	10.21	89.80

*fin-diff-grads* as Hessian update method for SQP active set method optimization algorithm.

In the Table I, where the solutions are given, we observe that for the problem stated in paper [9], the solution is  $F = 0$ .

When we would like to compare our results with results in [9], we have to change the objective function

$$\max_F J = C_D(t_f),$$

subject to the new constraints, which make in this situation some difficulties.

Now we want to present our results and some steps, which the algorithm made. We performed the study in this way. We divided the time domain in 2, 4 and 10 parts. It means, that we have to do 2, 4 and 10 shots, respectively. In every part we can use one constant control function. Then we solved the same problem with only 2 shots, but there were 2 and 5 piecewise control functions in each time interval. Thus we considered 5 ways and in this situation we can compare methods with less number of shots, but the same number of control functions, 4 and 10, respectively. In examples we divided time domain into equal parts.

#### A. 2 shots and 1 control function in each interval

We have two control functions  $u_0$  and  $u_1$  for first and second time interval, respectively. Because the control functions are constant, at the beginning of each interval they have to satisfy the constraints

$$0 \leq u_0 \leq 8.5$$

and

$$0 \leq u_1 \leq 8.5.$$

State variables at the beginning of the second interval should satisfy the inequalities

$$0 \leq C_A(0.5t_f) \leq 100,$$

$$0 \leq C_B(0.5t_f) \leq 70,$$

$$0 \leq C_D(0.5t_f) \leq 20.$$

The above constraints are not very restrictive. They are useful for shooting method, because now we can look for solutions in a reasonable range. Next constraint is more restrictive

$$C_A(0.5t_f) + C_B(0.5t_f) + C_D(0.5t_f) \leq 102.2314.$$

We have to converse this problem to the c-problem. The first step is to form vectors, which would be able to describe the

Table II  
SUCCESSIVE ITERATIONS FOR PROBLEM WITH 2 SHOTS AND 1 CONTROL FUNCTION IN EACH INTERVAL

iter	$u_0$	$C_{A_{0.5t_f}}$	$C_{B_{0.5t_f}}$	$C_{D_{0.5t_f}}$	$u_1$	c-prob
0	8.5000	59.0719	38.8453	5.2914	8.5000	8.2166
1	8.5000	58.1862	37.9737	4.7914	8.0594	7.6820
2	6.8264	57.3532	37.1814	5.0910	6.7671	9.2803
3	5.9685	57.1764	37.0381	6.3927	6.2319	9.8480
4	5.7684	57.3315	37.2011	6.4248	6.1903	9.8829
5	5.6935	57.2165	37.7093	6.3056	6.2560	10.9499
6	5.9701	57.2828	38.1142	6.2759	6.3548	11.0845
7	6.1807	57.2784	38.4279	6.1808	6.4828	11.3588
8	6.1790	57.2763	38.4529	6.1687	6.4910	11.3766
9	6.2562	57.3946	38.4548	6.1249	6.5671	11.4592
10	6.2815	57.4468	38.4576	6.0859	6.6125	11.4756
11	6.2819	57.4646	38.4571	6.0607	6.6228	11.4872
12	6.3616	57.5148	38.4753	5.9604	6.6138	11.5295
13	6.5238	57.6393	38.5097	5.7744	6.6390	11.5597
14	7.0229	58.0028	38.6019	5.2676	6.7087	11.6343
15	7.0690	58.0363	38.6104	5.2196	6.7166	11.6414
16	7.0697	58.0367	38.6105	5.2182	6.7175	11.6416
17	7.0698	58.0368	38.6105	5.2182	6.7175	11.6416
18	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
19	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
20	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
21	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
22	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
23	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416

parts of the problem. So, this vector, which represents results of one part, can have a form

$w_i = (\text{initial values of state variables, control variable})$ .

We know, that each interval has one control variable. Then we can write

$$w_0 = (C_A(0), C_B(0), C_D(0), u_0),$$

$$w_1 = (C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f), u_1).$$

Because we want to keep state constraints, we have to check, if the state variable are near from bounds. If they are, we use the procedure described in paper. As the result we have

$$\tilde{w} = \begin{pmatrix} 100.0000 & 0 & 0 & 8.5000 \\ 59.0719 & 38.8453 & 5.2914 & 8.5000 \end{pmatrix}$$

Now we have to introduce defect functions. This is the main idea. From vector  $w_0$  we know, that we start from the point  $(C_A(0), C_B(0), C_D(0))$  together with control function  $u_0$ . At the end of the first interval the process finishes with some results. Then the optimization algorithm should choose the variables  $(C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f))$ , that the differences between results from the first interval and the initial points for second interval are minimized. Together with the final condition we have 4 constraints. Now we can start the optimization process with 5 variables: 3 discretized state variables  $(C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f))$ , and 2 control variables  $u_0$  and  $u_1$ . Because this example was quite easy, we can give a Table II with all iterations ( $c = 1$ ).

The solution we can see on the Figure 1.

When  $c$  is too small, for example  $c = 0.1$ , then we can have an useless solution (Figure 2).

Figure 1. Problem with 2 shots and 1 control function in each interval.

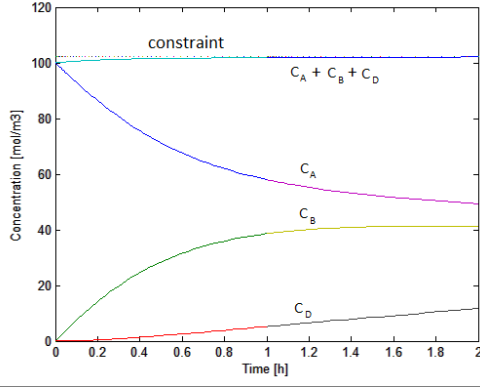
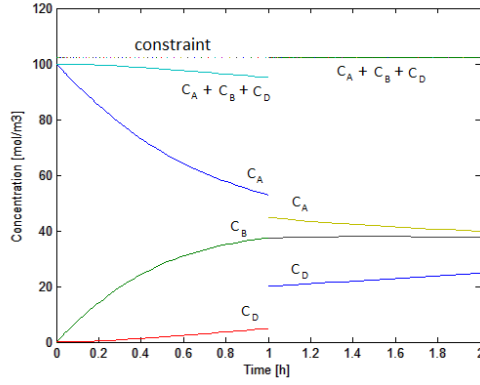


Figure 2. Problem with 2 shots and 1 control function in each interval, when c is too small.



### B. 4 shots and 1 control function in each interval

We start to construct vectors  $w_i$

$$w_0 = (C_A(0), C_B(0), C_D(0), u_0),$$

$$w_1 = (C_A(0.25t_f), C_B(0.25t_f), C_D(0.25t_f), u_1),$$

$$w_2 = (C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f), u_2),$$

$$w_3 = (C_A(0.75t_f), C_B(0.75t_f), C_D(0.75t_f), u_3).$$

Now we have 9 discretized state variables and 4 control functions. Together there are 13 variables with 3 constraints

$$C_A(0.25t_f), C_B(0.25t_f), C_D(0.25t_f) \leq 102.2314,$$

$$C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f) \leq 102.2314,$$

$$C_A(0.75t_f), C_B(0.75t_f), C_D(0.75t_f) \leq 102.2314,$$

and 10 defect functions: 9 for discretized state variables and one defect function for final state.

Then the algorithm has to formulate the c-problem with a startpoint matrix

$$\tilde{w} = \begin{pmatrix} 100.000 & 0 & 0 & 8.5000 \\ 71.8132 & 28.5364 & 1.9502 & 8.5000 \\ 59.0719 & 38.8453 & 5.2914 & 8.5000 \\ 53.8719 & 41.4803 & 8.6989 & 8.5000 \end{pmatrix}$$

Table III

RESULTS FOR PROBLEM WITH 4 SHOOTS AND 1 CONTROL FUNCTION IN EACH INTERVAL

time	state variables			control function
	$C_A$	$C_B$	$C_D$	
$[0, 0.25t_f)$	100	0	0	7.1969
$[0.25t_f, 0.5t_f)$	71.2741	28.4493	1.9381	7.7283
$[0.5t_f, 0.75t_f)$	58.3454	38.6533	5.2325	6.6451
$[0.75t_f, t_f)$	52.5205	41.1556	8.5552	6.2900

Figure 3. Problem with 4 shots and 1 control function in each interval.

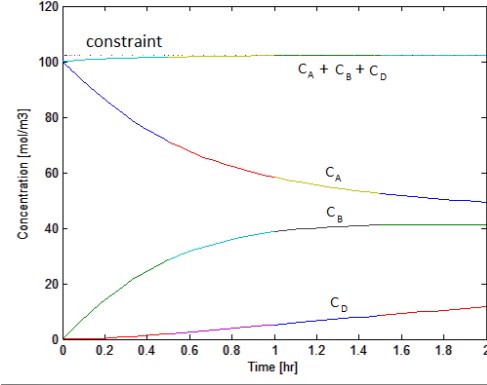


Table IV

PROBLEM WITH 10 SHOOTS AND 1 CONTROL FUNCTION IN EACH INTERVAL

time	state variables			control function
	$C_A$	$C_B$	$C_D$	
$[0, 0.1t_f)$	100	0	0	7.8360
$[0.1t_f, 0.2t_f)$	86.5555	14.2030	0.4044	7.7366
$[0.2t_f, 0.3t_f)$	75.7938	24.6084	1.3562	7.6587
$[0.3t_f, 0.4t_f)$	67.8853	31.6167	2.5723	7.3356
$[0.4t_f, 0.5t_f)$	62.2766	36.0457	3.8956	6.8303
$[0.5t_f, 0.6t_f)$	58.2879	38.6949	5.2485	6.7313
$[0.6t_f, 0.7t_f)$	55.4486	40.1879	6.5948	6.6217
$[0.7t_f, 0.8t_f)$	53.3526	40.9592	7.9195	6.4818
$[0.8t_f, 0.9t_f)$	51.7257	41.2896	9.2160	6.3075
$[0.9t_f, t_f)$	50.3911	41.3540	10.4855	6.1464

As the results we have values of control variables and discretized state variables (Table III).

The results we can see on the Figure 3.

### C. 10 shots and 1 control function in each interval

This problem is bigger than the previous. We have 37 variables:  $3 \cdot 9$  discretized state variables and 10 control variables. Because we have 27 discretized state variables, at the moment we need 27 defect functions. Together with one defect function for final state, we have to consider problem with 28 defect functions.

As the results we have values of control variables and discretized state variables (Table IV). There are results on the Figure 4.

### D. 2 shots and 2 control functions in each interval

We considered this situation, because it is similar to problem with 1 shot and 4 control functions. In both situations control functions contribute 4 degrees of freedom. Now each control

Figure 4. Problem with 10 shots and 1 control function in each interval.

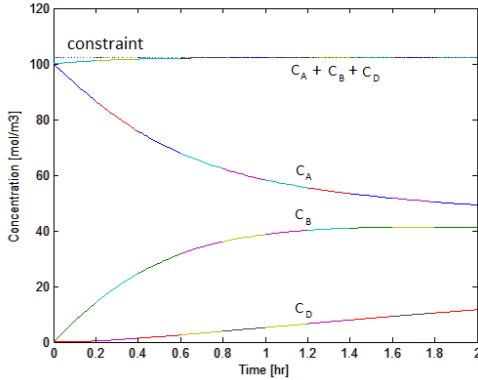
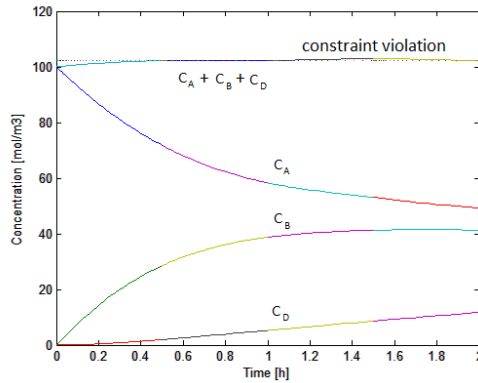


Figure 5. Problem with 2 shots and 2 control functions in each interval.



function consists of two steps, so vector, which represents a solution of each part can have a form  $w_i = (\text{initial values of state variables, control functions})$ . In particular, there are

$$w_0 = (C_A(0), C_B(0), C_D(0), u_{01}, u_{02}),$$

and

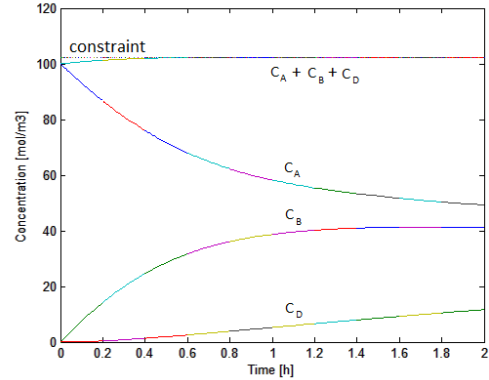
$$w_1 = (C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f), u_{11}, u_{12}).$$

Together there are 3 discretized control variables and 2 control functions. Each control function has two parameters. In all there are 7 variables and 4 defect functions: 3 for discretized state variables and one for the final state.

Now we can repeat the Betts' question: What Can Go Wrong? [3]. The algorithm can influence the selection of discretized state variables and control variables. But we not can be sure, that in each interval, when controls variables are changing, the constraints are satisfied. The obtained solution, which is feasible for c-problem, can violate the constraints of basic discretized problem (Figure 5). Then it should be regularized.

An important topic. We have less variables in this problem, but we lose a possibility of parallel computations too.

Figure 6. Problem with 2 shots and 5 control function in each interval.



### E. 2 shots and 5 control functions in each interval

In this problem we have 3 discretized state variables too. But we have to consider situation with 10 control functions and 4 defect functions: 3 for discretized state variables and one for the final state. We can have the same problem like in the previous example (Figure 6).

Methods, which have more than 1 control variable in each time interval give good results for BVP's.

### F. Comparison of results

The results are summarized in Table V.

We can see, that methods, which have the same number of time intervals as number of control functions give results more similar to the results of the basic problem.

All experiments were performed on the processor Intel(R) Core(TM) i5 CPU 2.67 GHz.

Important questions are the CPU performance time and a parallel computing possibility. Especially, when shooting method is used, one is able to use more processors. Although  $m$  processors were used, the computations performance time would not be  $m$  times smaller. Before starting the parallel computing one has to have in mind a communications overhead. Running 2, 3 or 4 local workers as the clients on the same machine needs about 0.35 CPU time.

When can we expect a performance time improvement? The performance time depends on number of functions evaluations and communications time with local workers. If we denote  $x$  as CPU time processing on one processor,  $y$  - number of function evaluations and  $n$  - number of local workers, one can compute number of local workers are needed

$$\frac{1}{n} \cdot x + 0.35 \cdot y \leq x.$$

If  $0.35 \cdot y \leq x$ , then

$$\frac{x}{x - 0.35 \cdot y} \leq n.$$

In the presented example parallel computing should not improve the performance time. It would be useful in more complex applications.

Table V  
COMPARISON OF RESULTS

Problem		Results			
shots	control functions	CPU time	fun eval	c-prob	basic prob
2	1	14.2429	303	11.6416	11.6536
4	1	80.2313	1623	11.6998	11.6998
2	2	23.7434	522	11.7815	11.6867
10	1	701.6457	8148	11.7160	11.7160
2	5	104.8015	1391	11.8278	11.6885

## V. CONCLUSION

Modified multiple shooting algorithms for the optimization of complex DAE systems are proposed. They are aimed at the determination of a suboptimal feasible solution of a high practical meaning. To this end an initial feasible solution is found by the c-conversion of the basic discretized multipoint problem and the analysis of the results of the consecutive shots of the system trajectory. The employment of the feasible-SQP approach dealing with compatible QP subproblems is guaranteed. The optimized solution can be regularized by the bound-constrained trust-region and inexact Newton method to ensure a high degree of its applicability.

## REFERENCES

- [1] S. Bellavia and B. Morini, "Subspace trust-region methods for large bound-constrained nonlinear equations," *SIAM Journal on Numerical Analysis*, vol.44., pp.1535-1555, 2006.
- [2] S. Bellavia, M. Macconi, and B. Morini, "An affine scaling trust-region approach to bound-constrained nonlinear systems," *Applied Numerical Mathematics*, vol. 44, pp. 257-280, 2003.
- [3] J.T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Philadelphia: SIAM, 2010.
- [4] L.T. Biegler, *Nonlinear Programming. Concepts, Algorithms, and Chemical Processes*. Philadelphia: SIAM, 2010.
- [5] K.E. Brenan, S.L. Campbell, and L.R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Philadelphia: SIAM, 1996.
- [6] L.Chen, Y. Wang, and G. He, "A feasible active set QP-free method for nonlinear programming," *SIAM Journal on Optimization*, vol.17, pp. 401-429, 2006.
- [7] E. Gelman and J. Mandel, "On multilevel iterative methods for optimization problems," *Mathematical Programming*, vol. 48, pp. 1-17, 1990.
- [8] J. Herskovits, "A two-stage feasible directions algorithm for nonlinear constrained optimization," *Mathematical Programming*, vol.36, pp.19-38, 1986.
- [9] Y. J. Huang, G. V. Reklaitis, V. Venkatasubramanian, "Model decomposition based method for solving general dynamic optimization problems," *Computers and Chemical Engineering*, vol. 26, pp. 863-873, 2002
- [10] J.-B. Jian, C-M. Tang, Q.-J. Hu, H.-Y. Zheng, "A feasible descent SQP algorithm for general constrained optimization without strict complementarity," *Journal of Computational and Applied Mathematics*, vol. 180, pp. 391-412, 2005. pp.1531-1542, 1986.
- [11] D.B. Leineweber, I. Bauer, H.G. Bock, J.P. Schlöder, "An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part 1: theoretical aspect," *Computers and Chemical Engineering*, vol. 27, pp. 157-166, 2003.
- [12] D.Q. Mayne and E. Polak, "Feasible directions algorithms for optimization problems with equality and inequality constraints," *Mathematical Programming*, vol.11, pp.67-80, 1976.
- [13] R. Pytlak, "Optimal control of differential-algebraic equations of higher index, Part 1: First-order approximations," *J. Optimiz. Theory Appl.*, vol. 134, pp. 61-75, 2007.
- [14] M.J. Śmiałowski, "Inexact quasi-Newton global convergent method for solving constrained nonsmooth equations," *International Journal of Computer Mathematics*, vol. 84, pp.1757-1770, 2007.
- [15] Z. Zhu, W. Zhang, and Z. Geng, "A feasible SQP method for nonlinear programming," *Applied Mathematics and Computation*, vol. 215, pp.3956-3969, 2010

# On the implementation of stream ciphers based on a new family of algebraic graphs

Jakub Kotorowicz  
 Maria Curie-Skłodowska University  
 Institute of Mathematics,  
 pl. M. Curie-Skłodowskiej 5,  
 20-031 Lublin, Poland.  
 Email: jkotor@hektor.umcs.lublin.pl

Urszula Romańczuk\*  
 Maria Curie-Skłodowska University  
 Institute of Mathematics,  
 pl. M. Curie-Skłodowskiej 5,  
 20-031 Lublin, Poland.  
 Email: urszula\_romanczuk@yahoo.pl

Vasyl Ustimenko\*  
 Maria Curie-Skłodowska University  
 Institute of Mathematics,  
 pl. M. Curie-Skłodowskiej 5,  
 20-031 Lublin, Poland.  
 Email: vasy1@hektor.umcs.lublin.pl

## I. ON THE FAMILIES OF DIRECTED GRAPHS OF LARGE GIRTH

**T**HE READER can find the missing theoretical definitions on directed graphs in [8]. Let  $\Phi$  be an irreflexive binary relation over the set  $V$ , i.e.,  $\Phi \subset V \times V$  and for each  $v$  the pair  $(v, v)$  is not an element of  $\Phi$ .

We say that  $u$  is the neighbour of  $v$  and write  $v \rightarrow u$  if  $(v, u) \in \Phi$ . We use the term *balanced binary relation graph* for the graph  $\Gamma$  of an irreflexive binary relation  $\phi$  over a finite set  $V$  such that for each  $v \in V$  the sets  $\{x | (x, v) \in \phi\}$  and  $\{x | (v, x) \in \phi\}$  have the same cardinality. It is a directed graph without loops and multiple edges. We say that a balanced graph  $\Gamma$  is  $k$ -regular if for each vertex  $v \in \Gamma$  the cardinality of  $\{x | (v, x) \in \phi\}$  is  $k$ .

Let  $\Gamma$  be the graph of binary relation. The *path* between vertices  $a$  and  $b$  is the sequence  $a = x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_s = b$  of length  $s$ , where  $x_i, i = 0, 1, \dots, s$  are distinct vertices.

We say that the pair of paths  $a = x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_s = b, s \geq 1$  and  $a = y_0 \rightarrow y_1 \rightarrow \dots \rightarrow y_t = b, t \geq 1$  form an  $(s, t)$ -commutative diagram  $O_{s,t}$  if  $x_i \neq y_j$  for  $0 < i < s, 0 < j < t$ . Without loss of generality we assume that  $s \geq t$ .

We refer to the number  $\max(s, t)$  as the rank of  $O_{s,t}$ . It is greater than or equal to 2, because the graph does not contain multiple edges.

Notice that the graph of antireflexive binary relation may have a directed cycle  $O_s = O_{s,0}: v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_{s-1} \rightarrow v_0$ , where  $v_i, i = 0, 1, \dots, s-1, s \geq 2$  are distinct vertices.

We will count directed cycles as commutative diagrams.

For the investigation of commutative diagrams we introduce the *girth indicator*  $gi$ , which is the minimal value for  $\max(s, t)$  for parameters  $s, t$  of a commutative diagram  $O_{s,t}, s+t \geq 3$ . The minimum is taken over all pairs of vertices  $(a, b)$  in the digraph. Notice that two vertices  $v$  and  $u$  at distance is less than  $gi$  are connected by the unique path from  $u$  to  $v$  of length is less than  $gi$ .

We assume that the *girth*  $g(\Gamma)$  of a directed graph  $\Gamma$  with the girth indicator  $d+1$  is  $2d+1$  if it contains a commutative

diagram  $O_{d+1,d}$ . If there are no such diagrams we assume that  $g(\Gamma)$  is  $2d+2$ .

In case of a symmetric binary relation  $gi = d$  implies that the girth of the graph is  $2d$  or  $2d-1$ . It does not contain an even cycle  $2d-2$ . In the general case  $gi = d$  implies that  $g \geq d+1$ . So in the case of the family of graphs with unbounded girth indicator, the girth is also unbounded. We also have  $gi \geq g/2$ .

In the case of symmetric irreflexive relations the above mentioned general definition of the girth agrees with the standard definition of the girth of a simple graph, i.e., the length of its minimal cycle.

We will use the term *the family of graphs of large girth* for the family of balanced directed regular graphs  $\Gamma_i$  of degree  $k_i$  and order  $v_i$  such that  $gi(\Gamma_i) \geq c \log_{k_i}(v_i)$ , where  $c$  is a constant independent of  $i$ .

It follows from the definition that  $g(\Gamma_i) \geq c \log_{k_i}(v_i)$  for an appropriate constant  $c$ . So, it agrees with the well known definition for the case of simple graphs.

The diameter of the strongly connected digraph [8] is the minimal length  $d$  of the shortest directed path  $a = x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_d = b$  between two vertices  $a$  and  $b$ . Recall that a graph is  $k$ -regular, if each vertex of  $G$  has exactly  $k$  edges. Let  $F$  be the infinite family of  $k_i$  regular graphs  $G_i$  of order  $v_i$  and diameter  $d_i$ . We say that  $F$  is a family of small world graphs if  $d_i \leq C \log_{k_i}(v_i), i = 1, \dots$  for some constant  $C$  independent of  $i$ . The reader can find the definition of small world simple graphs and related explicit constructions in [4]. For the studies of small world simple graphs without small cycles see [12], [16].

## II. ON THE $K$ -THEORY OF AFFINE GRAPHS WITH INCREASING GIRTH INDICATOR AND ITS CRYPTOGRAPHICAL MOTIVATIONS

We use the concepts of [19] here, where the reader can find additional examples of affine graphs over rings or fields.

Let  $K$  be a commutative ring. A *directed algebraic graph*  $\phi$  over  $K$  consists of two things, such as the *vertex set*  $Q$  being a quasiprojective variety over  $K$  of nonzero dimension and the *edge set* being a quasiprojective variety  $\phi$  in  $Q \times Q$ . We assume that  $(x\phi y)$  means  $(x, y) \in \phi$ .

\* Research supported by a project "Human - The Best Investment". The project is co-funded from the sources of the European Union within the European Social Fund.

The graph  $\phi$  is *balanced* if for each vertex  $v \in Q$  the sets  $\text{Im}(v) = \{x | v\phi x\}$  and  $\text{Out}(v) = \{x | x\phi v\}$  are quasiprojective varieties over  $K$  of the same dimension.

The graph  $\phi$  is *homogeneous* (or  $(r, s)$ -homogeneous) if for each vertex  $v \in Q$  the sets  $\text{Im}(v) = \{x | v\phi x\}$  and  $\text{Out}(v) = \{x | x\phi v\}$  are quasiprojective varieties over  $F$  of fixed nonzero dimensions  $r$  and  $s$ , respectively.

In the case of *balanced homogeneous algebraic graphs* for which  $r = s$  we will use the term  $r$ -homogeneous graph. Finally, a *regular algebraic graph* is a balanced homogeneous algebraic graph over the ring  $K$  if each pair of vertices  $v_1$  and  $v_2$  is a pair of isomorphic algebraic varieties.

Let  $\text{Reg}(K)$  be the totality of regular elements (or nonzero divisors) of  $K$ , i.e., nonzero elements  $x \in K$  such that for each nonzero  $y \in K$  the product  $xy$  is different from 0. We assume that  $\text{Reg}(K)$  contains at least 3 elements. We assume here that  $K$  is finite, thus the vertex set and the edge set are finite and we get a usual finite directed graph.

We apply the term *affine graph* for the regular algebraic graph such that its vertex set is an affine variety in the Zarisski topology.

Let  $G$  be an  $r$ -regular affine graph with vertex set  $V(G)$ , such that  $\text{Out}v$ ,  $v \in V(G)$  is isomorphic to the variety  $R(K)$ . Let the variety  $E(G)$  be its arrow set (a binary relation in  $V(G) \times V(G)$ ). We use the standard term *perfect algebraic colouring of edges* for the polynomial map  $\rho$  from  $E(G)$  onto the set  $R(K)$  (the set of colours) if for each vertex  $v$  different output arrows  $e_1 \in \text{Out}(v)$  and  $e_2 \in \text{Out}(v)$  have distinct colours  $\rho(e_1)$  and  $\rho(e_2)$  and the operator  $N_\alpha(v)$  of taking the neighbour  $u$  of vertex  $v$  ( $v \rightarrow u$ ) is a polynomial map of the variety  $V(G)$  into itself.

We will use the term *rainbow-like colouring* in the case when the perfect algebraic colouring is a bijection. Let  $\text{dirg}(G)$  be a directed girth of the graph  $G$ , i.e., the minimal length of a directed cycle in the graph. Obviously  $\text{gi}(G) \leq \text{dirg}(G)$ .

Studies of infinite families of directed affine algebraic digraphs over commutative rings  $K$  of large girth with the rainbow-like colouring is a nice but difficult mathematical problem. Good news is that such families do exist. In the next section we consider an example of such a family for each commutative ring with more than 2 regular elements.

At the end of section we consider cryptographical motivations for studies of such families.

1) Let  $G$  be a finite group and  $g \in G$ . The discrete logarithm problem for  $G$  is about finding a solution for the equation  $g^x = b$  where  $x$  is an unknown positive integer. If the order  $|g| = n$  is known we can replace  $G$  with a cyclic group  $C_n$ . So we may assume that the order of  $g$  is sufficiently large to make the computation of  $n$  unfeasible. For many finite groups the discrete logarithm problem is  $NP$  complete.

Let  $K$  be a finite commutative ring and  $M$  be an affine variety over  $K$ . Then the Cremona group  $C(M)$  of all polynomial automorphisms of the variety  $M$  can be large. For example, if  $K$  is a finite prime field  $F_p$  and  $M = F_p^n$  then  $C(M)$  is a symmetric group  $S_{p^n}$ .

Let us consider the family of affine graphs  $G_i(K)$ ,  $i = 1, 2, \dots$  with the rainbow-like algebraic colouring of edges such that  $V(G_i(K)) = V_i(K)$ , where  $K$  is a commutative ring, and the colour sets are algebraic varieties  $R_i(K)$ . Let us choose a constant  $k$ . The operator  $N_\alpha(v)$  of taking the neighbour of a vertex  $v$  corresponding to the output arrow of colour  $\alpha$  are elements of  $C_i = C(V_i(K))$ . We can choose a relatively small number  $k$  to generate  $h = h_i = N_{\alpha_1} N_{\alpha_2} \dots N_{\alpha_k}$  in each group  $C_i$ ,  $i = 1, 2, \dots$

Let us assume that the family of graphs  $G_i(K)$  is the family of graphs of increasing girth. It means that the girth indicator  $\text{gi}_i = \text{gi}(G_i(K))$  and the parameter  $\text{dirg}_i = \text{dirg}(G_i(K))$  are growing with the growth of  $i$ . Notice that  $|h_i|$  is bounded below by  $\text{dirg}_i/k$ . So there is a  $j$  such that for  $i \geq j$  the computation of  $|h_i|$  is impossible. In fact, the fastest grow of girth indicator will be in the case of a family of large girth. Finally we can take the base  $g = T^{-1}h_jT$  where  $T$  is a chosen element of  $C_j$  to hide the graph up to conjugation. We may use some package of symbolic computations to express the polynomial map  $g$  via the list of polynomials in many unknowns. For example, if  $V_j(K)$  is a free module  $K^n$  then we can write  $g$  in a public mode fashion

$$x_1 \rightarrow g_1(x_1, x_2, \dots, x_n), x_2 \rightarrow g_2(x_1, x_2, \dots, x_n), \dots, x_n \rightarrow g_n(x_1, x_2, \dots, x_n).$$

The symbolic map  $g$  can be used for the Diffie - Hellman *key exchange protocol* (see [4] for the details). Let Alice and Bob be correspondents. Alice computes the symbolic map  $g$  and send it to Bob via an open channel. So the variety and the map are known for the adversary (Cezar).

Let Alice and Bob choose natural numbers  $n_A$  and  $n_B$ , respectively.

Bob computes  $g^{n_B}$  and sends it to Alice, who computes  $(g^{n_B})^{n_A}$ , while Alice computes  $g^{n_A}$  and sends it to Bob, who is getting  $(g^{n_A})^{n_B}$ . The common information is  $g^{n_A n_B}$  given in "public mode fashion".

Bob can be just a public user (no information on the way in which the map  $g$  was created), so he and Cezar make computations much slower than Alice who has the decomposition  $g = T^{-1}N_{\alpha_1}N_{\alpha_2} \dots N_{\alpha_k}T$ .

We may modify slightly the Diffie - Hellman protocol using the action of the group on the variety. Alice chooses a rather short password  $\alpha_1, \alpha_2, \dots, \alpha_k$ , computes the public rules for the encryption map  $g$  and sends them to Bob via an open channel together with some vertex  $v \in V_j(K)$ . Then Alice and Bob choose natural numbers  $n_A$  and  $n_B$ , respectively.

Bob computes  $v_B = g^{n_B}(v)$  and sends it openly to Alice, who computes  $(g^{n_A})(v_B)$ , while Alice computes  $v_A = g^{n_A}(v)$  and sends it to Bob, who receives  $(g^{n_B})(v_A)$ .

The common information is the vertex  $g^{n_A n_B}(v)$ .

In both cases Cezar has to solve one of the equations  $E^{n_B}(u_A) = z$  or  $E^{n_A}(u_B) = w$  for unknowns  $n_B$  or  $n_A$ , where  $z$  and  $w$  are known points of the variety.

2) We can construct the *public key* map in the following manner:

The key holder (Alice) chooses the variety  $V_j(K)$  and the sequence  $\alpha_1, \alpha_2, \dots, \alpha_t$  of length  $t = t(j)$  to determine the



encryption map  $g$  as above. Let  $\dim(V_j(K)) = n = n(j)$  and each element of the variety be determined by independent parameters  $x_1, x_2, \dots, x_n$ . Alice presents the map in the form of public rules, such as

$$x_1 \rightarrow f_1(x_1, x_2, \dots, x_n), x_2 \rightarrow f_2(x_1, x_2, \dots, x_n), \dots, x_n \rightarrow f_n(x_1, x_2, \dots, x_n).$$

We can assume (at least theoretically) that the public rule depending on parameter  $j$  is applicable to the encryption of a potentially infinite text (the parameter  $t$  is a linear function on  $j$  now).

For the computation she may use the Gröbner base technique or alternative methods, special packages for the symbolic computation (popular "Mathematica" or "Maple", package "Galois" for "Java" as well special fast symbolic software). So Alice can use the decomposition of the encryption map into  $T^{-1}$ , maps of kind  $N_\alpha$  and  $T$  to encrypt fast. For the decryption she can use the inverse graph  $G_j(K)^{-1}$  for which  $VG_j(K)^{-1} = VG_j(K)$  and vertices  $w_1$  and  $w_2$  are connected by an arrow if and only if  $w_2$  and  $w_1$  are connected by an arrow in  $G_j(K)$ . Let us assume that colours of  $w_1 \rightarrow w_2$  in  $G_j(K)^{-1}$  and  $w_2 \rightarrow w_1$  in  $G_j(K)$  are of the same colour. Let  $N'_\alpha(x)$  be the operator taking the neighbour of vertex  $x$  in  $G_j(K)^{-1}$  of colour  $\alpha$ . Then Alice can decrypt applying sequentially  $T, N'_{\alpha_t}, N'_{\alpha_{t-1}}, \dots, N'_{\alpha_1}$  and  $T^{-1}$  to the ciphertext. So the decryption and the encryption for Alice take the same time. She can use a numerical program to implement her symmetric algorithm.

Bob can encrypt with the public rule but for a decryption he needs to invert the map. Let us consider the case  $t_j = kl$ , where  $k$  is a small number and the sequence  $\alpha_1, \alpha_2, \dots, \alpha_{t_j}$  has the period  $k$ , and the transformation  $h = T^{-1}N_{\alpha_1}N_{\alpha_2} \dots N_{\alpha_k}T$  is known for Bob in the form of public key mode. In such a case a problem to find the inverse for  $g$  is equivalent to a discrete logarithm problem with the base  $h$  in the related Cremona group of all polynomial bijective transformations.

Of course for further cryptanalysis we need to study the information on possible divisors of the order of the base of the related discrete logarithm problem, alternative methods to break the encryption. In the next section the family of digraphs  $RE_n(K)$  will be described.

3) We may study the security of the private key algorithm used by Alice in the algorithm of the previous paragraph but with a parameter  $t$  bounded by the girth indicator of graph  $G_j(K)$ . In this case different keys produce distinct ciphertexts from the chosen plaintext. We prove that if the adversary has no access to plaintexts then he can break the encryption via the brute-force search via all keys from the key space. The encryption map has no fixed points.

### III. ON THE FAMILY OF AFFINE DIGRAPH OF INCREASING GIRTH OVER COMMUTATIVE RINGS

E. Moore used the term *tactical configuration* of order  $(s, t)$  for biregular bipartite simple graphs with bidegrees  $s + 1$  and  $r + 1$ . It corresponds to the incidence structure with the point

set  $P$ , the line set  $L$  and the symmetric incidence relation  $I$ . Its size can be computed as  $|P|(s + 1)$  or  $|L|(t + 1)$ .

Let  $F = \{(p, l) | p \in P, l \in L, pIl\}$  be the totality of flags for the tactical configuration with partition sets  $P$  (point set) and  $L$  (line set) and an incidence relation  $I$ . We define the following irreflexive binary relation  $\phi$  on the set  $F$ :

Let  $(P, L, I)$  be the incidence structure corresponding to regular tactical configuration of order  $t$ .

Let  $F_1 = \{(l, p) | l \in L, p \in P, lIp\}$  and  $F_2 = \{(l, p) | l \in L, p \in P, lIp\}$  be two copies of the totality of flags for  $(P, L, I)$ . Brackets and parentheses allow us to distinguish elements from  $F_1$  and  $F_2$ . Let  $DF(I)$  be the directed graph (double directed flag graph) on the disjoint union of  $F_1$  with  $F_2$  defined by the following rules:

$$(l_1, p_1) \rightarrow [l_2, p_2] \text{ if and only if } p_1 = p_2 \text{ and } l_1 \neq l_2,$$

$$[l_2, p_2] \rightarrow (l_1, p_1) \text{ if and only if } l_1 = l_2 \text{ and } p_1 \neq p_2.$$

Below we consider the family of graphs  $A(k, K)$ , where  $k > 5$  is a positive integer and  $K$  is a commutative ring. Such graphs are disconnected and their connected components were investigated in [17] (for the case when  $K$  is a finite field  $F_q$  see [7]).

Let  $P$  and  $L$  be two copies of Cartesian power  $K^N$ , where  $K$  is the commutative ring and  $N$  is the set of positive integer numbers. Elements of  $P$  will be called *points* and those of  $L$  *lines*.

To distinguish points from lines we use parentheses and brackets. If  $x \in V$ , then  $(x) \in P$  and  $[x] \in L$ . It will also be advantageous to adopt the notation for co-ordinates of points and lines introduced in [20] for the case of a general commutative ring  $K$ :

$$(p) = (p_{0,1}, p_{1,1}, p_{1,2}, p_{2,2}, p_{2,3}, \dots, p_{i,i}, p_{i,i+1}, \dots),$$

$$[l] = [l_{1,0}, l_{1,1}, l_{1,2}, l_{2,2}, l_{2,3}, \dots, l_{i,i}, l_{i,i+1}, \dots].$$

The elements of  $P$  and  $L$  can be thought of as infinite ordered tuples of elements from  $K$ , such that only a finite number of components are different from zero.

We now define an incidence structure  $(P, L, I)$  as follows. We say that the point  $(p)$  is incident with the line  $[l]$ , and we write  $(p)I[l]$ , if the following relations between their co-ordinates hold:

$$l_{i,i} - p_{i,i} = l_{1,0}p_{i-1,i}$$

$$l_{i,i+1} - p_{i,i+1} = l_{i,i}p_{0,1}$$

This incidence structure  $(P, L, I)$  we denote as  $A(K)$ . We identify it with the bipartite *incidence graph* of  $(P, L, I)$ , which has the vertex set  $P \cup L$  and the edge set consisting of all pairs  $\{(p), [l]\}$  for which  $(p)I[l]$ .

For each positive integer  $k \geq 2$  we obtain an incidence structure  $(P_k, L_k, I_k)$  as follows. First,  $P_k$  and  $L_k$  are obtained from  $P$  and  $L$  respectively by simply projecting each vector onto its  $k$  initial coordinates with respect to the above order. The incidence  $I_k$  is then defined by imposing the first  $k - 1$  incidence equations and ignoring all others. The incidence graph corresponding to the structure  $(P_k, L_k, I_k)$  is denoted by  $A(k, K)$ .

For each positive integer  $k \geq 2$  we consider the standard graph homomorphism  $\phi_k$  of  $(P_k, L_k, I_k)$  onto  $(P_{k-1}, L_{k-1}, I_{k-1})$  defined on  $L_k$  by simply projection of each vector from  $P_k$  and  $L_k$  onto its  $k-1$  initial coordinates with respect to the above order.

Let  $DA_n(K)$  ( $DA(K)$ ) be the double directed graph of the bipartite graph  $A(n, K)$  ( $A(K)$ , respectively). Remember, that we have the arc  $e$  of kind  $(l^1, p^1) \rightarrow [l^2, p^2]$  if and only if  $p^1 = p^2$  and  $l^1 \neq l^2$ . Let us assume that the colour  $\rho(e)$  of the arc  $e$  is  $l_{1,0}^1 - l_{1,0}^2$ .

Recall, that we have the arc  $e'$  of kind  $[l^2, p^2] \rightarrow (l^1, p^1)$  if and only if  $l^1 = l^2$  and  $p^1 \neq p^2$ . Let us assume that the colour  $\rho(e')$  of arc  $e'$  is  $p_{1,0}^1 - p_{1,0}^2$ . It is easy to see that  $\rho$  is a perfect algebraic colouring.

If  $K$  is finite, then the cardinality of the colour set is  $(|K| - 1)$ . Let  $\text{Reg}K$  be the totality of regular elements, i.e., not zero divisors. Let us delete all arrows with colour, which are zero divisors. We will obtain a new graph  $RA_n(K)$  ( $RA(K)$ ) with the induced colouring into colours from the alphabet  $\text{Reg}(K)$ . The vertex set for the graph  $DA_n(K)$  consists of two copies  $F_1$  and  $F_2$  of the edge set for  $A(n, K)$ .

If  $K$  is finite, then the cardinality of the colour set is  $(|K| - 1)$ . Let  $\text{Reg}K$  be the totality of regular elements, i.e., non-zero divisors. Let us delete all arrows with colour, which are zero divisors. We can show that a new infinite affine graph  $A(K)$  does not contain cycles (see [9]). This means that the directed graph  $RA(K)$  does not contain commutative diagrams and the digraphs  $RA_n(K)$  form a family of digraphs with increasing girth indicator. In fact computer simulations support the following assertion.

CONJECTURE: Graphs  $RA_n(K)$  form a family of digraphs of large girth.

#### IV. ON THE IMPLEMENTATION OF THE STREAM CIPHER BASED ON $RA_t(K)$

The set of vertices of the graph  $RA_n(K)$  is a union of two copies of a free module  $K^{n+1}$ . So the Cremona group of the variety is the direct product of  $C(K^{n+1})$  with itself, expanded by polarity  $\pi$ . In the simplest case of a finite field  $F_p$ , where  $p$  is a prime number,  $C(F_p)$  is a symmetric group  $S_{p^{n+1}}$ . The Cremona group  $C(K^{n+1})$  contains the group of all affine invertible transformations, i.e., transformation of kind  $x \rightarrow xA + b$ , where  $x = (x_1, x_2, \dots, x_{n+1}) \in C(K^{n+1})$ ,  $b = (b_1, b_2, \dots, b_{n+1})$  is a chosen vector from  $C(K^{n+1})$  and  $A$  is a matrix of a linear invertible transformation of  $K^{n+1}$ .

The graph  $RA_n(K)$  is a bipartite directed graph. We assume that the plaintext  $K^{n+1}$  is a point  $(p_1, p_2, \dots, p_{n+1})$ . We choose two affine transformations  $T_1$  and  $T_2$  as linear transformation of kind  $p_1 \rightarrow p_1 + a_1p_2 + a_2p_3 + \dots + a_np_{n+1}$ . We will follow a general scheme, so Alice and Bob compute chosen  $T_1$  and  $T_2$ , and choose a string  $(\beta_1, \beta_2, \dots, \beta_l)$  of colours for  $RE_n(K)$ , such that  $\beta_i \neq -\beta_{i+1}$  for  $i = 1, 2, \dots, l-1$ . They will use  $N_l = N_{\beta_1} \times N_{\beta_2} \times \dots \times N_{\beta_l}$ . Recall that  $N_\alpha$ ,  $\alpha \in \text{Reg}(K)$  is the operator of taking the neighbour of the vertex  $v$  alongside the arrow with the colour  $\alpha$  in the graph  $RA_n(K)$ .

Alice and Bob keep chosen parameters  $T_1, (\beta_1, \beta_2, \dots, \beta_l)$  and  $T_2$  secret and use the encryption map  $g$  which is the composition of  $T_1, N_l$  and  $T_2$ .

In the case of  $RA_n(K)$  the degree of transformation  $N_l$  is 3, independent of the choice of length  $l$  like in the case of graphs  $D(n, K)$  [9]. We can prove that for arbitrary key the encryption map is a cubical polynomial map of the free module  $K^{n+1}$  onto itself.

In our computer implementations we used  $T_1$  and  $T_2$  of kind  $p_1 \rightarrow p_1 + a_1p_2 + a_2p_3 + \dots + a_np_{n+1}$ , where all  $a_i$  are not zero divisors.

#### V. ON THE COMPARISON OF PRIVATE KEYS BASED ON $A(n, K)$ AND $D(n, K)$

In the paper [5] we have implemented the stream cipher based on graphs  $D(n, K)$  with vertex set the union of two copies of the free module  $K^n$ . The time of execution of the encryption map and its mixing properties and comparison with other private keys (DES and RC4 are considered in [5] for cases of rings  $Z_2^8, Z_2^{16}$  and  $Z_2^{32}$ . The reader can find speedy evaluation for cases of rings  $Z_2^8, Z_2^{16}$  and  $Z_2^{32}$  in [16]. Recently, private keys based on  $D(n, F_q)$ ,  $q = 2^8, q = 2^{16}$  and  $q = 2^{32}$  were implemented (see [13], where time evaluation is presented).

The mixing properties of  $D(n, Z_2^m)$ ,  $m = 8, 16, 32$  based encryption in combination with special affine transformations were investigated in [20]. If we change one character of the string  $\alpha_1, \alpha_2, \dots, \alpha_s$  (the graphical part of the key related to the pass of the graph) then at least 97 percent of characters of the ciphertext will be changed. If we change one character of the plaintext then again at least 97 percent of the characters of the ciphertext will be changed.

We present at the conference similar results of statistics for mixing properties of an  $A(m, Z_q)$  based stream cipher.

We can see that graphs the  $A(n, K)$  and  $D(n, K)$  are given by equations which use  $n-1$  additions (or subtractions) and multiplications. So algorithms based on these graphs or corresponding digraphs have the same speed evaluations.

Graphs  $A(n, Z_m)$ ,  $m > 2$  are connected but  $D(n, K)$  are not. It means that if we fixed affine maps, then for each pair of vectors  $v_1$  and  $v_2$  from the plainspace there is a string  $\alpha_1, \alpha_2, \dots, \alpha_s$  such the corresponding  $A(n, Z_m)$  based encryption map converts the plaintext  $v_1$  into the ciphertext  $v_2$ . The reader can find some theoretical results on  $A(n, K)$  in [17].

##### A. Evaluation of the order of encryption map

We assume that the product of our affine transformations is the identity. So the order of the encryption map is the same with  $N = N_{\alpha_1} N_{\alpha_2} \dots N_{\alpha_s}$ . We assume that  $s$  is even and our string is obtained by repetition of the word  $\alpha_1, \alpha_2$ , where  $\alpha_1 + \alpha_2 \in \text{Reg}(K)$ . So the security of our encryption is related to the discrete logarithm problem with base  $b = N_{\alpha_1} N_{\alpha_2}$ . It turns out that in cases of  $K = Z_n$ ,  $n > 2$  the order of  $b$  does not depend on the choice of  $\alpha_1$  and  $\alpha_2$ .

B. Case of primes

We have run computer tests, to measure the length of the cycles generated by powers of  $b$  for graphs  $A(n, K)$  with different  $n$ , and different  $K$  (cycles of permutation  $b$  acting on  $K^n$ ). Table I shows these results for the first few prime numbers  $p$  ( $K = Z_p$ ). Each test was repeated at least 20 times, every time with a random start point, and random  $(\alpha_1, \alpha_2)$  parameter.

TABLE I  
CYCLE LENGTH FOR  $K = Z_q$ , WHERE  $q$  IS PRIME

$p \setminus n$	4	10	30	50	100	200	400	600	1000
3	9	27	81	81	243	243	729	729	2187
5	5	25	125	125	125	625	625	625	3125
7	7	49	49	343	343	343	2041	2041	2041
11	11	11	121	123	121	1331	1331	1331	1331

It is easy to see that the cycle length is always a power of the prime number  $p$ . Another property is that cycle length does not depend on starting point, nor parameters  $(\alpha_1, \alpha_2)$ . This property does not hold for  $p = 2$ . In that case the cycle length is always a power of 2, but for the same  $n$  we have different results depending on start point  $x$ , and  $(\alpha_1, \alpha_2)$ .

C. Case of composite numbers

TABLE II  
CYCLE LENGTH FOR SOME COMPOSITE NUMBERS  $q$

$q \setminus n$	4	10	30	50	100	200	400
4	16	32	64	128	256	512	1024
6	72	432	2592	5184	31104	62208	
8	32	64	128	256	512	1024	2048
9	27	81	243	243	729	729	2187
15	45	675	10125	10125	30375	151875	455625

TABLE III  
CYCLE LENGTH FOR  $q = 15$ , CASE OF THE  $A(n, Z_q)$

$n_{MIN}$	$n_{MAX}$	cycle length
4	4	45
5	8	225
9	24	675
25	26	3375
27	80	10125
81	120	30375
140	240	151875
260	620	455625
640	720	2278125
760		6834375

The comparison of cycles in cases  $A(n, K)$  and  $D(n, K)$  encryption demonstrates big advantage of  $A(n, K)$ . The typical example is below.

VI. CONCLUSION, ON THE IMPORTANCE OF NUMERICAL ALGORITHM FOR EVALUATION OF SYMBOLIC CRYPTOGRAPHICAL TOOLS

As we mentioned above the private key algorithm based on graphs  $A(n, K)$  turns out to be good stream cipher. It compares well with  $D(n, K)$  based encryption.

TABLE IV  
CYCLE LENGTH FOR  $q = 15$ , CASE OF  $D(n, Z_{15})$  ENCRYPTION

$n_{MIN}$	$n_{MAX}$	cycle length
4	7	45
8	17	225
18	53	675
54	65	3375
150	249	10125
250	299	30375
300	649	151875
650	1000	455625

On another hand this algorithm is a private decryption tool for corresponding symbolic public key algorithms. Encryption map  $g$  with arbitrarily chosen password is a cubic polynomial map. All powers of  $g$  are cubic maps, so cyclic group generated by  $g$  can be used for the symbolic key exchange protocol. Studies of the properties of the stream cipher are crucial for the evaluation of the main parameters of symbolic algorithms because the speed of symbolic computations is much slower in comparison with our numerical algorithm.

Usually the order of nonlinear polynomial map  $g^k$  from Cremona group (composition of  $g$  with itself, corresponding to permutation  $\pi^k$ ) is growing with the growth of  $k$ . The computation of the order  $t$  of "pseudorandom"  $g$  is a difficult task. Really, if  $t$  is known then the inverse map for  $g$  is  $g^{t-1}$ , but the best known algorithm of finding  $g^{-1}$  has complexity  $d^{O(n)}$ , where  $d$  is the degree of  $g$ . The efficient general algorithm of finding  $g^{-1}$  is known only in the case when degree of  $g$  is one, i. e.  $g$  is affine map  $xA + b$ , where  $x$  and  $b$  are row vectors from  $V$  and  $A$  is nonsingular square matrix. So there is a serious complexity gap between linearity and nonlinearity.

The discrete logarithm problem (dlp) for the cyclic group generated by "pseudorandom" polynomial map  $g$ , i. e. problem of finding solution for equation  $g^x = b$  looks very hard. If  $x$  is known, then  $g^{t-x} = b^{-1}$ , but the computation of  $b^{-1}$  takes  $d^{O(n)}$ . So in the case of "pseudorandom" polynomial base  $g$  we can use the term *hidden symbolic* discrete logarithm problem, word *hidden* is used because the order  $t$  of cyclic group is unknown, *symbolic* is used because generation of polynomial maps  $g$  and  $b$  can be done via tools of symbolic computations (popular "Maple" or "Mathematica" operating on polyomial maps or special fast programs of Computer Algebra). Certainly the choice of the nonlinear base  $g$  for the dlp for  $C(K^n)$  is an important heuristic problem. Obviously one needs to find  $g$  of very large order. If the degree of  $g^x$  is growing linearly with the growth of  $g$ :  $\deg(g(x)) = ax + b$  then  $x$  can be obtained from the linear equation  $ax + d = \deg(b(x))$ . This fact is a motivation of the following concept.

The sequence of subgroups  $G_l$  of  $C(K^l)$ ,  $l \rightarrow \infty$  is a *family of stable groups* if degree of each  $g$ ,  $g \in G_l$  is bounded by constant  $c$  independent on  $l$ . The construction of large stable subgroups  $G_l$  with  $c \geq 2$  of Cremona group is an interesting mathematical task.

There is an easy way to construct stable subgroups via conjugation of  $AGL_l(K)$  (subgroup of all automorphisms of  $K^n$  of degree 1) with the nonlinear polynomial maps  $f_l \in C(K^l)$ . Let us refer to members of such families as pseudolinear groups. Degrees of  $f_l$  and  $f_l^{-1}$  are at least 2. So in case of the use "pseudorandom" polynomials,  $f_i$  such that  $\max(f_l, f_l^{-1})$  is bounded by constant, we obtain a stable family with  $c \geq 4$ . Let  $\tau$  be a Singer cycle from  $AGL_l(F_q)$  of order  $q^n - 1$  ( $K = F_q$ ),  $f_l$  and  $f_l^{-1}$  are nonlinear maps. Then  $g = f_l^{-1}\tau f_l$  looks as appropriate base for the hidden symbolic discrete logarithm problem. Certainly one may use other linear transformations of large order instead of Singer cycle.

So the case of families of stable degree with  $c \in \{2, 3\}$  is the most interesting one.

The new family of large stable subgroups of  $C(K^l)$  over general commutative ring  $K$  containing at least 3 regular elements (non zero divisors). with  $c = 3$  is presented in our paper.

New transformations  $g_n$  of  $K^n$  also form stable subgroups with  $c = 3$ . Computer simulations demonstrate the faster growth of order in comparison with previously known family.

*Remark.* The map of kind  $h = f_1 g_n f_2$ , where  $f_1, f_2 \in C(K^n)$  of bounded degree can be used as a public key algorithm with public rules  $h_1 = h_1(x_1, x_2, \dots, x_n)$ ,  $h_2 = h_2(x_1, x_2, \dots, x_n), \dots, h_n = h_n(x_1, x_2, \dots, x_n)$ .

Computer simulations show that even in the case of distinct affine transformations  $f_1, f_2^{-1}$  the powers  $\delta^k$  of cubic map  $\delta = f_1 g(n) f_2$  have rather sophisticated degrees  $t(n, k)$  which are growing with the growth of parameters  $n$  and  $k$ . In practice the computation of order for  $\delta$  (or its inverse) are much harder in comparison with studies of order for conjugates of  $g_n$ . Notice that in the case of field  $F_p$  invertible affine transformations and  $g(n)$  generate entire group  $S_{p^n}$  (Cremona group of the vector space  $F_p^n$ ).

So we do hope that our method of generation of public key maps produces good approximation of random maps of degree 3 of large order (see [22] for the results of symbolic computations).

#### REFERENCES

- [1] F. Bien, *Constructions of telephone networks by group representations*, Notices Amer. Math. Soc. **3** (1989), 5–22.
- [2] B. Bollobás, *Extremal graph theory*, Academic Press, London, 1978.
- [3] M. Klisowski, V. A. Ustimenko *On the public keys based on the extremal graphs and digraphs*, International Multiconference on Computer Science and Informational Technology, October 2010, Wisla, Poland, CANA Proceedings, 12 pp.
- [4] N. Koblitz, *Algebraic aspects of cryptography*, Algorithms and Computation in Mathematics, vol. 3, Springer, 1998.
- [5] S. Kotorowicz and V. Ustimenko, *On the implementation of cryptological algorithms based on algebraic graphs over some commutative rings*, Condens. Matter Phys. **11** (2008), no. 2(54), 347–360.
- [6] S. Kotorowicz, V. Ustimenko, *On the comparison of mixing properties of stream ciphers based on graphs  $D(n, q)$  and  $A(n, q)$*  (to appear)
- [7] F. Lazebnik, V. A. Ustimenko, and A. J. Woldar, *A new series of dense graphs of high girth*, Bull. Amer. Math. Soc. (N.S.) **32** (1995), no. 1, 73–79.
- [8] R. Ore, *Graph theory*, Wiley, London, 1971.
- [9] U. Romańczuk, V. Ustymenko *On some cryptographic applications of new family of expanding graphs*, Presentation in Conference CECC'2011, Debrecen, Hungary
- [10] T. Shaska and V. Ustimenko, *On some applications of graph theory to cryptography and turbocoding*, Albanian J. Math. **2** (2008), no. 3, 249–255, Proceedings of the NATO Advanced Studies Institute: "New challenges in digital communications".
- [11] T. Shaska, V. Ustimenko, *On the homogeneous algebraic graphs of large girth and their applications*, Linear Algebra Appl. **430** (2009), no. 7, 1826–1837, Special Issue in Honor of Thomas J. Laffey.
- [12] M. Simonovits, *Extremal graph theory*, Selected Topics in Graph Theory 2 (L. W. Beineke and R. J. Wilson, eds.), no. 2, Academic Press, London, 1983, pp. 161–200.
- [13] A. Touzene, V. Ustimenko, Marwa AlRaissi, Imene Boude-liouua, *Performance of Algebraic Graphs Based Stream-Ciphers Using Large Finite Fields* (to appear)
- [14] V. Ustimenko, *Maximality of affine group and hidden graph cryptosystems*, J. Algebra Discrete Math. **10** (2004), 51–65.
- [15] V. Ustimenko, *On the extremal graph theory for directed graphs and its cryptographical applications*, Advances in Coding Theory and Cryptography (T. Shaska, D. W. C. Huffman, Joener, and V. Ustimenko, eds.), Series on Coding Theory and Cryptology, vol. 3, World Scientific, 2007, pp. 181–199.
- [16] V. Ustimenko, *On the extremal regular directed graphs without commutative diagrams and their applications in coding theory and cryptography*, Albanian J. Math. **1** (2007), no. 4, Special issue on algebra and computational algebraic geometry.
- [17] V. Ustimenko, *Algebraic groups and small world graphs of high girth*, Albanian J. Math. **3** (2009), no. 1, 25–33.
- [18] V. Ustimenko, *On the cryptographical properties of extremal algebraic graphs*, Algebraic Aspects of Digital Communications (Tanush Shaska and Engjell Hasimaj, eds.), NATO Science for Peace and Security Series - D: Information and Communication Security, vol. 24, IOS Press, July 2009, pp. 256–281.
- [19] V. A. Ustimenko, *Linguistic Dynamical Systems, Graphs of Large Girth and Cryptography*, Journal of Mathematical Sciences, Springer, vol.140, N3 (2007) pp. 412-434.
- [20] V. Ustimenko and J. Kotorowicz, *On the properties of stream ciphers based on extremal directed graphs*, Cryptography Research Perspective (Roland E. Chen, ed.), Nova Science Publishers, April 2009, pp. 125–141.
- [21] A. Wróblewska, *On some applications of graph based public key*, Albanian J. Math. **2** (2008), no. 3, 229–234, Proceedings of the NATO Advanced Studies Institute: "New challenges in digital communications".
- [22] M. Klisowski, U. Romanczuk, V. Ustimenko, *On the implementation of cubic public keys based on new family of algebraic graphs*, Annales UMCS Informatica Lublin - Polonia, Proceedings of the conference "Cryptography and Security Systems 2011, Nalenczow, September, 2011 (to appear).

# Implementation of Movie-based Matrix Algorithms on OpenMP Platform

Dmitry Vazhenin

Graduate School Department of Information Systems  
University of Aizu  
Tsuruga, Ikki-machi, Aizu-Wakamatsu, Japan  
Email: d8052102@u-aizu.ac.jp

Alexander Vazhenin

Graduate School Department of Information Systems  
University of Aizu  
Tsuruga, Ikki-machi, Aizu-Wakamatsu, Japan  
Email: vazhenin@u-aizu.ac.jp

**Abstract**—The convenience and programmer’s productivity are the main point of visual programming systems and languages. From the other side, the parallel programming is mainly focused on reaching the high performance by optimization of executable code. The Movie-based Programming is based not only on the introduction of special symbols and images with semantic support, but also on a series of images that can present dynamical features of algorithms. The presented paper describes a technique of OpenMP parallelization of Movie-based algorithms in order to obtain the suitable program performance. The results of numerical experiments are also presented showing applicability of the proposed technique including implementation, code validity checking and performance testing.

**Index Terms**—Visual Programming, Movie-based programming, Matrix Computing, Parallel Programming, OpenMP Platform.

## I. INTRODUCTION

ORIGINALLY, the Visual Programming Languages (VPL) are oriented to increase mainly the programmer’s productivity by operating with visual expressions, direct manipulating visual information as well as supporting visual interactions. The most of modern applications of this kind include a big variety of attractive multimedia functions with icons, pictures, animations, sound and other multimedia components allowing reliable understanding as well as effective dialogue with the complex objects. Visual programming languages and tools may be classified according to the type and extent of visual expression used, into icon-based languages, form-based languages and diagram languages as shown in reviews [1], [2]. They provide graphical or iconic elements which can be manipulated by the user in an interactive way according to some specific spatial grammar for program.

Usually, the parallel programming and algorithm design are focused on reaching the high performance by optimization of executable code as shown, for example, in [3]-[4]. This may contradict with the VPLs in which convenience and programmer’s productivity are the main point of system requirements. For example, a graphical toolkit is described in [5] consisting of exploratory tools and estimation tools which allow the programmer to navigate through complex distributions and to obtain graphical ratings with respect to load distribution and communication. The toolkit has been implemented in a mapping design and visualization tool which

is coupled with a compilation system for the HPF [6] predecessor Vienna Fortran. Since this language covers a superset of HPFs facilities, the tool may also be used for visualization of HPF data structures. The GASPARD (Graphical Array Specification for PARallel and Distributed computing) is a visual programming environment devoted to the development of parallel applications [7]. Task and data parallelism paradigm are mixed in GASPARD to achieve a simple programming interface based on the printed circuit metaphor.

The Movie-based Programming is our approach for promoting high-level language constructs introducing not only special symbols and images with semantic support, but also on a series of images that can present dynamical features of algorithms. The usage of this approach for numerical solution of some linear algebra problems allows to generate automatically rather effective sequential executable C-code [8]-[10]. OpenMP, a portable programming interface for shared memory parallel computers, was adopted as an informal standard in 1997 by computer scientists who wanted a unified model on which to base programs for shared memory systems [11]. The usage of OpenMP offers a comprehensive introduction to parallel programming concepts. The goal of this work is to develop a technique of OpenMP parallelization of Movie-based algorithms in order to obtain their high performance.

The rest of the paper is organized as follows. In Section 2, we discuss a concept of the Movie-based Programming including component description and definitions. The third section describes code generation process and adaptation it to the OpenMP environment. In Section 4, demonstrates the least-squares polynomial curve fitting as an example of the usage of our approach. The last section contains conclusion.

## II. MOVIE-BASED COMPONENTS AND DEFINITIONS

The Movie-based representation of computational methods and algorithms is based on a correspondence between algorithmic movie frames and problem solution steps. Accordingly, each frame should visualize/animate a corresponding step of a program/algorithm execution. Within the frame, structures are shown as a static images representing parameterized sets of nodes which can be connected by links in 4D space-time. The structure is any geometrical construction in a 3D space. Let us

consider the programming process as a specification of the following statements: *WHEN*-statement, *WHERE*-statement and *WHAT*-statement.

**WHEN-statement.** As was mentioned above, a *frame* is an image representing dynamical features of an algorithm at a particular time step. So, sequences of frames are observable as an animation and can be composed and visually debugged. A computational step is visualized as a combination of visual symbols within a frame. A set of frames joined into an animated sequence – i.e. a visualization of a computation on a structure according to the traversal schemes and the computational formulas for the frame – is called an “algorithmic film” or just “film” that is a combination of  $\mathcal{S}$  as a collection of spatial structures,  $\mathcal{D}$  as a set of variable declarations,  $\mathcal{M}$ , a collection of metaframes.

A *metaframe* is a special object representing a set of rules and parameters which are meant to specify how frames should be produced (visualized) in a film, and, how they should be implemented in an executable program. The two main types of metaframes, *single* and *episode*, are used to specify single or multiple algorithmic steps respectively. Any metaframe is a combination of  $\mathcal{T}$  as a set of traversal schemes for node activation on structures in  $\mathcal{S}$ ;  $\mathcal{C}$  as a set of control-flow formulas exploited in the frame-generation process, and  $\mathcal{F}$  as a set of computational formulas to be performed on nodes affected by schemes in  $\mathcal{T}$ ;

**WHERE-statement.** Rather often, the data structures in applications can be regular ordered sets of elements presented as 1D, 2D or 3D structures where structure nodes contain operable objects. So, sets of elements involved in computations can be related to the sub-domain nodes of these structures. Any structure has the attributes shown in (Figure 1) including **Name**, **Dimension**, **Parameters** to define structure sizes ( $N, M$  for rows and columns respectively), **Structure Nodes** as well as a set of **Structure Variables** ( $A$ ) declared to store instances of data in nodes, a set of sub-domain variables ( $\bar{R}$ ) and domain variables ( $R_\Delta$ ). It includes also a set of **Control Lines** which are used to refer to spatial placements and domains.

A sub-domain is a set of nodes which coordinates are satisfying to a system of constraints  $\Omega = \{(i, j) | H_1 \leq i < H_2 \wedge V_1 \leq j < V_2 \wedge P_1(i, j) \wedge P_2(i, j) \wedge \dots \wedge P_n(i, j)\}$ , where  $(i, j)$  - coordinates of nodes,  $H_1, H_2, V_1, V_2$  - positions of appropriate control lines or structure bounds and predicates  $P_k(i, j)$  - special conditions (they are used to specify the sub-domain shape). Accordingly, a domain  $\Delta = \Omega_1 \cup \Omega_2 \dots \cup \Omega_n$  can also be defined as a composition of sub-domains  $\Omega_i$ .

Any domain needs to have a special attribute to be visually distinguishable from other domains. This attribute marks all domain elements by color, shape, size, etc. Usually, we are using a color attribute that allows the user to operate with parametric relation-based specification.

**WHAT-statement.** All frame attributes can change their parameters by assignment operations on them. To specify these changes, the user should assign operators on a metaframe attributes that have to be implemented during frame processing.

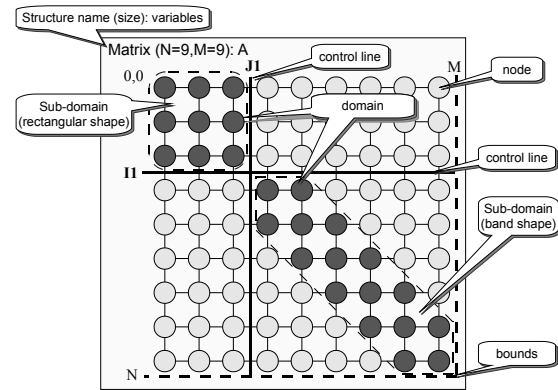


Fig. 1. Attributes and features of matrix structure and its sub-domains.

We distinguish two types of these operators: 1) *Computational formulas* to specify operations on node domains, 2) *Control-flow formulas* to define frame transitions (control line movements and episode conditions).

### III. MP-TEMPLATES AND CODE GENERATION

As was shown in previous section, a *domain*  $\Delta$  consists of elementary traversal schemes that parametrically enumerate elements arranged in a particular shape (dot, row, column, rectangle, triangle, band, etc). These schemes are called *Movie/Program template* or *MP-templates* each of which is a set of structure nodes which coordinates are satisfying to a system of constraints  $\Delta$ , and considered as a complete subpart of MP-program if it has C-formula attached. C-formula is defined as a subprogram containing a sequence of arithmetical and/or logical expressions.

By using appropriate types of metaframes, domain configurations and corresponding formulas, it is possible to specify an algorithm visually on a defined structure. Such an algorithm can be debugged on a structure of a small size appropriate for understanding algorithmic steps; however, the resulting code will be operable on a structure with a potentially unlimited size defined by corresponding parameters. The code generation is based on special template code snippets performing scanning steps on predefined elementary sub-domains: row, column, diagonal, triangle, etc, which are combined into scanning templates of an arbitrary complex domain. These template programs are used both 1) for visual demonstration of effected structure nodes within domains on a movie frame, and 2) for computation by applying formulas to appropriate elements.

The generation of executable code is realized using relation-based parameterized specifications of metaframes with domains, control lines and formulas as well as embedded transformation rules as shown in [10]. To obtain the parallel executable code, a set of template samples in C language for an OpenMP target platform was developed. This development was realized by embedding OpenMP Directives into original templates with automatic substitution of the metaframe attributes. As it is demonstrated below, this simple transformations allowed to obtain relatively effective parallel code.

#### IV. IMPLEMENTATION ON OPENMP ENVIRONMENT

##### A. Movie-based Least-Squares Optimization Method

The Least-Squares Optimization method is a mathematical procedure for finding the best-fitting polynomial curve  $f(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m$  along a given set of points  $(x_1, y_2), (x_2, y_2) \dots (x_n, y_n)$  by minimizing the sum of the squares of the differences between a curve and points. The coefficients of a curve  $a_0, a_1, \dots, a_m$  are found by solving the following SLAE  $A\vec{x} = \vec{b}$ :

$$\begin{aligned} a_0 \sum_{i=1}^n 1 + a_1 \sum_{i=1}^n x_i + \dots + a_m \sum_{i=1}^n x_i^m &= \sum_{i=1}^n y_i \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + \dots + a_m \sum_{i=1}^n x_i^{m+1} &= \sum_{i=1}^n x_i y_i \\ \dots & \\ a_0 \sum_{i=1}^n x_i^m + a_1 \sum_{i=1}^n x_i^{m+1} + \dots + a_m \sum_{i=1}^n x_i^{2m} &= \sum_{i=1}^n x_i^m y_i \end{aligned} \quad (1)$$

To generate the SLAE (1) it is necessary to calculate coefficients as sums that have arguments with degree parameter related to the position of a corresponding sum operation. Fig. 2 demonstrates a part of this algorithm related to the matrix builder. This episode is to fill up the matrix elements with the corresponding numbers from the left part of the SLAE.

There are three 2D grid structures defined. Structures “vector1” and “vector2” have a colored domain with the single sub-domain of row type defined to emphasize the first row which corresponds to the  $x$  coordinate of all points. Matrix structure “Matrix” has a colored domain with the single sub-domain of the minor-diagonal type moving from the top-left corner to the bottom-right corner. According to corresponding transitional expressions this frame-to-frame movement is controlled by two control lines attached:  $I_1$  and  $J_1$ . This episode is continued until  $I_1$  has reached the bottom border according to the specified conditional expression ( $I_1 < N$ ).

This specification allows the generation of executable source code which is presented at the bottom of Fig. 2. It is possible to see how the OpenMP operators are embedded into the C code generated. The rest of the algorithm includes vector  $\vec{b}$  building and SLAE solving metaframes which have been omitted due to the space limitations.

##### B. Numerical Experiments

The first type of experiments were implemented in order to evaluate the quality of the generated source code text in comparison with a sample solution taken from the tutorial [12] by the automatic code validator “splint” [13]. The other experiments evaluated the resulting code performance. The same code from the previous test have been used as a sequential sample to run on the single CPU. Then, OpenMP templates have been used to generate parallel sample codes. Example of results achieved in this experiment are demonstrated on Figure 3 with a dataset for  $n=10,000,000$ .

The left-side graphs in each dataset shows dependence between the time elapsed by each running code and the matrix size  $m$  (1). The right-side graphs in each dataset show speedup

of parallel code in respect to the sequential one. Testing platform: SMP server with four Intel Xeon 4-core X5550 2.67GHz (16 cores totally), with 12GB of memory, running Ubuntu 10.04 x86\_64 (server edition). The results demonstrate that both generated sequential and parallel programs can reach a suitable performance.

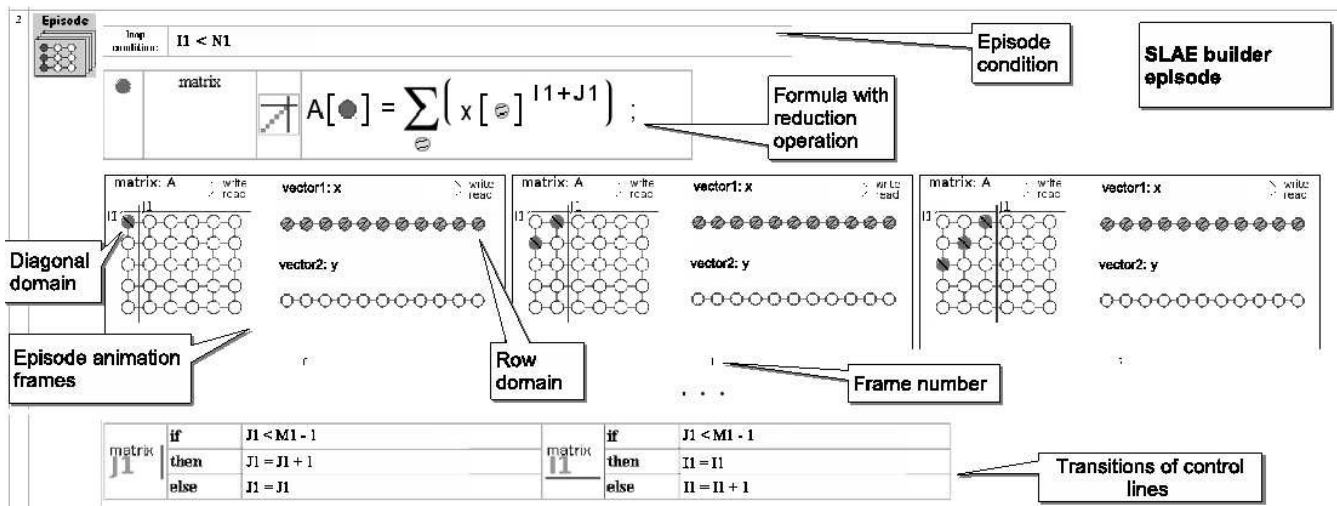
#### V. CONCLUSION

The Movie-based programming environment is presented using a concept of metaframes controllable by a special interface panels for specifying algorithmic movie-frames and spawning the automatic code generation. This includes the design of new visual symbols and the introduction of an automatic template generation technique supporting compact specification of computational expressions, and allowing an essential advance of the library of templates oriented to specify a variety of numerical matrix algorithms as well as parallel programming platforms. Visual semantic and syntactic rules have been presented defining domains with various shapes on computational structures, and various operations on those structures. The adaptation of a Movie-based program to parallel mode is implemented by modifying the shape-scanning templates according to the OpenMP programming rules. This simple modification allowed to generate a parallel code with relatively good quality. Applicability of the technique and environment has been demonstrated through the Least-squares Curve Fitting Method including implementation, code validity checking and performance testing.

#### REFERENCES

- [1] M. Burnett, *Visual Programming*, Wiley Encyclopedia of Computer Science and Engineering, John Wiley & Sons Inc., Hoboken, 1999.
- [2] Ph. T. Cox, *Visual Programming Languages*, Wiley Encyclopedia of Computer Science and Engineering, John Wiley & Sons Inc., Hoboken, 2008.
- [3] R. Rabenseifner, Optimization of collective reduction operations, *LNCS*, Springer-Verlag, Vol. 3036, 2004, 1-9.
- [4] J. Pjesivac-Grbovic, and T. Angskun, and G. Bosilca, and G. E. Fagg, and E. Gabriel and J. Dongarra, Performance analysis of MPI collective operations, *Cluster Computing*, Kluwer Acad. Publ., Vol. 10, No. 2, 2007, 168-179.
- [5] S. Grabner, and R. Koppler and J. Volkert, *Visualization of Distributed Data Structures for HPF-like Languages*, Technical Report, Johannes Kepler University Linz, 2005.
- [6] *High Performance Fortran (HPF)*, <http://www.netlib.org/hpf/>.
- [7] Fl. Devin, and P. Boulet, and J. L. Dekeyser and Ph. Marquet, “GASPARD - a Visual Parallel Programming Environment”, *Proc. of the Int. Conf. on Parallel Computing in Electrical Engineering (PARELEC'02)*, 2002, 145-150.
- [8] D. Vazhenin, A. Vazhenin and N. Mirenkov, “Movie-based Multimedia Environment for Programming and Algorithms Design”, *LNCS*, Springer-Verlag, Vol. 3333, No. 3, 2004, 533-541.
- [9] D. Vazhenin, A. Vazhenin and N. Mirenkov, “Movie-based templates for linear algebra problems”, *Int. Jour. of Comp. Sci. and Network Security*, Vol. 7, No. 1, 2007, 378-385.
- [10] D. Vazhenin, A. Vazhenin, “MP-templates Operating Toolkit in Movie-based Programming”, *Proc. of Japan-China Workshop on Frontier of Computer Science and Technology*, Nagasaki, Japan, 2008, 67-73.
- [11] *OpenMP: Simple, Portable, Scalable SMP Programming*, <http://www.openmp.org>.
- [12] T. Veerarajan, and T. Ramachandran *Numerical Methods: With Programs In C*, Tata McGraw-Hill, 2005.
- [13] *Splint: statical C code validator* <http://www.splint.org>.





```
main() {
    /* episode metaframe number 2 */
    while(matrix_I1 < matrix_N1) { // episode begin
        /* Domain color: rgb[255,0,0] */
        /* Subdomain Diagonal2 template begin */
        _St_Row=matrix_I1; _St_Col=0; _Fin_Row=matrix_N;
        _Fin_Col=matrix_J1; _Step_Row=1; _Step_Col=1;

        #pragma omp parallel for private(_Scan_I, _ScanF_J) default(shared)
        for(_Scan_I=0; _Scan_I < _Fin_Row - _St_Row &&
            _Scan_I < _Fin_Col - _St_Col; _Scan_I += _Step_Row)
        {
            _ScanF_I = _Scan_I + _St_Row;
            _ScanF_J = _Fin_Col - _Scan_I - 1;
            A[_ScanF_I][_ScanF_J] = fc_reduce_sum1(x, _control_data);
        }
        /* Subdomain Diagonal2 template end */
        /* moving control lines: */
        if(matrix_J1 < matrix_M1 - 1) matrix_J1 = matrix_J1 + 1;
        if(matrix_J1 < matrix_M1 - 1; else matrix_I1 = matrix_I1 + 1;
    } // episode end
}

/* generated temporary template for reduction sum */
double fc_reduce_sum1(double **x, _ControlDataP _control_data) {
    #include "filmdefs.h"
    double result=0.0;
    /** Mirroring main control lines and structure size via _control_data */
    int vector1_M = _control_data->size[0];
    ...

    #pragma omp parallel reduction(+: result)
    {
        /* Domain color: rgb[255,0,255] */
        /* Subdomain Row template begin */
        _St_Row=0; _St_Col=0; _Fin_Col=vector1_M;
        _Step_Row=1; _Step_Col=1; _Scan_I = _St_Row;

        #pragma omp parallel for private(_Scan_J, _ScanF_J) default(shared)
        for(_Scan_J = _St_Col; _Scan_J < _Fin_Col; _Scan_J += _Step_Col) {
            _ScanF_I = _Scan_I;
            _ScanF_J = _Scan_J;
            result += (pow(x[_ScanF_I][_ScanF_J], (matrix_I1 + matrix_J1)));
        }
        /* Subdomain Row template end */
    }

    return result;
}
```

Fig. 2. Movie-based program of least-squares method and an excerpt of generated code

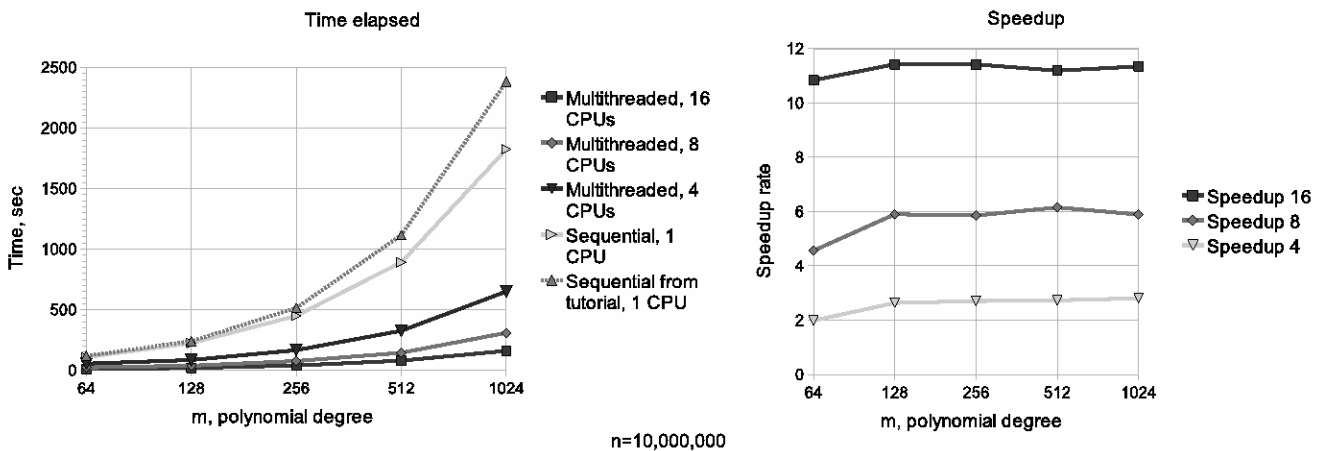


Fig. 3. Movie-based program of least-squares method and an excerpt of generated code

# 3<sup>rd</sup> International Symposium on Services Science

FOR THE third time, the "International Symposium on Services Science (ISSS)" will offer a unique platform for advancing research and discussions in Service Science. ISSS welcomes research from academia and practice on Service Science, which includes general concepts, design principles and paradigms, engineering methodologies and all activities associated with managing IT-based services.

Topics of ISSS 2011 include, but are not limited to:

- Service Science theories and principles
- Service design and engineering
- Service innovation and management
- Service marketing
- Service modeling and simulation
- Service industrialization und standardization
- Service pricing and revenue strategies and organization models for services
- Service configuration and evaluation
- Service systems complexity
- Service components and architecture
- Service platforms for orchestrating and maintaining services
- Service governance
- Service customization and localization
- Service operation
- Service technologies
- Service Science in application domains, such as logistics, finance, etc.

## GENERAL CHAIRS

**Rainer Alt**, Institute for Applied Informatics (InfAI) and University of Leipzig, Germany

**Klaus-Peter Fährnich**, Institute for Applied Informatics (InfAI) and University of Leipzig, Germany

**Bogdan Franczyk**, Institute for Applied Informatics (InfAI) and University of Leipzig, Germany

## PROGRAM COMMITTEE

**Witold Abramowicz**, Poznan University of Economics, Poland

**Hamideh Afsarmanesh**, University of Amsterdam, Netherlands

**Jörn Altmann**, Seoul National University, Korea

**Martin Benkenstein**, University of Rostock, Germany

**Luis M. Camarinha-Matos**, University of Lisbon, Portugal

**Haluk Demirkan**, Arizona State University, USA

**Klaus-Peter Fährnich**, University of Leipzig, Germany

**Ruben Dario Franco Pereyra**, Universidad Politecnica de Valencia, Spain

**Bogdan Franczyk**, University of Leipzig, Germany

**Walter Ganz**, Fraunhofer IAO, Germany

**Gerhard Heyer**, University of Leipzig, Germany

**Dimitris Karagiannis**, University of Vienna, Austria

**Reuven Karni**, Shenkar College of Engineering and Design, Israel

**Koichi Kijima**, Tokyo Institute of Technology, Japan

**Stefan Klein**, University of Muenster, Germany

**Ryszard Kowalczyk**, Swinburne University of Technology, Australia

**Helmut Krcmar**, Technical University of Munich, Germany

**Andrew Kusiak**, University of Iowa, USA

**Jari Kuusisto**, SC Research, Finland

**Pierfrancesco La Mura**, Leipzig Graduate School of Management, Germany

**Christine Legner**, European Business School, Germany

**Howard Lightfoot**, Cranfield University, UK

**Achim Luhn**, Siemens Business Services, Germany

**Helge Löbler**, University of Leipzig, Germany

**Kathrin Möslin**, University of Erlangen-Nürnberg, Germany

**Alex Norta**, University of Helsinki, Finland

**Thorsten Posselt**, Fraunhofer Center for Central and Eastern Europe, Germany

**Thomas Puschmann**, Direct Management Institute, Switzerland

**Diane Robers**, PriceWaterhouseCoopers, Germany

**Gerhard Satzger**, Karlsruhe Institute of Technology, Germany

**Gerik Scheuermann**, University of Leipzig, Germany

**Markus Stolze**, HSR Technical University of Rapperswil, Switzerland

**Günther Schuh**, RWTH Aachen University, Germany

**Charles Shoniregun**, University of East London, UK

**Miguel-Angel Sicilia**, University of Alcalá, Spain

**Martin Smits**, Tilburg University, The Netherlands

**Dieter Spath**, Fraunhofer IAO, Germany

**Rudi Studer**, Karlsruhe Institute of Technology, Germany

**Gerrit Tamm**, University of Berlin, Germany

**Wilhelm Taurel**, AFISM International, Germany

**Marja Toivonen**, Helsinki University of Technology, Finland

**John Wang**, Montclair State University, USA

**Florian von Wangenheim**, Technical University of Munich, Germany

**Christof Weinhardt**, Karlsruhe Institute of Technology, Germany

## ORGANIZING COMMITTEE

**Roman Belter**, University of Leipzig, Germany

**Rolf Kluge**, University of Leipzig, Germany



## Learning to Innovate in Distributed Mobile Application Development: Learning Episodes from Tehran and London

Neek Alyani

LLAKES Centre, IOE, University of London  
London, United Kingdom  
TIR Group, Faculty of World Studies,  
University of Tehran, Tehran, I.R. Iran  
Email: n.alayani@ioe.ac.uk

Sara Shirzad

Technenovate Associates, Tehran  
Islamic Republic of Iran

Email: SaraShirzad@hotmail.com

**Abstract**—This paper reports on the activities of an entrepreneurial small software firm, operating in telecoms value-added services based in Tehran, Iran, with project partners in London, UK. Mobile and smart phone applications are altering our professional and social interactions with innovative business models, glocal content and eco-systems, fusing the multifaceted aspects of mobile software development. To analyze these types of activities in the context of rapidly changing catching-up economies, development of mobile applications by entrepreneurial NTBFs, initially imitating as a way to innovate, require distributed up-skilling, rapid problem-solving and pragmatic learning. Specifically, we focus on knowledge brokerage and sourcing activities in distributed Scrums. Drawing on longitudinal analysis of projects [2004-2010], an iterative 'learning to innovate' model, entitled DEAL (Design, Execute, Adjust, Learn) within 'project-enhanced learning episodes', is constructed and outlined utilizing knowledge brokers and boundary sources in enterprise challenges. We conclude by reflecting on distributed learning and skills in practice.

### I. INTRODUCTION

THE overall aim of this paper is two fold: firstly to report in outline, the longitudinal learning and innovation activities in distributed small teams within the context of global software development (GSD) and secondly, by introducing our learning-to-innovate model, pave the way for a re-examination of the originally integrated (and now evolving) concept of 'learning' in practice. Specifically, in our model's construction, we take account of knowledge creation activities and skills within the service innovation process, by means of knowledge brokerage and sourcing. Mobile and smart phones applications are significantly altering our social domain and professional interactions [1] with innovative business models, content and eco-systems emerging glocally (globally modelled yet locally scaled), outlining the 'mobile big bang' and fusing the business and technical aspects [2-4] of the mobile solutions development.

Since their introduction and uptake, project-oriented agile software development methods, such as Scrum and Lean [5-7], whilst subject to recent conceptual concerns (e.g. brief summary of critique under Section 2.1 of [8]), have been

well received in the practitioner community and are starting to significantly change and transform the software industry in small and large firms and projects. Additionally, there has also been some recent movement on exploring agile methods for mobile software applications and equally of interest, multiple systematic and historical reviews related to Scrum and Lean methodologies have found their way into the literature [9] providing a better background on the trajectory of their development, and underlying assumptions.

In rapidly changing external environments of catching-up economies, development of mobile applications by entrepreneurial small firms, initially imitating as a way to innovate [10], requires distributed up-skilling, rapid problem-solving and pragmatic learning [11]. In transitional and catching-up economic climates, Scrum, as a project methodology is viewed to have natural advantages in development of mobile applications based on having a disciplined and limited scope, high customer/end-user interaction, and condensed time to market cycles. Drawing on innovation management and workplace learning corpus, distributed innovation with technologies and developing dynamic capabilities, framed as the engine of the firm's sustainable competitive advantage [12], offers competitive action in an unstable and unpredictable market. Conversely, learning episodes in distributed project activities, such as in Scrums, provides some stabilisers to compete in the market.

The case study analysis, based on the longitudinal scope [2004-2010] outlines a learning cycle in the exploration and exploitation phases of projects [13], identified and expanded upon to highlight the 'project-enhanced learning episodes' as a unit of analysis, utilising knowledge brokers, knowledge creation methods, and networking modes [14-17], and cross-border knowledge sourcing strategies by SMEs [18], particularly micro New Technology Based Firms (NTBFs)<sup>1</sup> in technical and non-technical challenges. We also note that NTBFs in some developing economies (e.g. Iran) endowed with skilled, technology savvy and connected teams, tend to mimic MNC/TNCs' 'skill webs' [19] to survive and prosper.

<sup>1</sup> Within the high-tech sector, NTBFs regularly operate as the vanguard of product, process and service innovation: the firms are usually formed by highly educated and skilled entrepreneurs, with a high rate of growth and firm mortality significantly contributing to sectoral and national economies.

Study is supported by LLAKES Centre, IOE, University of London.

The paper concludes by further highlighting an analytical and empirical tension on whether the concept of learning in Scrum (and Sprints) has evolved and is thus, in need of re-conceptualisation. This is particularly with reference to small firms engaging in technological innovation in services and operating in the new and expanding sectors of the economy, such as the Creative and Cultural sectors [20]. Originally drawn from the game of rugby and applied to New Product Design, initially empirically based on a manufacturing mode of operation in mid to late 1980s [21-25], and transported into and enthusiastically taken-up by the software development community over a decade ago [26-27], we enquire on whether the learning opportunities are still present in Scrum, and if the metaphors still offer useful analytical means in exploring the challenges of inter-professional learning and work in our current era [28-29].

## II. CHALLENGES AND ANALYTICAL CONCEPTS

As an entrepreneurial software solutions firm, the overall challenge for the projects and personnel of this case study is simple: to develop and offer value-proposition, by the processes of identifying, evaluating, and exploiting business opportunities to create new digital services [30]. Entrepreneurial activities, following the Schumpeterian economics line of argument [31] differentiates between the invention of new knowledge and its commercialisation, and views the process of commercialising existing knowledge as essential for entrepreneurial activities, bridging as yet unconnected sources of expertise and knowledge and re-combining already known inventions in an effort to create new business opportunities [14].

Additionally, it is becoming increasingly apparent that within the context of globally distributed work and by extension, global software development, the 'production system' team needs to not only deal with cross-functional and inter- and intra-disciplinary work, as extensively studied previously (e.g. as in the evolution and use of the influential SECI model of knowledge creation) [21-25], [32], but also deal with locational, temporal and relational boundaries. These 'boundary crossing' [33] activities require continued orchestration of the firm's effort beyond strategy, structure and systems, and towards purpose, processes and people [34-36]. It is thus our contention that knowledge brokerage and knowledge sourcing, have taken centre stage in fulfilling the requirements of the new *knowledgeability* [28] demanded at NTBFs. We view knowledge brokerage and sourcing in innovation, as formal and informal means to bring people together, and create new purposefully productive and expandable relationships, in order to define, design for, and solve problems in firms. An important by-product of these processes is learning and skill development, which derives out of workplace brokering, embedded in largely episodic, unrecognised and unplanned activities.

As a way of introducing the context, it is critical to note that the study was conducted during a time of unprecedented growth both in the usage of mobile phones and expansion of associated services in Iran (spanning 2003 to 2010), when and where the mobile penetration rates far exceeds the internet usage and many subscribers started to explore and use their mobile phones as personal and often primary communication device. With an estimated current population of 75.5m in a geographic area, approximately three times the size of France, the mobile penetration rates stood at just over 58% [37] in 2009-2010, from next to nothing (0.4% to 0.8% based on ITU figures for 1998-1999 [38]), about a decade before. The increase in penetration rate whilst tapering off, remains robust to date, based on cheaper (variations of 'pay-as-you-go' and pre-paid) subscriptions and contracts, offered by incumbents and newer 3G (and 3.5G/HSPA+) mobile telecommunications license holders.

As the nascent information and communication technology market including the telecom, and telecom value-added services segments, was still shaping, the firm examined in this study, re-labeled here as Alpha Company was formed as a fully private company in 2002. As a small entrepreneurial firm, starting out with a few co-founders with technical (software, science and engineering) and business acumens, it initially engaged in testing the market with a range of software services based on the outsourcing model. Whilst it had initial small successes in securing outsourcing contracts from EU, the internal market and particularly the niche market of mobile application software and solutions looked more promising, leveraging on an implicit 'Blue-Ocean' [39] strategy emphasising strong technical innovation on sought-after global solutions for local needs, as part and parcel of the emerging global 'Mobile Big Bang' model, illustrated below in figure 1.

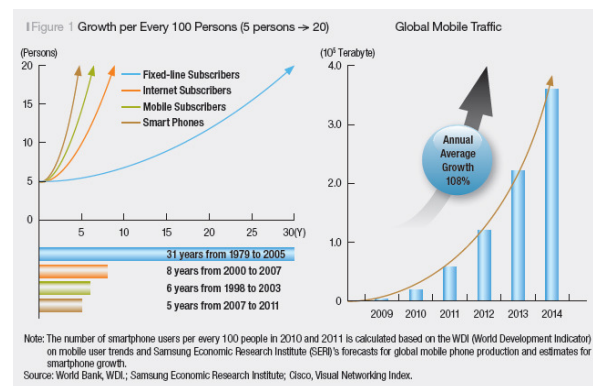


Fig. 1 'The mobile big bang' figures: SERI Quarterly, Kang 2010 [3]

Examining the market and the rapid changing patterns of mobile handsets, AlphaCo set out to develop a stable platform for business solutions, offered to both the public and private sector primarily in Tehran. Technically, drawing

on the partner in London and imitating to innovate, it opted for building applications and solutions on a tested ‘common denominator’ of SMS (short messaging service) as an embedded, and till then largely redundant feature (due to lack of popular use and small subscriber numbers mainly interested in core service of voice communication), within the Iranian national GSM network. Software development and testing using Java Platform, Micro Edition (Java ME<sup>2</sup>), previously known as Java 2 Platform, Micro Edition (J2ME), as a Java platform designed for embedded systems (e.g. for mobile devices) were undertaken under local and later distributed Scrum. As the mobile telecom market grew in size, the SMS VAS<sup>3</sup> (short messaging service value-added services) segment grew with it. The full force of mobility, as a business service revolution [4], whilst delayed for about a decade compared to Western Europe and Far East, had at last arrived in Iran. With the development of technological tools, NTBFs such as AlphaCo, and a select number of University-Industry based research labs, started to engage in pioneering service science in practice, exploring service design and innovation, in Tehran [40-41].

In telecommunication industry, a value-added service (VAS) is a term for non-core services, i.e. all services beyond standard voice calls and fax transmissions. It can also refer to any service industry, for services available at little or no cost, to promote their primary business. On a conceptual level, value-added services add value to the standard service offering, encouraging the subscriber to use their phone more and allowing the operator to drive up their ARPU (average revenue per user). For mobile phones, while technologies like SMS, MMS and GPRS have traditionally been considered as value-added services, a distinction is made between standard (peer-to-peer) content and premium-charged content. Value-added services are supplied either in-house by the mobile network operator themselves or as in the case of this study, by a third-party value-added service provider (VASP), also referred to as a content provider (CP). VASPs typically connect to the operator using protocols like short message peer-to-peer protocol (SMPP), connecting either directly to the short message service centre (SMSC) or, to a messaging gateway that allows the operator to control the content and speed of delivery better.

<sup>2</sup> Java ME was designed by Sun Microsystems, recently becoming a subsidiary of Oracle Corporation. There are presently an estimated 3 billion Java ME enabled mobile phones and PDAs used globally (<http://www.java.com/en/about/>), although the technology is increasingly viewed as ‘old’ technology as it is not used on any of today’s newest mobile platforms (such as iPhone by Apple, Android now owned by Google, Windows Phone 7 by Microsoft, MeeGo, initially supported by Intel and Nokia [now subject to change due to Nokia’s major reorientation] as well as Linux Foundation, Novell and AMD; and BlackBerry/RIM’s new QNX).

<sup>3</sup> Despite the technological progress on smart phones and mobile platforms in the West and Pacific Rim, based on their market size and large heterogeneity of mobile handsets, SMS VAS remain a significantly healthy segment in markets such as India, China, and much of developing Middle East, Africa and Far East. Smart phones and 3G do not, yet, rule globally.

#### A. *The Theoretical Concepts and Contexts*

The core of this study explores the processes of learning that leads to innovation in a small entrepreneurial private firm. Specifically, the role of knowledge brokerage and knowledge sourcing, to assist in and facilitate the process of knowledge creation is examined. The ‘wider lens’ of the study explores the changing relations between knowledge, learning and innovation in Iran’s private sector software development. For instance, tensions that arise within the digital technology sector, when the geo-political forces and national strategies collide were apparent early on. Equally, the paradox of a technically educated, yet unskilled workforce became increasingly a burden for the firm. The empirical case study evolved into focusing on a firm currently operating within the value-added services (VAS) engaged in designing joined-up advertising campaigns, banking and public services on mobile platforms.

In order to scale the project, and by extension, sharpen the focus of the lens of the study, many choices have had to be made and limitations imposed. These include concentration on a single firm, in a single city, in a single country, namely Tehran, Iran linked to a single external entity in London, UK. Theoretically however, only robust principles and fundamental concepts found in the academic (and at times, consultancy) literature are mobilised, and have guided and informed the directions pursued in the different stages of the study. Methodologically also, choices made around the ‘unit of analysis’ and the ‘longitudinal design’ intended to permit a deeper and more nuanced understanding of learning and innovation in the context of a firm operating in a transitional economy. An interdisciplinary approach is utilised outlining the macro- and meso-frameworks and factors, as well as close attention paid to the micro level practices, which breaks away from a one-dimensional and ‘cross-sectional’ snap-shot analysis. As researchers, we were conscious of exploring methods that avoid primarily snap-shot views. Using a metaphor from traditional photography, we were acutely interested to keep the ‘shutter’ open for long enough to be able to view and explore changes taking place, and yet, not overexpose the textural composition of the film. Closely congruent with our underlying epistemology and ontology, our methodology anchored around analytic induction [42] within a context of applied qualitative research [43]. The approach, although time-consuming, resource-hungry and iterative, leads to shedding light on interdependencies and interactions of often embedded social factors and institutions existing at the societal, sub-societal and subterranean level.

The study examined, en route, the changing relationship between the local and the global in Iran (and issues around locality), in particular highlighting the contradictions and tensions in simultaneously promoting an ‘insular’ and ‘connective’ approach within its economy and by extension, technological interactions. This is embedded within an

historically ambivalent pattern of political economy of a 'rentier' system, set as far back as the late 1960s to early 1970s, accentuated by the promotion of the self-sufficiency<sup>4</sup> discourse propagated during the Islamic revolution of 1978-1979, and embedded in the establishment of the Islamic Republic in 1979 and further reinforced and consolidated by the 'self-reliance' legacy of the Iran-Iraq war (1980-1988), while dealing with chronic U.S. sanctions [44].

Much has been written on related matters, and journalistic and scholarly literature on Iran has proliferated and there remains a continuing alarmist discourse, in the last decade. This has been further exaggerated by Iran's implicit positioning to regional super-power status following the regime changes in Iraq and Afghanistan and the 'Arab Spring'; insistence on the legality of its right to harness nuclear technology for peaceful purposes as well as an ongoing rhetorical dispute with Israel. Away from geopolitics however, there is a dearth of nuanced and objective accounts on the complexities of Iran's post-revolutionary socio-economical position, especially policies related to human resource development, skill formation and development and knowledge creation [38], [45-46] in its networked organizations and modes of operation, including means to circumvent the effects of long-term economic sanctions on technological and globally sought-after skills, and confronting the gales of 'compressed modernity' [47].

Iran has ambitious plans to become a well-connected advanced regional power, and to have the potential to 'project' its power<sup>5</sup>. This is legally manifested in the '*Iran 2025 Vision*' (locally referred to as "*Iran's 20 year Vision*<sup>6</sup> document") which is essentially a brief, yet overarching policy orientation of the next 15 years' development plans, setting the agenda for the four, 5 year development planning cycles (4<sup>th</sup> to 8<sup>th</sup> post-revolutionary five year development plans, spanning 2005-2025). The '2025 vision' envisages a range of social justice, revolutionary and Islamic ethos trajectories, and political and economic development programmes, as approved by IMF [48], amongst which a move towards an advanced knowledge-based rapid development and knowledge-creating modes of operations (in science and technology) is to be prioritised and promoted. There also exists a national emphasis and aspiration on developing a knowledge-creating mode of operation within the economy, as opposed to being 'a mere consumer' of externally produced technological knowledge.

<sup>4</sup> The closest international model, still actively promoted, to this discourse is that of "*Juche*" (*self-reliance*) in the Korean Peninsula.

<sup>5</sup> We utilize and expand the US DOD definition of 'power projection' and the DIME [-R] model standing for *Diplomatic, Information/Intelligence, Military and Economic* and in our addition, *Religious power*.

<sup>6</sup> For more details (in Persian), see *Iran's Expediency Council* website at: [www.irec.ir](http://www.irec.ir) and [www.maslahat.ir](http://www.maslahat.ir)

Within the focus of this study however, whilst significantly leveraging on the analysis offered and firm foundations laid by Guile [28] in unpacking the issues, we draw on a recent articulation by Lauder and colleagues [49], [19] as part of a UK ESRC programme of research that sheds light on the topic. In essence, Lauder asserts that while many claims about the knowledge economy have been made, few stand the test of empirical scrutiny. One area however that displays a clear and substantiated trend is in the relationship between knowledge economy and innovation: innovation is at the heart of the knowledge economy, as it is about the intensification of competition. As innovation takes centre stage in the economic life of firms and nations, *learning to innovate* becomes a matter of high priority for firms. Learning and innovation in firms however are multifaceted and multi-layered processes, utilising different forms of knowledge/s and *knowledgeability*. A nuanced examination of evidence portrays the contemporary learning in firms as possessing complex, rapidly changing and context-specific characteristics. No firm in isolation, whether large or small, is likely to acquire and maintain the necessary portfolio of expertise and skills required in its development. This was uniquely captured by early research, stating "firms are not islands but are linked together in patterns of co-operation and affiliation" [50, p. 895]. As the source of competitive advantage shifts to human resources, there follows a consequent increased interest in learning and development issues. As Guile [51, p. 470] reiterates, "people, rather than such traditional factors of production as capital, will become the main source of value and economic growth in this new type of capitalism, and that in future, more and more productive activities will make use of employees' intellect and creative capabilities." This is further reflected by a differentiation between four categories of knowledge as: *know-what*, *know-how*, *know-why* and *know-who*, the latter of which is increasingly important in knowledge transfer [52] and viewing skill development [53].

The recent metaphors for learning in firms takes many shapes, such as acquisition, participation and inquiry-based [54] and practically, this is complemented by learning through multiple communities of practice and inquiry [55], [56] via networks (whether 'weak/loose' or 'strong/tight', formal or informal and real or virtual) along with other firms and partners, providing the potential opportunity to collaborate and share material resources and perhaps more significantly, non-tangible resources and new perspectives.

Additionally, within the study's empirical elements observed in the projects, spanning working days in London and Tehran<sup>7</sup>, project time (and by extension, hourly costs) and practical collaboration factors is worth reiterating. As

<sup>7</sup> Tehran is 3.5 hours ahead of GMT and Thursday (half day in private sector) and Friday are the weekend days. This means that the overlap "real-time" project hours are limited to, at best, 7 hours for 3.5 days per week.



Brown, Lauder and Ashton [49, pp. 136-137] observe: “To reduce the time from ‘innovation to invoice’ some companies use 24-hour design teams that work around the clock, moving through time zones across Asia, Europe and North America. This is not only intended to reduce the time between invention, application, and market launch, but also to reduce costs, due to lower salary levels in much of Asia... It is also assumed to reflect the importance of embedded capabilities as innovation rarely depends on the skills of individuals working in isolation but on a culture of mutual collaboration and purpose... companies are increasingly experimenting with research, design, market and product development activities in the emerging economies.”

Innovation<sup>8</sup> has thus received much attention in the recent years originating from varied epistemological perspectives. The concept of innovation has evolved from being a uni-dimensional, linear process to a systemic approach in which complex interactions between individuals, firms and their operating environment are paramount [12], with recurring themes in business literature that points to the strategic nature of innovation as a source of economic growth and sustainable competitive advantage, both on the national economy, and the firm level [57]. As a caveat that we have highlighted elsewhere [58-60] so we bracket-out here, there is a dearth of ‘knowledge work’ and ‘division of cognitive labour’ studies that consider developing or transitional economies. The highly cited examples within the literature are based on firms in the developed and advanced economies [13], [25], [61] and as such, there is a scarcity of exploratory and/or analytical frameworks dealing with developing and transitional economies, and even more so on small firms and learning within those environments.

#### B. Towards an Analytical Model: ‘DEAL’ Iteration

Thus in formulating our analytical framework, we took account of this anomaly and attempted to ground our observations. As no single strand of literature provided the necessary theory, we brought together arguments of several theories and soon traced patterns of cyclical exploitation and exploration. Exploitation refers to the firm’s refinement and development of existing knowledge with predictable outcomes, whereas exploration refers to the pursuit of new knowledge with uncertain outcomes [13]. We further noted that the nature of learning is in the form of generative interactions between individual and collective inquiries [54]. These are placed on the horizontal and vertical axis of our schematic, outlined in Figure 2. In the center of the figure, drawing on the ‘project-enhanced learning episodes’, we noted the zone of ‘collaboration’ and ‘coordination and control’ activities within projects, as articulated and facilitated by the cycles of Scrums (and Sprints).

<sup>8</sup> The remit of this study focuses on *innovation in services and service design and development* and not focused on new ‘products’.

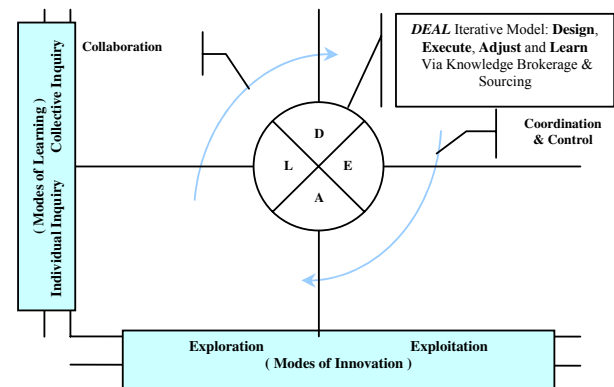


Fig. 2 DEAL iterative model schematic: learning and innovation dimensions in projects

At the heart of the activities however, we noted a range of processes which we labeled as *DEAL*, as an acronym that stands for the cycle of Design, Execute, Adjust and Learn. Within the DEAL model, various activities were enhanced via formal and informal knowledge brokering and knowledge sourcing. A sample series of questions, relating to each problem or inquiry, which are asked at each stage include:

**Design:** What is desirable and viable, and how feasible?

**Execute:** What is the expected outcome and impact?

**Adjust:** What worked and what did not, and why?

**Learn:** What is the core problem and cause? Reframe?

The cycle continues with framing and reframing of the new problem and inquiry, which then leads to a new design imperative, transforming prototype to archetype, till a solution is formulated. Brokerages and sourcing occur initially via formal means (e.g. IJVs) but mainly informally with trust gained, via spillovers, by 1. Visits to technology fairs/workshops, 2. Exposure to global professional/R&D networks, and 3. Participation in online developers’ space.

On a practical and longitudinal level, researching small firms has its own unique dynamics, as Guile [62] observes, “Conducting research in SMEs is notoriously tricky... for the following reasons: tight staffing and short deadlines means it is difficult to release people from their work roles; lack of space means that it is difficult to convene meetings; and the lack of a ‘learning culture’ means that SME owners are often reluctant to give up their time and that of their workforce to participate in a research activity.” In attempts to overcome some of these difficulties, the research formulated ways and means around the problem to fit-in with the priorities of the firm and a methodology that reflected some of the firms and staff’s concerns and priorities. On a different note, while the concept of design [63-64] (and derivatives such as *design thinking*, *unified*

*design* and *design-driven/inspired innovation*) has seen a surge in the business literature, our observation was that in practice, it is often condensed to a ‘balancing equation’ of desirability, viability and feasibility of the service innovation. This is concisely captured by the sketch below by Tim Brown at IDEO [64]. We outline a vignette of findings in the next section.

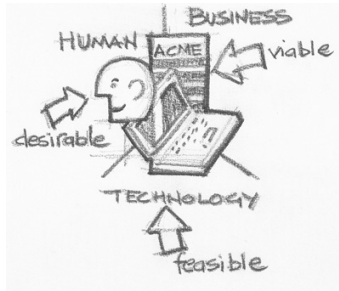


Fig. 3 A rough schematic of design thinking issues (*Desirable-Viable-Feasible*) by Tim Brown, 2008 [<http://designthinking.ideo.com/?p=49>]

### III. RESEARCH METHODS AND FINDINGS IN BRIEF

The empirical elements investigated sharing of problem-setting and -solving expertise on the issues that emerge out of daily business challenges in projects, which is both of a technical (software) and a commercial (business model and service design) nature. Two analytic tools were employed initially to scan the position of the firm within its operating terrain: these were a *PESTLE* analysis; outlining the Political, Economic, Social, Technological, Legal and Environmental factors (fed from the firm’s existing SWOT assessment) and *Scenario Planning* outlining where firms viewed themselves in relation to the operating environment and their potential pathways to growth. These analytic tools are commonly used in organizations and are often utilised by organizational consultants and senior managers, as despite shortcomings, are robust in providing a snapshot of the firm’s current posture, as a starting position.

The research was conducted as a case study [65-69], within the interpretive tradition, and in addition to the organisational ethnography and interviews (which leveraged on previous research in similar orientation [69-70]), a line of historical analysis was undertaken to enrich the context of the case and supplement the findings. In essence, based on analytic induction [42], an historical analysis, i.e. ‘historicity’ elements within the firm, sector and national/international factors embedded in the case study approach have guided the methods of data collection and analysis. The primary-sourced data is of a longitudinal nature and comprises of two focus group meetings, 18 semi-structured interviews and ongoing organisational ethnographic observations across 2004 to 2010, plus ad-hoc London-based meetings and updates. The longitudinal research design involves more than one episode of data

collection, in line with a ‘panel design’, where (as far as possible) the same people are contacted, observed and/or interviewed more than once [71], with the orientation and focal questions mirroring previous research [70]. This design strengthens the shortcomings of a single case study [72] and is of particular value when time-critical processes such as learning are observed. These included four weeks (December 2003- January 2004), two weeks (June 2007) and one week (June 2008) ‘immersion’ based in Tehran, followed by various days accumulating to two weeks (between July to September 2009 and in Spring 2010) as final follow-up in and from London, as well as ad-hoc virtual contacts and “issues’ tracing”, as necessary. As an exploratory study on how learning and innovation unfolds, this account captures a connected slice of reality, via the lens of activities on projects that facilitate what is labelled as *project-enhanced learning episodes*. Learning episodes are taken as the primary unit of analysis within the outlined model: they are here defined as “an occasion in which a [project] team learned something significant that advanced the project” in line with previous studies [69, p. S20]. Within the episodes, attention was directed at identifying circumstances when and where project team reaches a ‘break-through’ and/or a ‘cul-de-sac’, falling within the spheres of explorative or exploitative learning spheres. In the context of the episodes, ambidextrous learning was spotted on occasions, defined here as simultaneously exploring new knowledge and expertise domains while exploiting current ones, and derived from the unique co-configuration and design, drawing on prototyping and reflection (*Hansei*). As it was summarised by a team member “Our learning here is all about ‘beta’: learning and innovation are coupled and yet learning comes first”.

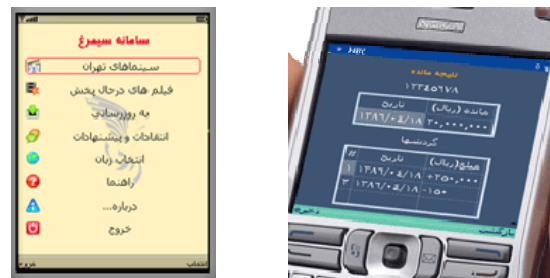


Fig. 4 An example of screenshots for public service offerings outlining a list of current programmes on Tehran’s cinemas (left), and a mini-statement by mobile banking in 2006 (enabled via SMS and J2ME)

### IV. RETROSPECTIVE AND REFLECTIONS

Within the context described above, namely undertaking project-based tasks, within a rapid and transitional societal and sectoral change, the study explored the following research propositions: firstly, the activities (as observed in ‘learning episodes’ and brokerage) that the project team members of the firms undertake in their joint projects that facilitate their ability to innovate; and secondly, how the

firm designs strategic learning approaches in practice, with due considerations for the firm's operating environment, as solutions to these latent needs. In addressing the research propositions, we aimed for fidelity in the analytical model and rigor in the analysis, as our prime methodological objective. While 'unpacking' the learning episodes, three interconnected processes were identified embedded within the DEAL iterative model as follows: firstly, conducting the inquiry; secondly, framing and/or reframing the problem/s (problem setting); and thirdly, mobilizing learning episodes to cultivate potential solution/s. Use of informal 'skill webs' [19] for knowledge brokerage and sourcing is also evident.

Our model is aligned with recent research both in SMEs [18], [57] and larger firms focusing on phronesis and acting with an 'idealistic pragmatists' [32], [73] mindset, required to tackle problems, in practice. As a concluding note, Nerur and Balijepally [74, p. 81] have recently cogently compared agile development to maturing design ideas in strategic management and design: "the new design metaphor incorporates learning and acknowledges the connectedness of knowing and doing (thought and action), the interwoven nature of means and ends, and the need to reconcile multiple world-views". This study has aspired to capture some of the delicate inter-connectedness between knowing and doing, in practice via Scrums, in the context of NTBFs in Tehran.

#### ACKNOWLEDGMENTS

We gratefully appreciate the invaluable comments of our five IEEE anonymous reviewers. Any remaining errors are solely our own responsibilities. The first author gratefully acknowledges the continued guidance of David Guile and assistance from LLAKES, WLE Centre and London Knowledge Lab, IOE, University of London, along the 'longitudinal path' of this research. In Tehran and London, Amir Jalilvand and Hamid Shayegan within the technology sector, and Saied R. Ameli of FWS, University of Tehran deserve sincere gratitude for their resourcefulness and pragmatic tenacity. Thanks are also due in Cambridge, Massachusetts to the scholarship assistance by MIT PEP for the studies at MIT in 2010 and to Amy C. Edmondson of HBS for her support on 'episodes' & 'fit', when it was most appreciated. Lastly, thanks are [long over-] due to the projects and firms' staff for openly and patiently sharing their experiences and insights.

#### REFERENCES

- [1] M. Castells, M. Fernandez-Ardevol, J. L. Qiu, and A. Sey, *Mobile communication and society: a global perspective*. Cambridge, MA: MIT Press, 2006.
- [2] E. W. T. Ngai and A. Gunasekaran, "A review for mobile commerce research and applications," *Decision Support Systems*, vol. 43, no. 1, pp. 3-15, Feb. 2007.
- [3] M. Kang, "The mobile big bang," *Samsung Economic Research Institute (SERI) Quarterly*, vol. 3, no. 4, pp. 78-85, Oct. 2010.
- [4] D. Steinbock, *The mobile revolution: the making of mobile services worldwide*. London and Philadelphia: Kogan page, 2007.
- [5] K. Schwaber, *Agile project management with scrum*. Redmond, WA, USA: Microsoft Press, 2004.
- [6] E. Woodward, S. Surdek, and M. Ganis, *A practical guide to distributed scrum*, 1st ed. Boston, MA: IBM Press, 2010.
- [7] E. V. Woodward, R. Bowers, V. S. Thio, K. Johnson, M. Srihari, and C. J. Bracht, "Agile methods for software practice transformation," *IBM Journal of Research and Development*, vol. 54, no. 2, 2010.
- [8] T. Dyba and T. Dingsoyr, "Empirical studies of agile software development: A systematic review," *Information and Software Technology*, vol. 50, no. 9-10, pp. 833-859, Aug. 2008.
- [9] B. Kitchenham, "What we learn from systematic reviews," in *Making Software: what really works, and why we believe it*, A. Oram and G. Wilson, Eds. Sebastopol, CA: O'Reilly Media, 2011, pp. 35-54.
- [10] L. Kim and R. R. Nelson, *Technology, learning and innovation: experiences of newly industrializing economies*. Cambridge and New York: Cambridge University Press, 2000.
- [11] B. Nicholson and S. Sahay, "Building Iran's software industry: an assessment of plans and prospects," *The Electronic Journal of Information Systems in Developing Countries*, vol. 13, no. 6, pp. 1-19, 2003.
- [12] J. Tidd and J. Bessant, *Managing innovation: integrating technological, market and organizational change*, 4th ed. Chichester: Wiley, 2009.
- [13] J. G. March, "Exploration and exploitation in organizational learning," *Organization Science*, vol. 2, no. 1, pp. 71-87, Jan. 1991.
- [14] A. B. Hargadon, "Brokering knowledge: linking learning and innovation," in *Research in Organizational Behavior*, B. M. Staw and R. M. Kramer, Eds. Greenwich, CT: JAI Press, 2002.
- [15] A. Boden and G. Avram, "Bridging knowledge distribution - The role of knowledge brokers in distributed software development teams," in *ICSE Workshop on Cooperative and Human Aspects on Software Engineering*, Los Alamitos, CA, USA 2009, pp. 8-11.
- [16] R. Thorpe, R. Holt, A. Macpherson, and L. Pittaway, "Using knowledge within small and medium-sized firms: A systematic review of the evidence," *International Journal of Management Reviews*, vol. 7, no. 4, pp. 257-281, 2005.
- [17] L. Pittaway, M. Robertson, K. Munir, D. Denyer, and A. Neely, "Networking and innovation: a systematic review of the evidence," *International Journal of Management Reviews*, vol. 5, no. 3-4, pp. 137-168, 2004.
- [18] R. Huggins, H. Izushi, N. Clifton, S. Jenkin, D. Prokop, and C. Whitfield, *Sourcing knowledge for innovation: the international dimension*. London: National Endowment for Science, Technology and the Arts (NESTA), 2010.
- [19] D. Ashton, P. Brown, and H. Lauder, "Skill webs and international human resource management: lessons from a study of the global skill strategies of transnational companies," *The International Journal of Human Resource Management*, vol. 21, no. 6, pp. 836-850, 2010.
- [20] D. Guile, "Moebius strip enterprises and expertise in the creative industries: new challenges for lifelong learning?," *International Journal of Lifelong Education*, vol. 26, no. 3, pp. 241-261, 2007.
- [21] K. Imai, I. Nonaka, and H. Takeuchi, "Managing the new product development process: how Japanese companies learn and unlearn," in *The Uneasy alliance: managing the productivity-technology dilemma*, K. B. Clark, R. H. Hayes, and C. Lorenz, Eds. Boston, MA: HBS Press, 1985, pp. 337-376.
- [22] H. Takeuchi and I. Nonaka, "The new product development game," *Harvard Business Review*, vol. 64, no. 1, pp. 137-146, Jan. 1986.
- [23] I. Nonaka, "The knowledge-creating company," *Harvard Business Review*, vol. 69, no. 6, pp. 96-104, Dec. 1991.
- [24] I. Nonaka, "A dynamic theory of organizational knowledge creation," *Organization Science*, vol. 5, no. 1, pp. 14-37, Feb. 1994.
- [25] I. Nonaka and H. Takeuchi, *The knowledge-creating company: how Japanese companies create the dynamics of innovation*. New York and Oxford: Oxford University Press, 1995.
- [26] K. Schwaber, "Scrum development process," in *OOPSLA Business Object Design and Implementation Workshop*, Austin, Texas, 1995, vol. 27, pp. 10-19.
- [27] K. Schwaber and M. Beedle, *Agile software development with scrum*. Upper Saddle River, NJ: Prentice Hall, 2001.

- [28] D. Guile, *The learning challenge of the knowledge economy*. Rotterdam: Sense Publishers, 2010.
- [29] N. Fonda and D. Guile, "Joint learning adventures," *People Management*, vol. 5, no. 6, p. 38, Mar. 1999.
- [30] S. Shane, "Technological opportunities and new firm creation," *Management Science*, vol. 47, pp. 205-220, 2001.
- [31] J. A. Schumpeter, *Capitalism, socialism and democracy*. London: Allen & Unwin, 1943.
- [32] I. Nonaka, R. Toyama, and T. Hirata, *Managing flow: a process theory of the knowledge-based firm*. Basingstoke, Hampshire: Palgrave Macmillan, 2008.
- [33] S. L. Star, "The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving," in *Readings in distributed artificial intelligence [2]*, M. Huhns and L. Gasser, Eds. Menlo Park, CA: M. Kauffmann, 1989, pp. 37-54.
- [34] C. A. Bartlett and S. Ghoshal, "Changing the role of top management: beyond strategy to purpose," *Harvard Business Review*, vol. 72, no. 6, pp. 79-88, Dec. 1994.
- [35] S. Ghoshal and C. A. Bartlett, "Changing the role of top management: beyond structure to processes," *Harvard Business Review*, vol. 73, no. 1, pp. 86-96, Jan. 1995.
- [36] C. A. Bartlett and S. Ghoshal, "Changing the role of top management: beyond systems to people," *Harvard Business Review*, vol. 73, no. 3, pp. 132-142, May. 1995.
- [37] The Economist, *Pocket world in figures 2011*. London: Economist and Profile Books, 2010.
- [38] N. Alyani, "A software strategy takes on growing urgency," *Iran Focus [Iran Strategic Focus]*, vol. 16, no. 10, pp. 8-11, 2003.
- [39] W. C. Kim and R. Mauborgne, "Blue ocean strategy," *Harvard Business Review*, vol. 82, no. 10, pp. 76-84, Oct. 2004.
- [40] R. C. Larson, "Service science: at the intersection of management, social, and engineering sciences," *IBM Systems Journal*, vol. 47, no. 1, pp. 41-53, 2008.
- [41] O. King and B. Mager, "Methods and processes of service design," *Touchpoint: the journal of service design*, vol. 1, no. 1, pp. 20-28, Apr. 2009.
- [42] P. Johnson, "Analytic induction," in *Essential guide to qualitative methods in organizational research*, C. Cassell and G. Symon, Eds. London: Sage Publications, 2004, pp. 165-179.
- [43] L. Bickman and D. J. Rog, "Applied research design: a practical approach," in *The SAGE handbook of applied social research methods*, 2nd ed., L. Bickman and D. J. Rog, Eds. London and Thousand Oaks, CA: SAGE, 2009, pp. 3-43.
- [44] P. Clawson, "U.S. sanctions," in *The Iran primer: power, politics, and U.S. policy*, R. Wright, Ed. Washington, DC: United States Institute of Peace Press, 2010, pp. 115-118.
- [45] M. H. Tayeb, "Human resource management in Iran," in *Human resource management in developing countries*, P. S. Budhwar and P. Debrah, Eds. London: Routledge, 2001, p. 121.
- [46] N. Alyani and T. Nahi, "Economic reform and the approaches to management and privatisation in Iran," *Iran Focus [Iran Strategic Focus]*, vol. 16, no. 5, pp. 9-12, 2003.
- [47] K.-S. Chang, "Compressed modernity and its discontents: South Korean society in transition," *Economy and Society*, vol. 28, no. 1, pp. 30-55, 1999.
- [48] IMF, *Islamic Republic of Iran: 2011 article IV consultation - Staff report; Public information notice on the executive board discussion; and statement by the executive director for Iran [August 2011 IMF CR 11/241]*. Washington, DC: International Monetary Fund, 2011.
- [49] P. Brown, H. Lauder, and D. Ashton, "EERJ Roundtable 2007: Education, globalisation and the future of the knowledge economy," *European Educational Research Journal*, vol. 7, no. 2, pp. 131-156, 2008.
- [50] G. B. Richardson, "The organisation of industry," *The Economic Journal*, vol. 82, no. 327, pp. 883-896, 1972.
- [51] D. Guile, "Education and the economy: rethinking the question of learning for the 'knowledge' era," *Futures*, vol. 33, no. 6, pp. 469-482, Aug. 2001.
- [52] B.-A. Lundvall, *The social dimension of the learning economy [DRUID Working Paper No. 96-1]*. Aalborg, Denmark: Department of Business Studies, Aalborg University, 1996.
- [53] F. Green, *What is skill? an inter-disciplinary synthesis*. London: Centre for Learning and Life Chances in Knowledge Economies and Societies (LLAKES), IOE, University of London, 2011.
- [54] B. Elkjaer, "Organizational learning: the 'third way'," *Management Learning*, vol. 35, no. 4, pp. 419-434, Dec. 2004.
- [55] J. S. Brown and P. Duguid, "Organizational learning and communities-of-practice: toward a unified view of working, learning, and innovation," *Organization Science*, vol. 2, no. 1, pp. 40-57, Jan. 1991.
- [56] J. S. Brown and P. Duguid, "Knowledge and organization: a social-practice perspective," *Organization Science*, vol. 12, no. 2, 2001.
- [57] G. Mason, K. Bishop, and C. Robinson, *Business growth and innovation: the wider impact of rapidly-growing firms in UK city-regions*. London: National Endowment for Science, Technology and the Arts (NESTA), 2009.
- [58] N. Alyani, "Learning to innovate collaboratively with technology: a study of workplace design projects in a telecoms services firm in Tehran and London [synopsis of PhD thesis at LLAKES, IOE, UL]," IOE DS Winter 2012 Conference, Institute of Education, University of London, forthcoming.
- [59] N. Alyani, "Learning to innovate collaboratively with technology: a study of workplace design projects in a telecoms services firm in Tehran [LLAKES DS symposium on 'challenges of learning economy in transitional countries']," 19-Jun-2010.
- [60] N. Alyani, D. Guile, Y. S. Jamjoom, E. Lamperti, S. Shirzad and A. Tehraninasr, *Learning to innovate during transition: policy-practice paradoxes of skill development in the emergent learning economies of Iran and Saudi Arabia [DS Working Paper 2011-2012]*. London: LCE, Faculty of Policy and Society, Institute of Education, University of London, forthcoming.
- [61] D. Leonard-Barton, *Wellsprings of knowledge: building and sustaining the sources of innovation*. Boston, MA: Harvard Business School Press, 1995.
- [62] D. Guile, "Learning through 'e-resources': the experience of SMEs," *Vocational Training: European Journal [CEDEFOP]*, vol. 27, no. 3, pp. 30-46, 2002.
- [63] S. L. Beckman and M. Barry, "Innovation as a learning process: embedding design thinking," *California Management Review*, vol. 50, no. 1, pp. 25-56, Fall. 2007.
- [64] T. Brown, "Design thinking," *Harvard Business Review*, vol. 86, no. 6, pp. 84-92, Jun. 2008.
- [65] K. M. Eisenhardt, "Building theories from case study research," *Academy of Management Review*, vol. 14, no. 4, pp. 532-550, 1989.
- [66] K. M. Eisenhardt and M. E. Graebner, "Theory building from cases: opportunities and challenges," *Academy of Management Journal*, vol. 50, no. 1, pp. 25-32, 2007.
- [67] D. Leonard-Barton, "A dual methodology for case studies: synergistic use of a longitudinal single site with replicated multiple sites," *Organization Science*, vol. 1, no. 3, pp. 248-266, Jan. 1990.
- [68] A. C. Edmondson and S. E. McManus, "Methodological fit in management field research," *Academy of Management Review*, vol. 32, no. 4, pp. 1155-1179, Oct. 2007.
- [69] D. Sole and A. C. Edmondson, "Situated knowledge and learning in dispersed teams," *British Journal of Management*, vol. 13, no. 2, p. S17-S34, 2002.
- [70] P. E. Waterson, C. W. Clegg, and C. M. Axtell, "The dynamics of work organization, knowledge and technology during software development," *International Journal of Human-Computer Studies*, vol. 46, no. 1, pp. 79-101, Jan. 1997.
- [71] J. Ritchie and J. Lewis, *Qualitative research practice: a guide for social science students and researchers*. London and Thousand Oaks, CA: Sage Publications, 2003.
- [72] R. K. Yin, *Case study research: design and methods*, 4th ed. Thousand Oaks, CA: Sage, 2009.
- [73] I. Nonaka and H. Takeuchi, "The wise leader," *Harvard Business Review*, vol. 89, no. 5, pp. 58-67, May. 2011.
- [74] S. Nerur and V. Balijepally, "Theoretical reflections on agile development methodologies," *Communications of the ACM*, vol. 50, no. 3, pp. 79-83, 2007.

# Configuring services regarding service environment and productivity indicators

Michael Becker, Stephan Klingner, Martin Böttcher

Department of Business Information Systems

Leipzig University, Germany

Email: {mbecker|klingner|boettcher}@informatik.uni-leipzig.de

**Abstract**—In course of the extensive changes in the service sector, methods and tools for modelling services, managing service-portfolios and optimising service-offers are required. This paper proposes an extension of a basic metamodel as described in various previous publications to be able to describe non-functional properties, global variables and the definition of configuration constraints.

## I. INTRODUCTION

**D**UE TO the widely acknowledged shift from the second to the third economic sector, services are of increasing importance. A growing market, driven, inter alia, by internationalisation, results in higher volumes of delivered services. At the same time complexity of provided services is increasing. This change in the quantitative as well as the qualitative dimension leads to the need for concepts, methods and models to conduct the engineering of services in a well-structured way [17]. Furthermore, both aspects in combination with a growing competitive environment require a stronger focus on productivity aspects of services.

This paper mainly presents an extension of various foundational concepts of a holistic metamodel for modelling, evaluating and optimising services, as described in multiple publications, such as [14]–[16]. Section II gives an aggregated overview of the different aspects of the metamodel described in the papers. Using a prototype, the scientific results were evaluated during several workshops. Subsequently this basic model was extended in accordance to the feedback of our industry partners. The extension comprises concepts that are required in real world applications but were not implemented in our service model so far. These include non-functional properties (*attributes*), global, system-wide definitions (*variables*) as well as the definition of rules as part of configuration queries (*constraints*).

In this work we present formalisations of these concepts and show their integration into the existing metamodel. Furthermore, we extend our metamodel to the ability of querying service components. To do so the remainder of this paper is structured as follows. Our method describes a two-step procedure, which consists firstly of modelling the service respectively the service portfolio (section II) and secondly of configuring the customised service offers based on that previously modelled portfolio (section III). These sections will recall necessary concepts for the formalisation of service components and extend our existing metamodel with attributes,

variables, and constraints. In section IV we present initial ideas and applications of querying services. Finally, sections V and VI give an overview about related work in this field and conclude the paper.

## II. MODELLING SERVICES

As mentioned in the introduction we presented initial ideas [15] and a formal approach [13] for modelling services based on components. Rather than recapitulating all concepts in detail we will briefly describe fundamental concepts that are necessary to understand the new concepts we introduce in this work. They are namely attributes to specify nonfunctional characteristics of components, external variables to detail the service environment, and constraints to refine valid service components during configuration. Using these extensions configurations can adapt customer wishes on specific aspects of components. Furthermore, they facilitate querying services.

In the course of this work we present the new modelling elements within the context of a small example portfolio shown in figure 1. In this example – based on a real world example of one of our industry partners – the described services are the provision of a call centre and the realisation of billing tasks. For this purpose, we define a component *CallCentre* encapsulating the general call centre provision process. This component is detailed by the subcomponents *CallCentre A*, *CallCentre B*, and *CallCentre C* defining specific steps for different call centre contractors. Due to comprehensibility reasons, the component *Billing* is not shown in full detail but only sketched. In the following section we describe the particular modelling concepts.

### A. Foundations

To comprehend the newly introduced concepts for service modelling it is necessary to recall some of the existing foundations we lay in past work. In [15] we have defined service components as an offering of a well-defined functionality via precisely described interfaces. Thus, a component represents a part of a service provision process. The relationship between the so-called configuration graph containing the components and its processual representation can be seen in figure 2. This figure shows the general call centre provision process consisting of the common activities *apply call numbers* and *train products* encapsulated in component *CallCentre*. Furthermore, it has three subprocesses *CallCentre A*, *CallCentre B*,

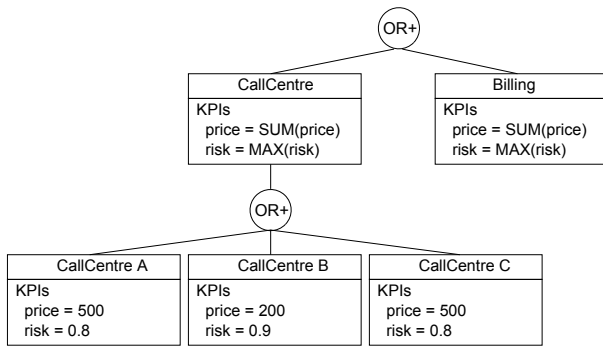


Figure 1. Example portfolio: of call centre provision and billing

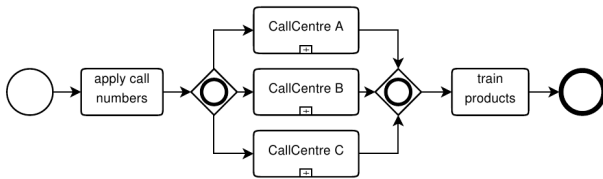


Figure 2. Call centre components represented as BPMN model

and *CallCentre C* containing specific activities for the three options.

Components can be used for composition (by combining components with other components) and decomposition (by separating existing components into distinct components). One application for composition is the creation of customer specific combinations by choosing components from a portfolio. Decomposition can be used to structure and detail existing services. These two possibilities are mapped to a process model representation using e.g. BPMN subprocesses.

Basically, a portfolio consists of a set of service components. We relate components with each other using connectors. Based on these concepts the definition of hierarchical dependencies between components is possible, e.g. component *A* consists of components *C* and *B*. These dependencies can be extended by cardinalities enabling statements about the required amount of subcomponents. For example, in the call centre use case the specific call centre components are connected using a disjunctive-obligatory connector ( $OR^+$ ) meaning that at least one of the child components must be selected. Since components represent processes, there are usually temporal dependencies between them. It is possible to integrate these dependencies using linear temporal logic enabling restrictions such as component *A* must be succeeded by component *B*. Furthermore, non-hierarchic dependencies between components are integrated using propositional logic. These dependencies are necessary to define additional constraints on service composition that cannot be displayed in the graph hierarchy, e.g. about mutually exclusive components. Additional details about connector semantics and both temporal and logical dependencies are presented in [16].

One goal of composition is to increase the productivity of services. Therefore, it is necessary to measure productivity in

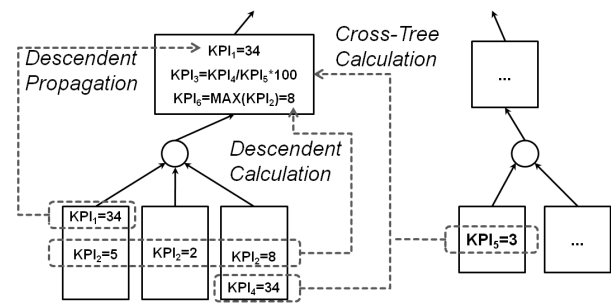


Figure 3. KPI inheritance strategies

some way. This is achieved using key performance indicators (KPIs) in components. In the call centre example we use the price (interpreted as costs to execute a component) and the risk (a lower number indicates a more mature process) as KPIs. Traditionally, productivity is defined as the fraction of output produced to inputs used. This approach is difficult for services since not only the input is hard to calculate but also the output [24]. However, this discussion is not in the focus of this work. In the component model, it is possible to use KPIs based on generic formulae.

KPIs are inherited through the service model in threefold manner. First, *Descendent Propagation* propagates a KPI to the parent component. Thus, the KPI characterises the parent component, too. Second, *Descendent Calculation* enables using KPIs of child components to calculate KPIs in parent components. Thus, in the use case in figure 1 the KPIs price and risk from the specific call centres can be accessed in the general call centre component. It is usually necessary to combine KPIs using arithmetic operators, e.g. the price of the general call centre provision is the sum of the specific prices. Another way to inherit KPIs is using *Cross-Tree Calculation* allowing to calculate KPIs based on KPIs from components that are not in hierarchical dependencies with each other. This allows for the representation of non-hierarchic component combinations on productivity. The different strategies for KPI inheritance are shown in figure 3.

### B. External Variables

Service provision cannot be seen in isolation from the environment where services are to be provided. For example, in the call centre use case the amount of expected incoming calls per day plays an important role both for the price of the overall service and the calculation of the risk of the service. Since they depend on customer characteristics and preferences, the values for these environmental impact factors are not known during modelling. In our model we use variables to represent characteristics of specific service environments.

External variables affect the whole modelled service system. While we only show the application for call centre components in our example, the amount of incoming calls might also influence other components, e.g. the processing of reshipment (since every 10th call may be a reshipment enquiry). Figure 4 shows the integration of variables to calculate specific KPIs. To



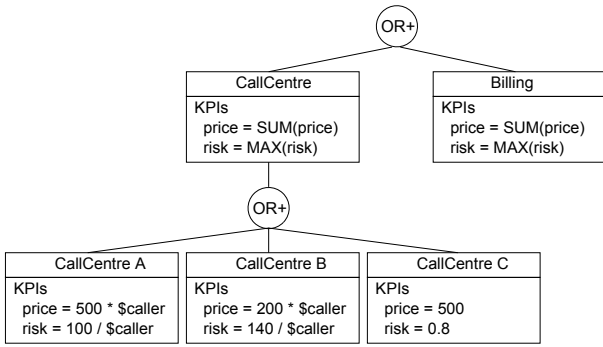


Figure 4. Example portfolio with variables

distinguish between variables and KPIs, variables are preceded by a dollar symbol. In the example the price and risk is calculated depending on the variable *caller*. Using variables facilitates the reusability of service components since KPIs do not need to be static and can also reference the service environment.

All used variables are formally represented in the set *VariableNames*. To allow greatest possible flexibility, variables are not restricted to specific data types. Since they represent the service environment, variables affect all components in the portfolio and have a global namespace. Thus, the variable *caller* in figure 4 has the same meaning both in component *CallCentre A* and in component *CallCentre B*. The values of variables are defined during configuration of customer specific services as shown in section III-B.

### C. Attributes

Nonfunctional properties of services are of great importance to define service prospects and limitations. For example, a call centre might have a capacity limit of manageable calls per day. Though no restriction on the functionality itself, this is an important information about the capabilities of a component. Therefore, to enable meaningful configurations, it is necessary to specify nonfunctional properties in a consistent and formal way.

Especially in the domain of web services there exist a variety of different approaches to represent nonfunctional properties, e.g. [1], [36], [40]. However, for the sake of brevity we omit formal attribute definition and represent nonfunctional properties using simple free text attributes of service components. Based on these attributes, further constraints can be formulated to allow and disallow specific components, c.f. III-C. An extensive discussion about possible non-functional properties is conducted in [32]. Amongst others, the properties temporal and local availability, price, payment methods, penalties, right, and obligations are presented. Based on a given set of nonfunctional properties the restrictions of service components can be described very detailed.

The formal representation of attributes is similar to the representation of KPIs. They are determined by the sets *AttributeNames* and *AttributeValues* containing the names and values of used attributes. To connect a component with attributes we

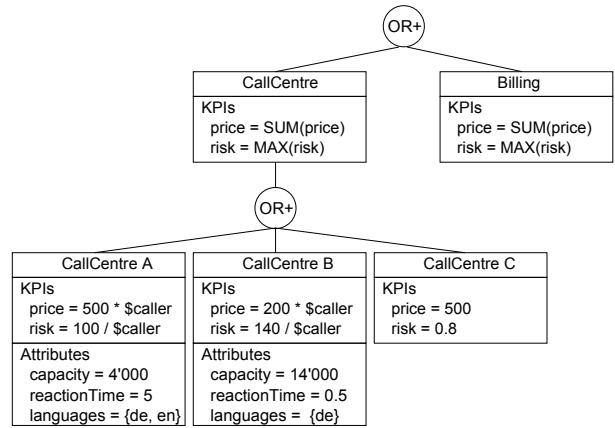


Figure 5. Example portfolio with variables and attributes

use the mapping *AttributeMapping* defined as follows.

$$\begin{aligned} & \textit{AttributeMapping} : \\ & \textit{Components} \rightarrow P(\textit{AttributeName} \times \textit{AttributeValue}) \end{aligned} \quad (1)$$

To simplify attribute access during configuration we additionally define the mapping *AttributeValue*.

$$\begin{aligned} & \textit{AttributeValue} : \\ & \textit{Components} \times \textit{AttributeName} \rightarrow \textit{AttributeValue} \end{aligned} \quad (2)$$

Attributes describe a specific component (and thereby characteristics of the underlying process). An attribute itself is represented by an arbitrary data type. The visualisation of attributes in the service model as well as its formal representation according to the definition of the mapping *AttributeMapping* is shown in figure 5. Both call centres are enriched with the attributes capacity and reaction time represented by numerical values and the attribute languages represented by a set of values. To clarify the difference between KPIs and attributes, we added a visual distinction between them in the components.

The formal representation of the attributes used in the call centre use case is as follows.

$$\begin{aligned} & \textit{AttributeName} = \\ & \{ \textit{capacity}, \textit{reactionTime}, \textit{languages} \} \\ & \textit{AttributeValue} = \\ & \{ 4'000, 14'000, 5, 0.5, \textit{de}, \textit{en} \} \\ & \textit{AttributeMapping}(\textit{CallCentreA}) = \\ & \{ (\textit{capacity}, 4'000), (\textit{reactionTime}, 4), \\ & \quad (\textit{languages}, \{ \textit{de}, \textit{en} \}) \} \\ & \textit{AttributeMapping}(\textit{CallCentreB}) = \\ & \{ (\textit{capacity}, 14'000), (\textit{reactionTime}, 0.5), \\ & \quad (\textit{languages}, \{ \textit{de} \}) \} \end{aligned}$$

In contrast to KPIs, attributes have a fixed value defined during modelling phase. Furthermore, they are not automatically inherited through the service model. This restriction is



necessary because during modelling time it is not clear how attributes will be combined during configuration. Depending on customer preferences, different combinations are possible. In the call centre example, one customer might be able to combine different call centres to increase the capacity where other customers do not allow this combination. However, it is possible to use attributes in the calculation of KPIs, e.g. the monthly costs of a call centre may depend on the dimensions of a rented office space. Thus, attributes can be used in descendent and in cross-tree calculations as shown in figure 3 for KPIs but are not propagated.

It is not necessary to define attributes for every component in the model. For example, in figure 5 *CallCentre C* has no attributes. Missing attributes can occur when additional details about a specific service offer are not (yet) known. Therefore, missing attributes can be (but do not have to be) an indicator for underspecified and not well-known components in the portfolio.

### III. CONFIGURING SERVICES

Until now we have shown the integration of external variables representing the service environment and the description of nonfunctional properties of service components using attributes. Based on these concepts in this section we move on from modelling service portfolios to configuration of customer specific service offers considering KPIs of services. For comprehensibility we will recall configuration foundations in the next section.

#### A. Foundations

During service modelling, components and their dependencies between each other are defined. The formal definition of components can be used in twofold manner. First, it is possible to generate predefined service bundles consisting of different components. However, real benefit is achieved when customer specific configurations are generated. During configuration, components are selected and combined. Due to the formal representation of dependencies it is possible to verify the validity of service offers. The configuration is one approach to tackle the opposition between standardisation to reduce costs and customisation to offer a flexible portfolio. Due to aggregation of KPIs the performance of combined services can be assessed. We presented a formal definition of the configuration in [9].

If external variables are used during modelling it is necessary to define their values for a specific configuration. This is shown in section III-B. Furthermore, in section III-C we present a way to automatically constraint the selection of service components. Constraints can be used as restrictions on configurations.

#### B. Variable value definition

In section II-B, we stated that external variables have a global namespace. Therefore, the definition of their values is

straightforward using the mapping *VariableValue* as follows.

$$\begin{aligned} & \text{VariableValue} : \\ & \text{VariableName} \rightarrow \text{VariableValues} \end{aligned} \quad (3)$$

The call centre example consists of only one external variable *caller* representing the expected amount of incoming calls per day. Using  $\text{VariableValue}(\text{caller}) = 7000$  we set its value to 7000. After referenced variables have been set to a fixed value the calculation of KPIs for components using these variables is possible. Therefore, the price of component *CallCentre A* is set to 3'500'000 and its risk is set to 1/70. Analogously, *CallCentre B* has a price of 1'400'000 and a risk of 1/50.

#### C. Constraints

Using the above mentioned concepts of component attributes and environment variables as well as the already known performance indicators we are now able to formulate detailed constraints on service components. We consider constraints as a filter to choose only suitable components. For example, in the call centre scenario a typical constraint states that the capacity of a call centre must be greater or equal to the amount of incoming calls. This is an essential requirement for successful service provision.

During configuration it is possible to distinguish between hard and soft constraints. While members of the first one have to be satisfied because otherwise service provision is impossible, the latter ones should be satisfied because they are a potential risk during provision. However, we cannot define the type of a constraint in advance. For example, for a call centre provider it is of utmost importance to satisfy the capacity constraints mentioned above. In contrast to this, for a service provider offering but not focusing on call centre provision this constraints may not be as equally important. Finally, one service provider may have different clients emphasising service aspects in different ways. To overcome this challenge, all possible constraints on service models are specified in one set *Constraints* and during configuration it is possible to define which constraints are hard and which are soft by assigning them to the sets *HardConstraints* and *SoftConstraints* where unassigned constraints are automatically considered as being soft. In the next paragraphs, we show the definition and evaluation of constraints.

1) *Definition of constraints*: To allow for greatest possible freedom in constraint formulation the constraints are based on predicate logic. This is similar to the concepts used in feature models in software product lines, c.f. [29]. Constraints can be defined over values of attributes or KPIs. We use the mappings *AttributeValue* defined in section II-C and *KPIValue* analogously defined in [9] to access values of attributes and KPIs. The capacity constraint for components mentioned above combined with a price constraint is specified with the

following formula using the generic identifier *Components*.

$$\begin{aligned} & \forall c \in \text{Components} : \\ & \text{AttributeValue}(c, \text{capacity}) \geq 7000 \wedge \\ & \text{KPIValue}(c, \text{price}) \leq 1'100 \end{aligned} \quad (4)$$

To increase reusability of constraints, in addition to using fixed values it is also possible to use variables. This results in a simplified configuration since variables have to be entered only once and can be used both for constraint and KPI definition. The capacity constraint referencing the variable *caller* can be formulated as follows.

$$\begin{aligned} & \forall c \in \text{Components} : \\ & \text{AttributeValue}(c, \text{capacity}) \geq \$\text{caller} \end{aligned} \quad (5)$$

More often than not, it will be the case that different unrelated components are characterised by attributes with equal names. However, these attributes may not have the same meaning, e.g. the attribute *capacity* might also be used in different components that are not relating it to the expected amount of incoming calls. Therefore, developers need techniques to distinguish between these model parts. This is achieved by using a specific component name in the quantifier of a constraint. For example, the constraint

$$\begin{aligned} & \forall c \in \text{CallCentre} : \\ & \text{AttributeValue}(c, \text{capacity}) \geq \$\text{caller} \wedge \\ & \text{KPIValue}(c, \text{price}) \leq 1'100 * \$\text{caller} \end{aligned} \quad (6)$$

will only be evaluated for components that are in the transitive closure of child components of the component with the identifier *CallCentre*. The transitive closure of a component contains all its direct and indirect child components. As the service configuration graph is a tree and thus a specialisation of a directed graph, existing algorithms to calculate the transitive closure for graphs can be applied. A selection is, for example, shown in [26].

2) *Evaluation of constraints*: During constraint evaluation it is necessary to distinguish between constraints over attributes and constraints over KPIs. The former constraints are checked against the set of components contained in the transitive closure as defined above. The latter ones are checked against the complete configuration. Thus, different evaluation strategies are applied. In the course of this section we refer to the capacity constraint defined in equation 6 and set the variable *caller* to 7'000.

Constraints over attributes affect single components. For every component that is referenced in the constraint it is a) checked, whether the component contains the mentioned attribute and b) verified, whether the component fulfils the constraint. Therefore, it is possible to clearly identify components that violate attribute constraints. For example, the capacity constraint is violated in component *CallCentre A* since its capacity is only 4'000. On the other hand, it is fulfilled in *CallCentre B* because the capacity is more than 7'000 and in *CallCentre C* because this component does not have an attribute *capacity* and accordingly the constraints is

not checked against this components. If an attribute constraint is violated it is violated throughout the whole configuration process regardless of what other components are selected. The only ways to fix a violated attribute constraint is to deselect the respective component. Depending on the type of constraint the configuration will either be invalid (hard constraint) or have warnings (soft constraints).

On the contrary, constraints over KPIs are evaluated at the topmost occurrence of the KPI. The KPIs are identified by a breadth-first search and comparison is based on their name. Thus, the capacity constraint is evaluated at the component *CallCentre*. Since KPIs are usually a combination of underlying KPIs it is not possible to clearly identify components that violate KPI constraints. The capacity constraint states that the price in the component *CallCentre* has to be less than or equal to 1'100 per caller. Selecting *CallCentre A* and *CallCentre B* this constraint is still fulfilled, while adding *CallCentre C* violated the constraint. However, it is not possible to determine the violating component since deselecting *CallCentre A* would fulfil the constraint. In summary, a violated KPI constraint can be fixed by different configurations even containing the last selected component that lead to the violation.

#### IV. QUERYING SERVICES

Based on the previously developed formalisations for service modelling it is possible to structure extensive service portfolios. This is to a great extent focused on the viewpoint of service providers. However, from a customer point of view different considerations have to be taken into account. In provider-driven configuration of customer specific configurations, customers can only select from services of one provider. The configuration process is usually guided by the provider to support the component selection. But on electronic service markets customer are not aware what services exist and what services meet their specific requirements. Therefore, they need ways to identify suitable service components.

Finding suitable components is a problem with a long history in software engineering when existing software has to be reused. On this account, [28] has introduced four general techniques for reusing software artefacts that can be transferred to service engineering, too. First, *abstraction* is the basic reuse technique and enables developers to comprehend different artefacts. Because we only reuse service components, abstraction is inherently present in our model. Second, by *selection* developers are supported in the process of selecting different artefacts. Heretofore, we support selection by customer specific configuration. However, for the selection process itself there is no support since customers need to find suitable services on their own. By using queries over service components we are looking forward to improve the selection process. The two remaining techniques are *specialisation* to represent similar artefacts in a consolidated way and *integration* to support developers during assembling of reused artefacts to complete systems. While the latter one can be achieved using the formal metamodel of our service model,

the first one is possible by integrating predefined component types into the model.

Queries over service components are used to search through a collection of components and automatically identify components that match specific customer requirements. As stated in section III-C constraints are one approach to capture requirements and validate the adherence of components against requirements. A challenge to keep in mind is the often occurring gap between query formulation and component description identified in [25]. Generally, component descriptions focus on the functionality by answering the question how a component works. In our model this fact is typically represented by underlying process models. On the other hand, queries usually focus on the problem itself and ask what components are capable of solving a problem. It is possible to close (or at least reduce) this gap by making extensive use of the above presented concept of component attributes. Though it increases initial modelling effort, the detailed description of component prospects and limitations supports efficient and accurate component search.

The definition of queries focuses the ability of customers to find service components they can use. Therefore, it is necessary that query formulation is simple and not too complex. The structure of queries has to adhere to our underlying service component metamodel. Similar as in constraint definition based on predicate logic the query language must provide selectors to select the components that are affected by a query. Furthermore, it must be possible to define constraints that are matched against components. A source of inspiration for queries might be the well-known database query language SQL, e.g. presented in [18].

## V. RELATED WORK

In response to a growing complexity, a more individualised market and an economically motivated stronger focus on productivity aspects, the division of monolithic tasks in manageable components is being utilised in various areas. Although in other scientific disciplines this process is called *modularisation* we refer to *components* as a synonym for *modules*. Particularly in the field of industrial engineering the concepts of modelling complex structures with the use of smaller components is quite established [5]. Similarly in software engineering the use of components is employed on a broad base [37]. A hybrid form of software and classical services are IT-services, where modularisation is applied likewise [10].

Based on the modularisation, service configuration is enabled. Configuration is well known in industrial engineering [38] and in software engineering [20]. In service engineering, configuration gains importance as well [3]. However, existing approaches like [2] do not focus dependencies between service components and customer-driven configuration but rather on formalising overall service systems.

In the course of this work we presented one approach to define constraints on service components. Boustil et al. stated that existing approaches for service selection do not integrate user requirements [12]. In the domain of web services, Tosic

et al. present an approach to define constraints [39]. In order to so, they introduce an XML-based constraint language called Web Service Offerings Language. Yu et al. use a graph based approach to select web services adhering to specific QoS constraints [41]. The academic literature has produced a variety of approaches to formulate queries for services. However, most of them are tailored to web services and do not take the specialities of complex business services into account. We tackle this problem by using the underlying formal metamodel for the specification of service components. Most existing query approaches focus on one specific aspect of services. For example, Baraka developed a query language for mathematical services [6], [7]. It works in conjunction with a description language for mathematical services presented in [8]. The approach also uses SQL-like query statements. Another stream of researches focus on the query for resources [23], quality of service aspects [21], [30], and service mash-up [22]. The integration of different service aspects is in the focus of the work of Pantazoglou et al. They present a query language [33], a matching algorithm [35], and an associated engine [34]. Though this approach targets more than one aspect its application is limited to web services. To facilitate optimal service selection, Bonatti and Festa have examined different approaches in [11].

Components in our service model represent process models. Therefore, query languages and constraint definition for process models can be an interesting object of study, too. Awad has presented a visual query language for BPMN models [4] and Jin et al. query process repositories using graph isomorphisms [27]. To specify requirements (e.g. constraints) on process models, Momotko et al. developed an integrated methodology. Therefore, they developed a metamodel for a process description language [31].

## VI. CONCLUSION

In this research paper we presented the extension of a previously defined service component model with variables and attributes. These concepts can be used to formulate constraints on the customer specific configuration of components. Based on these constraints it is possible to create more meaningful configurations. We are currently applying the concepts contained in the extended service component metamodel together with our industry partner in a real world scenario. It has shown that proper tool support is of utmost importance to manage complex service portfolios. Therefore, we have developed a prototype supporting the modelling and configuration of service components.

Furthermore, we investigated challenges that arise when the service selection process is customer-driven without guidance by service providers. A great obstacle for realising service component markets is the existing gap between service descriptions and the representation of customer needs. In our approach this gap is reduced by the extensive use of attributes and the ability to query for service components based on constraints.

Using attributes and variables in constraints and queries depends to a great extent on a homogeneous taxonomy. Till now integration of existing components in a portfolio needs a careful check of used attributes and variables. This problem is aggravated when customers query marketplaces with components from different providers. Generally, different providers use different taxonomies what results in poor component recalls. One approach to tackle this problem is the Unified Service Description Language (USDL), presented in [19]. This would allow for a consistent definition of service components. Another approach might be predefining service characteristics for specific components and establish service types based on these characteristics. Selection of components can then be based on predefined service types.

In addition, increasing query capabilities requires integrating a holistic approach using various modelling concepts. In this work we presented queries based on attributes and KPIs. As components are based on processes, existing approaches from this domain can be examined for their applicability. Furthermore, integration of resources is an important point to keep in mind. More often than not it will be the case that components need specific inputs—querying for components producing this input as output will be challenging.

#### REFERENCES

- [1] Stephan Aier, Philipp Offermann, Marten Schönherr, and Christian Schröpfer. Implementing non-functional service descriptions in soas. In Dirk Draheim and Gerald Weber, editors, *Trends in Enterprise Application Architecture*, volume 4473 of *Lecture Notes in Computer Science*, pages 40–53. Springer Berlin / Heidelberg, 2007.
- [2] Hans Akkermans, Ziv Baida, Jaap Gordijn, Nieves Pena, Ander Altuna, and Inaki Laresgoiti. Value webs: Using ontologies to bundle real-world services. *IEEE Intelligent Systems*, 19:57–66, 2004.
- [3] Luis Araujo and Martin Spring. Complex performance, process modularity and the spatial configuration of production. In N. Caldwell and M. Howard, editors, *Procuring Complex Performance: Studies in Innovation in Product-Service Management*. Routledge, London, 2010.
- [4] Ahmed Awad. Bpmn-q: A language to query business processes. In *Enterprise Modelling and Information Systems Architectures - Concepts and Applications*, *Proceedings of the 2nd International Workshop on Enterprise Modelling and Information Systems Architectures*, pages 115–128, 2007.
- [5] Carliss Y. Baldwin and Kim B. Clark. Managing in an age of modularity. *Harvard Business Review*, 1997.
- [6] Rebhi Baraka. Mathematical services query language: Design, formalization, and implementation. Technical report, Johannes Kepler University, Linz, Linz, Austria, September 2005.
- [7] Rebhi Baraka and Wolfgang Schreiner. Querying registry-published mathematical web services. *Advanced Information Networking and Applications, International Conference on*, 1:767–772, 2006.
- [8] Rebhi S. Baraka. *A Framework for Publishing and Discovering Mathematical Web Services*. Dissertation, Johannes Kepler Universität Linz, Linz, August 2006.
- [9] Michael Becker. Formales metamodel für dienstleistungskomponenten. Technical report, Universität Leipzig, 2011. to appear.
- [10] Tilo Böhm, Richard Gottwald, and Helmut Krcmar. Towards mass customized it services: Assessing a method for identifying reusable service modules and its implication for it service management. In *AMCIS 2005 Proceedings*, 2005.
- [11] P. A. Bonatti and P. Festa. On optimal service selection. In *Proceedings of the 14th international conference on World Wide Web*, WWW '05, pages 530–538, New York, NY, USA, 2005. ACM.
- [12] Amel Boustil, Nicolas Sabouret, and Ramdane Maamri. Web services composition handling user constraints: towards a semantic approach. In *Proceedings of the 12th International Conference on Information Integration and Web-based Applications & #38; Services*, iiWAS '10, pages 913–916, New York, NY, USA, 2010. ACM.
- [13] Martin Böttcher and Klaus-Peter Fähnrich. Service systems modeling: Concepts, formalized meta-model and technical concretion. In Haluk Demirkan, James C. Spohrer, and Vikas Krishna, editors, *The Science of Service Systems*. Springer, New York et al., 2011.
- [14] Martin Böttcher and Stephan Klingner. Der Komponentenbegriff der Dienstleistungsdomäne. In K.-P. Fähnrich and B. Franczyk, editors, *Informatik 2010—GI Jahrestagung*, volume 1, pages 59–66, Leipzig, 2010. Lecture Notes in Informatics (LNI).
- [15] Martin Böttcher and Stephan Klingner. The basics and applications of service modeling. In *SRII Global Conference 2011*, 2011. to appear.
- [16] Martin Böttcher and Stephan Klingner. Providing a method for composing modular b2b-services. *Journal of Business & Industrial Marketing*, 2011. To appear.
- [17] Hans-Jörg Bullinger, Klaus-Peter Fähnrich, and Thomas Meiren. Service engineering - methodical development of new service products. *International Journal of Production Economics*, 85:275–287, 2003.
- [18] S. Cannan and G. Otten. *SQL—The standard handbook: based on the new SQL standard*. McGraw-Hill, 1993.
- [19] Jorge Cardoso, Alistair Barros, Norman May, and Uwe Kylau. Towards a unified service description language for the internet of services: Requirements and first developments. *Services Computing, IEEE International Conference on*, 0:602–609, 2010.
- [20] Krzysztof Czarnecki, Simon Helsen, and Ulrich Eisenecker. Staged configuration using feature models, 2004.
- [21] Giuseppe Damiano, Ester Giallonardo, and Eugenio Zimeo. onqos-ql: A query language for qos-based service selection and ranking. In Elisabetta Di Nitto and Matei Ripeanu, editors, *Service-Oriented Computing - ICSSOC 2007 Workshops*, volume 4907 of *Lecture Notes in Computer Science*, pages 115–127. Springer Berlin / Heidelberg, 2009.
- [22] Weilong Ding, Jing Cheng, Kaiyuan Qi, Yan Li, Zhuofeng Zhao, and Jun Fang. A domain-specific query language for information services mash-up. *Services, IEEE Congress on*, 0:113–119, 2008.
- [23] Sebastian Günther, Claus Rautenstrauch, and Niko Zenker. Service-oriented architecture: Introducing a query language. In Martin Bichler, Thomas Hess, Helmut Krcmar, Ulrike Lechner, Florian Matthes, Arnold Picot, Benjamin Speitkamp, and Petra Wolf, editors, *Multikonferenz Wirtschaftsinformatik*. GITO-Verlag, Berlin, 2008.
- [24] Christian Grönroos and Katri Ojasalo. Service productivity: Towards a conceptualization of the transformation of inputs into economic results in services. *Journal of Business Research*, 57(4):414–423, 2004. European Research in service marketing.
- [25] Scott Henninger. Using iterative refinement to find reusable software. *IEEE Software*, 11:48–59, 1994.
- [26] Yannis Ioannidis, Raghu Ramakrishnan, and Linda Winger. Transitive closure algorithms based on graph traversal. *ACM Trans. Database Syst.*, 18:512–576, September 1993.
- [27] Tao Jin, Jianmin Wang, Marcello La Rosa, Arthur H.M. ter Hofstede, and Lijie Wen. Efficient querying of large process model repositories. Report 39060, Queensland University of Technology, December 2010.
- [28] Charles W. Krueger. Software reuse. *ACM Comput. Surv.*, 24:131–183, June 1992.
- [29] Marcilio Mendonca, Andrzej Wasowski, and Krzysztof Czarnecki. Sat-based analysis of feature models is easy. In *Proceedings of the 13th International Software Product Line Conference*, SPLC '09, pages 231–240, Pittsburgh, PA, USA, 2009. Carnegie Mellon University.
- [30] Delnavaz Mobedpour, Chen Ding, and Chi-Hung Chi. A qos query language for user-centric web service selection. *Services Computing, IEEE International Conference on*, 0:273–280, 2010.
- [31] Mariusz Momotko and Kazimierz Subieta. Process query language: A way to make workflow processes more flexible. In András Benczúr, János Demetrovics, and Georg Gottlob, editors, *Advances in Databases and Information Systems*, volume 3255 of *Lecture Notes in Computer Science*, pages 306–321. Springer Berlin / Heidelberg, 2004.
- [32] Justin James O'Sullivan. *Towards a precise understanding of service properties*. PhD thesis, Queensland University of Technology, Faculty of Information Technology, 2008.
- [33] Michael Pantazoglou and Aphrodite Tsalgatidou. The unified service query language. Technical report, National and Kapodistrian University of Athens, Athens, Greece, July 2009.
- [34] Michael Pantazoglou, Aphrodite Tsalgatidou, and George Athanasopoulos. Discovering web services and jxta peer-to-peer services in a unified manner. In Asit Dan and Winfried Lamersdorf, editors, *Service-Oriented Computing—ICSSOC 2006*, volume 4294 of *Lecture Notes in Computer Science*, pages 104–115. Springer Berlin / Heidelberg, 2006.

- [35] Michael Pantazoglou, Aphrodite Tsalgatiidou, and George Athanasopoulos. Quantified matchmaking of heterogeneous services. In Karl Aberer, Zhiyong Peng, Elke Rundensteiner, Yanchun Zhang, and Xuhui Li, editors, *Web Information Systems—WISE 2006*, volume 4255 of *Lecture Notes in Computer Science*, pages 144–155. Springer Berlin / Heidelberg, 2006.
- [36] Stephan Reiff-Marganiec, Hong Yu, and Marcel Tilly. Service selection based on non-functional properties. In Elisabetta Di Nitto and Matei Rippeanu, editors, *Service-Oriented Computing - ICSC 2007 Workshops*, volume 4907 of *Lecture Notes in Computer Science*, pages 128–138. Springer Berlin / Heidelberg, 2009.
- [37] Clemens Szyperski. *Component software - beyond object-oriented programming*. Addison-Wesley, London et al., 2002.
- [38] J. Tiihonen and T. Soinen. Product configurators—inforamtion system support for configurable products. In T. Richardson, editor, *Using Information Technology During the Sales Visit*. Hewson Group, Cambridge, 1997.
- [39] Vladimir Tasic, Kruti Patel, and Bernard Pagurek. Wsol—web service offerings language. In Christoph Bussler, Richard Hull, Sheila McIlraith, Maria Orłowska, Barbara Pernici, and Jian Yang, editors, *Web Services, E-Business, and the Semantic Web*, volume 2512 of *Lecture Notes in Computer Science*, pages 57–67. Springer Berlin / Heidelberg, 2002.
- [40] Hiroshi Wada, Junichi Suzuki, and Katsuya Oba. Modeling non-functional aspects in service oriented architecture. *Services Computing, IEEE International Conference on*, 0:222–229, 2006.
- [41] Tao Yu, Yue Zhang, and Kwei-Jay Lin. Efficient algorithms for web services selection with end-to-end qos constraints. *ACM Trans. Web*, 1, May 2007.

## Service quality description – a business perspective

Marija Bjeković, Sylvain Kubicki  
Public Research Centre Henri Tudor  
29, avenue JF Kennedy  
L-1855 Luxembourg Kirchberg  
Luxembourg  
Email: {firstname.lastname}@tudor.lu

**Abstract**—Business fields characterized by collective activities are numerous and require well-adapted software-based services to improve the efficiency of business collaborations. The design of services supporting the activities in such domains is usually ad-hoc and relies on the know-how of various involved actors. Based on our experience of designing innovative services for Architecture, Engineering and Construction projects, we proposed a service design method involving business actors, service and technical experts and being intrinsically collective. This article focuses more precisely on integrating non-functional (i.e. service quality) aspects of services in such an approach. The service-business practices alignment should not only be tackled from functional but as well non-functional perspective, so that not only business-level service quality requirements are clearly understood and taken care of, but as well that business practitioners get a clear view of all the characteristics of the designed service. While concepts referring to the technical service quality are well-known and vastly used by service experts, what are the concepts defining service quality in terms of specific business context remains an issue to be addressed both by domain practitioners and service experts. We propose in this article an initial business-level service quality model, aimed at qualifying services for construction projects.

### I. INTRODUCTION

IN THE age of computing and Information Technology (IT), Architecture, Engineering and Construction (AEC) firms (i.e. largely SMEs) are increasingly making use of internal IT systems to improve their competitiveness. When coming to the use of project management support systems, practitioners have to deal with the uncertainty of construction projects organization. Indeed, these projects involve numerous practitioners for short durations and in various contractual contexts. Work processes cannot really be pre-defined and have to remain flexible enough to fit real projects context. Moreover, service innovation R&D projects often lead to the development of prototypes or demonstrators that only partially reflect the reality of the future final services that will be used by business practitioners. When dealing with “experimental protocols” of service prototypes in these projects, researchers have to find ways to assess quality expectations of future services.

This work has been supported by the Dest2Co project funded by Fonds National de la Recherche, Luxembourg

### A. Context

Platforms providing project management services have to be flexible enough to support business actors’ activities in collaborative projects, especially in AEC ones. One can observe however that in numerous cases, AEC project management platforms are not surviving the duration of a construction project because they fail to provide adequate support for business processes [1].

Service science brings prospects to such situations. In collaborative business environments the design of services is nowadays often ad-hoc, or relying on the know-how of the involved actors. Service science-based methods are then supposed to increase the success of innovation in services for highly collaborative environments, an example of which are construction projects. From our previous experience when we were led to design innovative services for these projects [2], we noticed the paramount importance of aligning services to business practices. In order to formalize the collective business practices (i.e. the ones involving multiple actors) in these projects together with domain actors, we guided them to agree on the way collaborative tasks are most often performed. Although the process was difficult and required numerous working groups to be held, we managed to define a set of consensual best practices, i.e. collective practices.

### B. Design of innovative services for construction projects

Discussed applied work allowed us to understand that service innovation requires a clear understanding of business activities (i.e. practices). Service design should then be intrinsically collective, involving business actors, service and as well technical experts. This would, on one side, enable business actors to refine the requirements regarding the support of their activities, at the same time allowing them a better understanding of services capability. On the other side, the adaptability of IT services can only be successfully reached if domain knowledge has been appropriately considered at the service design stage already.

This lets us envisage that prototyping technical services could be defined for demonstration/experimental purposes before the final service is developed, provided and maintained. In addition, besides functional aspect, each prototype service should be associated, to the extent of possible, with non-functional properties of the future real service. The aim is that final service is clearly understood by business experi-

menters (i.e. early adopters) or that their expectations regarding future service quality are clearly understood by researchers to favor the (commercial) transfer conditions.

Indeed, the development of collaboration amongst business enterprises reinforces the need for an integration of non-functional aspects from the highest level of description of the system (business perspective): in a service-oriented business system, business partner delivering the service remains indeed the owner of the asset; there is no transfer of ownership but only an exchange of information. The business partner consuming the service has therefore to clearly state its expectations in terms of operational level of quality in order to manage its own business risks.

In this article, we discuss how non-functional aspects of services can be integrated in the design of innovative services for highly-collaborative construction projects. We discuss different models for describing them and motivate the need of addressing the quality concern differently at business and technical levels. Whereas technical-level initiatives proposing service quality (reference) models are numerous, research work tackling the same topic from the business perspective is not abundant. We propose an initial business-level service quality model dedicated to highly-collaborative contexts, which is developed in the scope of an ongoing research project (Dest2Co).

The rest of the article is organized as follows: the section 2 introduces the service design method developed within Dest2Co project, with the focus on discussing models for describing service quality. The section 3 presents the synthesized view on relevant related work, and the proposition of business-level quality taxonomy is presented in the section 4. After having discussed a possible application scenario in the section 5, the conclusion will highlight our prospects regarding this topic.

## II. SERVICE QUALITY WITHIN DEST2CO METHOD

Dest2Co project (2009-2011) aims at defining a model-based and multi-viewpoints method for the design of services for highly collaborative context. The method takes into account specific concerns of actors involved in service design through the following viewpoints:

- Business requirements viewpoint (BRV) addresses business concerns such as: what are business practices to be applied, what are the needs associated to the use of practices in the business context. This viewpoint is more focused on global objectives than on the realization of services. The central concept to BRV is the one of collective practices [5], whose identification for a given collective situation (i.e. project) is the first step towards the elicitation of requirements.
- Business solution viewpoint (BSV) focuses on how the practice can be accessed as a service, whether some existing practices are already realized as services, what are required interactions to support the use of the service, what is the level of service guaranteed by the provider and required by the consumer etc.
- Technical solution viewpoint (TSV) consists in the view of IT experts on the service. It is created based on the BSV and taking into account technical specificities, architectures, etc.

The three viewpoints are formalized through metamodels and organized according to the architectural framework [ISO/IEC 42010:2007], which aims to organize the models of complex systems. Moreover, each viewpoint covers multiple service aspects so as to fully support the design of services: 1) functional, 2) non-functional and 3) transactional aspect. The detailed presentation of Dest2Co method can be found in [4]. The present article focuses on defining the service quality model that reflects specific non-functional aspects of services for highly collaborative (construction) projects.

The quality of service (QoS) is the umbrella concept abstracting some non-functional aspects of the service. Research in service engineering has seen a number of initiatives aiming to define the concepts covered by QoS, (i.e. quality model, quality taxonomy) [8] [9] [11] as well as to define a language (i.e. metamodel) for expressing service quality characteristics and constraints [8] [10]. Current research aims to exploit semantic technologies to enable the automatic evaluation of QoS of already deployed (web) service, so that services required by the consumer can be dynamically discovered, selected and composed [18][19].

However, in most of the current initiatives, service quality is addressed from predominantly technical perspective, with very few works tackling non-functional aspects of service from the early stages in service development (i.e. from business perspective). While concepts used to express the technical quality of service are clearly understood and vastly used by service technical experts, what concepts define service quality from the business point of view in the specific business context remains an issue to be addressed both by domain practitioners and service experts.

Therefore, in the scope of Dest2Co approach, existing reference models defining “technical” service quality factors can easily be reused at TSV, but can’t be directly transposed for business-level qualification of the service during service design (BRV, BSV).

We propose a business-level service quality model to be used within Dest2Co service design method, and eventually for qualifying services in other highly-collaborative contexts. This model is strongly driven by the extensive experience of the experts we have. Nonetheless, it also takes into account the state-of-the-art work on the topic of service quality (seen from different perspectives). The related work is summarized in the following section.

## III. RELATED WORK ON SERVICE QUALITY MODELING

As already mentioned, a number of service quality reference models exist nowadays. Those adopting technical perspective on quality rely on [ISO/IEC 9126-1:2001] or [ISO/IEC 25010:2011] software quality measurement standards. In addition, most quality models we analyzed have the hierarchical organization as in these ISO standards. Some reference models define service quality attributes as general in order to be applicable across different domains. This in turn necessitates a certain adaptation of the model when applied in the specific domain. Such is the case with QoS catalog, defined within OMG QFTP specification [8]. QoS cata-



log aims to provide concepts for the qualification of services that are common to various domains and projects. Defined general QoS categories include: *Performance*, *Dependability* (comprising reliability, availability, safety and integrity), *Security*, *Integrity*, and *Coherence* (details on characteristics defined within these categories can be found in [8]). These QoS categories should be extended or specialized for the specific domain, so as to reflect its relevant non-functional requirements. Therefore each domain/project would have its own specific QoS catalog. Within QFTP specification, OMG's QoS catalog is specialized for the real-time and high-confidence systems.

Business-related quality aspects are not always left out of the service quality models: current draft version (v2.0) of OASIS's Web services Quality Model (WSQM) [9], which discusses the quality of Web services in use, addresses the business value perceived by the user while using Web Services as one of the important quality factors. WSQM in fact considers three layers of Web Service quality in use, namely *business*, *service* and *system level layer*. Different dimensions of quality are defined for each of these layers. *Business level layer* corresponds to the user's view, and in that context business value quality is detailed through the attributes such as business suitability, business effect and business recognition. *Business value* is seen as dependent on all other quality elements of the model, as well as on the type of business and its characteristics. The service and system level layers address primarily technical aspects of service quality: while the former groups quality attributes related to the measurable *performance quality* of Web Services perceived by the user (*stability*, *efficiency*, and *scalability*), the latter deals with *interoperability*, *security* and *manageability* aspects of Web Service quality.

Amongst all the initiatives situated at more technical level, the Quality Reference Model (QRM) [11] defined by Software Services and Systems Network (<http://www.s-cube-network.eu/>) defines the most comprehensive set of well-defined and relevant quality attributes for service-based applications (SBA). The model aims to cover end-to-end view on service quality (considers attributes relevant for both provider and requestor). QRM defines both domain-independent and domain-specific attributes of quality (e.g. data-related attributes for services that operate/produce data, or quality-of-use context for context-aware adaptive services). Defined quality categories include *performance*, *dependability*, *usability*, *configuration*, *data-related quality*, *network-related quality*, *quality in use*, *security*, and cost. Though comprehensive, this model does not explicitly include service functionality as a quality factor. However, we would expect it to be the case: not only that it is practically always addressed both in software and service quality models, but we believe that without adequate service functionality it can't be stated that minimal QoS has been provided at all.

QRM is also an illustrative of reference quality models incorporating the cost as one of the service quality factors. In our opinion, although cost naturally influences the decision of the consumer, cost itself is not the determinant of the quality as e.g. performance or reliability. The latter two

qualities can be considered as the result of the proper design, deployment and configuration of the service, while cost is determined taking into account many different factors (and not only service quality), as for example the political aspects of the business relationship between contractual parties, which don't characterize the service itself.

Out of quality models addressing the quality of conventional service, and hence taking a business perspective, Servqual [12] [13] [14] is the most prominent one to be discussed. The model tackles customer's perception of the quality of the delivered service and therefore represents the basis for expressing customer's quality needs (business-level quality objectives). Servqual enables measuring the perceived service quality over so-called RATER dimensions [14]:

- *Reliability* – regarding consistency in performance and dependability;
- *Assurance* – as the customer's perceived confidence and trust in the employees delivering the service;
- *Tangibles* – concerning the outward, physical evidence of the service;
- *Empathy* – regarding the individual attention given to the customer;
- *Responsiveness* – as the willingness to help customers and to provide prompt service.

For the application in a specific domain, Servqual has to be adapted in order to really be useful, and several attempts of adapting Servqual for assessing e-services quality may be noted in [6] [15] [17]. More precisely, O'Sullivan suggests making use of Servqual as a starting point for electronic services quality model [6] [7]. The Servqual authors proposed as well an adaptation of their original model to assess e-service quality, E-S-Qual [15]. This model measures e-service (i.e. online shops in the study) quality over the following dimensions:

- *Efficiency* – regarding ease and speed of accessing and using the web site
- *Fulfillment* – as the extent to which the site's promises on order delivery and item availability are fulfilled
- *System availability* – regarding the correct technical functioning of the site
- *Privacy* – relating to the degree to which a site is safe and protects customer information.

Finally, Zarvić and Wieringa [16] rely on Servqual and O'Sullivan's work to define their own reference model, where service quality is discussed from the point of view of business value. The proposed service quality attributes are:

- *Reliability* – involving consistency in performance and dependability,
- *Responsiveness* – concerns the willingness or readiness to provide services,
- *Access* – involves ease of contact,
- *Communication* – means keeping customers informed in language they can understand,
- *Credibility* – involves trustworthiness, believability and honesty,
- *Security*,

- *Understandability* – involves making the efforts to understand client's needs,
- *Availability* – concerns times when and place where service is available,
- *Trust* – deals with trusting the competence and intentions of a service provider.

The authors define the initial mapping of service quality attributes to the quality attributes of the realizing information system (those of ISO/IEC 9126-1:2001). It is stated that hypotheses of the impact of service quality attributes on software quality attributes, based on which the mapping is established have yet to be validated.

#### IV. BUSINESS-LEVEL SERVICE QUALITY MODEL – A PROPOSITION

##### A. Motivation

We have already discussed the need to clearly separate business and technical viewpoints on services, and thus on their non-functional aspects. Business view of non-functional aspects should provide the first version of what level of quality is expected, while the technical view should provide more details on QoS, according to the chosen solution. The attributes of service quality at technical level are thoroughly discussed in many different works and more or less well established (in our work, we intend to rely on S-Cube QRM for identifying important attributes for technical description of service quality). However, defining the model for business-level description of service quality (expressing both requirements and qualifying services) represents the real challenge, and the state-of-the-art work showed that there are very few works addressing this topic.

It is evident that the quality of a business service is influenced by that of technical services enabling business activities. Nevertheless, what characterizes business service quality is the adequacy of the service for users and activities it is aimed to support, e.g. in terms of relying on domain-specific vocabulary, alignment with domain activities, support of the existing regulation and standards relative to the domain etc.

The following paragraphs present our initial proposition of a model for expressing service quality requirements from a business perspective, which is adapted to highly-collaborative construction projects (and within Dest2Co project). The model aims to provide business experts with a way to discuss service quality with terms and concepts they are used to, i.e. level of confidentiality of data exchanged in a project, adequation to collective practices of actors etc.

We first discuss the method used to develop this model, and then proceed with detailed presentation of the current proposition.

##### B. Research method

The approach adopted for developing the model seems more like a bottom-up approach. Although quality aspects covered by the model were chosen from relevant literature review, their sense is expressed in the framework of the given business context (i.e. service design and innovation in construction collaborative projects) and this has been done by involving domain experts. Furthermore, in each stage of

model development, the attention has been put on gathering and formalizing only what appears to be the relevant quality requirement for the service in this concrete domain, rather than covering all possible quality aspects discussed in theory. At last, it may be worthwhile underlining the iterative nature of the approach, although the way it is presented below may suggest to the reader it is a waterfall-like approach.

The approach used to develop the proposed model comprises the following steps:

1. *Objectives definition*: As already discussed, our goal is to define the service quality model enabling the qualification of services from early stages of service development, and thus approaching quality concern from the business perspective. The model should at the first stage be adapted to construction projects, since our experience in developing services in this domain drives the definition of this quality model. In the end, this quality model should enable business experts to express the quality requirements for services in terms and concepts they are used to.
2. *State-of-the-art review*: Relevant quality models have been selected (cf. Related work on service quality modeling) and their applicability in the highly-collaborative context has been analyzed. We specifically focused on Servqual and e-S-Qual, since they enable categorizing user's business-level expectations in a way which seems to be consistent regardless of the services' domain. The WSQM's concept of business level layer quality was also taken into account for it as well corresponds to the user's view on service quality. Our approach was to try to integrate most of the quality aspects discussed in above-mentioned models in the developing proposition, not being too extensive, but rather focusing on the usability of the model.
3. *Selection of high-level quality categories*: Relying on the selected models, a set of relevant quality categories is proposed (cf. Table I), to structure the developing model.
4. *Validation of initial set of categories with domain experts*: These initial quality categories have been validated against a number of already achieved and ongoing projects in the construction sector: e.g. CRTI-weB services developed and now transferred to construction sector, and innovative services regarding mobile computing for construction site (ongoing project). Domain experts involved in these projects validated that service quality aspects discussed and pointed out by business actors could mostly be covered with the initial categories set.
5. *Identification of relevant quality attributes*: As concrete quality characteristics should reflect the specific non-functional aspects of the domain, we aimed to identify them relying on the experience in developing IT-supported services. This was done by reviewing the quality requirements elicited in mentioned projects for the construction sector with involved domain experts. The current proposition of the model is presented in detail later in this section.
6. *Validation*: Two perspectives for validating the model have been identified. Firstly, service design experts

would assess the applicability of the model for qualifying services already developed and used (i.e. case study approach). In the second place, the validation with business actors would consist in assessing the appropriateness of the model for specifying quality requirements in experimental phases of innovative services design. This paper later elaborates on the first validation perspective, that is, it presents the application scenario (cf. Application scenario section) to which the initial model proposition is confronted.

Several difficulties we came upon when identifying quality attributes of the model require further discussion. Firstly, the level of detail with which we were able to define quality attributes (reflecting business concerns) differs from one category to the other. This may be due to the type of experience our experts had, and/or to the difference in relative importance of a certain quality category in different project types. So, our initial proposition may be regarded as “unbalanced”. Nonetheless, we assume that with the following application scenarios it will be possible to deflect this. Secondly, we were faced to a problem when identifying important aspects of performance and security from the business viewpoint, since it appeared that those would be expressed in all those well-known attributes used at qualifying these categories in technical terms.

Table I.  
Quality categories

Quality categories	Quality categories definitions
Business suitability	Business suitability refers to the suitability of the service for conducting the activities in the given business domain (e.g. highly-collaborative context).
Stability (Dependability)	Stability refers the ability of the service to be available, reliable and accurate.
Performance	Performance refers to the efficiency of support the service provides for the business activities.
Security	Security refers to the degree of information protection so that unauthorized persons or systems can't read or modify them and authorized persons or systems are not denied access to them.
Usability	Usability refers to the degree to which the service is easily understood, learned, and used.
Regulatory and interoperability	This category refers to whether the services is able of supporting the existing regulations and to which extent the service is able interoperating with other services that are defined within the same context.

This may be due to these two quality aspects gaining recently so much on importance for the business, that business actors are in general quite familiar with the corresponding technical terminology and very often express the requirements that are and technical in nature. In the context of Dest2Co method, it may as well mean that business experts'

viewpoint (BRV) overlaps with the one of service technical experts (TSV), especially regarding the requirements on performance, security, usability of the service (interface), and that it is difficult to make a clear distinction between these different viewpoints on quality. As there is no doubt of the importance of security and performance for the business, these are kept within the model, despite being defined with a very general set of attributes. Further experimentation would allow us to validate our hypotheses and to complete the model.

Regardless of discussed difficulties and the fact that this proposal has still to mature, we believe the added value of the proposed model is two-fold: First, it allows business actors to define non-functional aspects of business services, and consequently to assess the quality (and other) aspects of service compositions proposed as support of given collective business practices. Such an approach is really important in the design of services at the beginning of a construction project. Furthermore, the model makes possible expressing non-functional requirements for services from high-level standpoint, and may ultimately lead to better understanding and precisely defining requirements for related technical services.

C. Initial model proposition

In this section, we present quality categories and their belonging quality attributes, which constitute an initial service quality model (from the business perspective).

**Business suitability**

*Business domain adequacy* refers to how well the service corresponds to the defined problematic of the domain, in terms of the domain coverage (*applicable area of services*), and *flexibility to major changes* that may occur in the collaborative context.

*Effect on collaborative practices* refers to which extent the collaborative practices (for which the service provides support) are supported and to which extent they would have to be aligned if the service would be used.

*Reputation within the sector* refers to how well the service is perceived by others business actors inside the sector or within relevant communities.

**Stability**

*Reliability* refers to whether the service can function in a predictable way and be trustworthy, i.e. whether service produces correct information and does not impede the performance of users in achieving their goals.

*Availability* refers to whether the service is in a state to perform its required function.

*Accuracy* refers to whether the information provided by the service has the needed degree of precision.

**Performance**

*Time behavior* refers to the response and processing times and throughput rates of the service when performing its function, under stated conditions.

*Resource utilization* refers the amounts and types of resources used when the service performs its function under stated condition

## Security

*Confidentiality* is the degree of protection from unauthorized disclosure of data or information, whether deliberate or accidental.

*Integrity* is the degree to which a system (service) prevents unauthorized access to, or modification of, computer programs or data.

*Non-repudiation* is the degree to which actions or events can be proven to have taken place, so that the events or actions cannot be repudiated later.

*Accountability* is the degree to which the actions of an entity can be traced uniquely to the entity.

*Authenticity* is the degree to which the identity of a subject or resource can be proved to be the one claimed.

## Usability

*Understandability* refers to whether the information on the service is provided so that users can recognize the appropriateness of the service for their needs. This can include demonstrations, tutorials, documentation etc.

*Learnability* is the degree to which the service enables users to easily learn how the service operates, i.e. whether the service is intuitive enough to be learnt easily.

*Ease of use* refers to the degree to which the service has attributes that make it easy to operate and control.

*User error protection* refers to the degree to which the service protects users against making errors.

## Regulatory and interoperability

*Supported standards* addresses the issue of to which extent the service takes into account or can support the existing relevant regulations (defined for and) applied in the domain. It also refers to whether the service is open (or flexible with regards) to supporting other regulations that may be relevant within the domain.

*Interoperability* tackles the question of whether the service is capable of interoperating with the existing services defined/used. This is assessed based on the *interoperability of the organizational roles and responsibilities* (e.g. Does the service rely on the same definition of roles, or if not, is it possible to establish the finite, precise and bidirectional mapping between the set of roles for each service?), and the *interoperability in terms of the information structure* (e.g. Do services rely on the same information structure, or if not, is it possible to establish the finite, precise and bidirectional mapping between the two structures (for each service)?).

## V. APPLICATION SCENARIO

The proposed model has firstly been validated through the application to a scenario observed in a real project, with the aim of assessing its coverage and applicability. In this scenario, the initial quality model is used for qualifying services at the design stage and from the business point of view.

The scenario addresses collaborative practices related to the collaborative thermal assessment of an architectural project, and its services support, designed with the Dest2Co method. We present one of the services and focus on its non-functional aspect description using the proposed quality model.

### A. Collaborative context of the scenario

The situation regarding the assessment of environmental quality of building project is challenging nowadays, since even higher energy efficiency is sought for, and the related certification of the building projects have to be obtained. Depending on the project nature (e.g. wood or concrete-based construction) the assessment can be performed at different times in the project (both in the design and during construction), and also may be done by various experts.

For the given context of the application scenario, the following collaborative practice has been identified: *the energy efficiency assessment of the design project is required by the architect at the end of the early design stage and it is performed by an independent expert. Results of the assessment are transmitted both to the architect and eventually to the building owner.* This collaborative practice involves the project's architect, thermal expert(s) and possibly the building owner.

### B. Service design: functional and nonfunctional description<sup>1</sup>

To support the presented collaborative practice, a business service has been designed: *Thermal assessment of the project based on design document requested by the architect (and performed by external expert).*

The functional description of the service is given as follows (in accordance with concepts of Dest2Co BSV meta-model): The *goal* of the service is to allow the expert's performing of thermal assessment of the design project as well as reporting to the architect. Its *input* is a project design description (both graphical and textual representations) and its *output* is a written thermal assessment of the project. *Pre-conditions* applying to this business service are 1) that the design project is enough advanced (designed) to be assessed, 2) that design documents are complete enough to enable assessment and 3) that criteria for the assessment are known or defined (usually based on standards and regulations applicable to thermal norms). The expected *effect* is that the design project is assessed with respect to thermal aspects, and that in case of negative assessment (low energy efficiency) the redesign/modifications may be suggested.

The non-functional aspects of the designed service are expressed relying on the model proposed in this article. Since the qualification of services is taking place at the design stage, it is not really possible to precisely characterize those quality aspects related to run-time (i.e. performance).

### Business suitability

Business service designed is adapted to AEC domain projects (*business domain adequacy*). It supports a typical collaborative practice that can be found in all project types in the domain, but remains very dependent on the organizational context, because it's fixed that the architect-designer is the one who asks the evaluation of the design project (*flexibility*). As it supports a typical collaborative practice, using this service wouldn't have a side-effect on other practices applied

<sup>1</sup> The transactional aspect of the service, which is the third aspect covered by Dest2Co method, representing the information flow between service activities is detailed in [5].

in the same situation. In the scope of this application scenario, we were not able to qualify the *reputation* of the service.

### Stability

Since current design of the service doesn't completely take into account all possible alternative scenarios, we may say that *reliability* of the service is not excellent: all related business rules are not systematically being specified in present design descriptions.

### Performance

Except for assuming that the performance of the related technical service is not going to be influenced negatively by the business logic it will implement, since not much alternative scenarios are managed by the service, we couldn't provide more details on this aspect.

### Security

The alignment between the rights assigned to participants of this service and the responsibilities assigned to their corresponding organizational roles defined in the domain affects the security. The security is enhanced if this alignment is done adequately. However, it needs to be verified how the responsibilities of organizational roles are assigned, in order to be sure that general security recommendations are respected.

## VI. CONCLUSION

This article presents an initial proposition of a service quality model, aiming to support qualifying services from a business viewpoint. This model is primarily aimed at being used in the design of services for the construction projects, but eventually, could in the future be used for qualifying services in other highly-collaborative projects. The method we relied on when developing this model combines top-down and bottom-up approach, i.e. the proposition is elaborated relying both on relevant literature review and the input from domain experts.

Utility of such a model is justified firstly in the context of (existing) service design and composition for a given collaborative project: business-technical service alignment is enhanced if business actors are able to clearly understand non-functional properties of business services, beyond more usual functional descriptions. Secondly, such a model is highly applicable in the context of innovation through service design, where defining non-functional characteristics of a business service can help in defining requirements for software services.

Indeed, our initial application scenario, presented in the article, showed that the model was useful, as it forced to make quality concerns explicit early at the design stage and from high-level (i.e. business) point of view. This helps not only to consider business and technical concerns of services separately so as to enhance the alignment of services to the business, but it enables business actors to clearly understand the capacities and constraints of designed services, before they are finalized and delivered.

Going further, and in the context of service-based sectorial innovation, we consider that innovative business services can

only be demonstrated to business actors through prototypes of future technical services. Enhancing these prototypes with quality description of future real services to be developed can ensure the understanding of their characteristics very early in innovation projects, and provide clear technical requirements for their development.

When elaborating this proposition, we experienced several difficulties. Namely, we noticed that business and technical viewpoint on service quality may overlap, notably with regards to expressing performance and security requirements. In addition, because the model is in its initial stage and because it's driven by our previous experience, some dimensions of quality are more "developed" than others are. Nevertheless, we believe that by experimenting with the model in real-world projects, we may arrive to a mature and ready-to-use service quality model.

The prospects related to this ongoing work are now to validate and possibly improve the model through the confrontation with business experts, as well as in the framework of other innovation projects (e.g. the experimentation with business users of prototyped mobile services). Applications to other collaborative domains could also be relevant in order to assess the genericity and applicability of the proposed quality concepts. Furthermore, once the model has reached a sufficient maturity, we plan to elaborate on the quantification scales for the adopted quality attributes.

## VII. REFERENCES

- [1] P. Nitithamyong, and M.J. Skibniewski, "Key Success/Failure Factors and their Impacts on System Performance of Web-Based project Management Systems in Construction", *ITcon Electronic Journal of Information Technology in Construction*, 12, 2007, pp. 39-59.
- [2] S. Kubicki, E. Dubois, and E., G.Halin, and A. Guerriero, "Towards a Sustainable Services Innovation in the Construction Sector", *21th Conference on Advanced Information Systems Engineering (CAISE'09)*, 2009, Amsterdam, The Netherlands.
- [3] Cherbakov et al., "Impact of service orientation at the business level", *IBM Systems Journal*, vol 44, N°4, 2005.
- [4] S. Ramel, S.Kubicki, A.Vagner and L.Brave, "Viewpoints Reconciliation in Services Design: A Model-Driven Approach for Highly Collaborative Environments", in *Enterprise, Business-Process and Information Systems Modeling, LNCS*, Springer Berlin Heidelberg, 2010, vol. 50, pp. 62–68.
- [5] D. Zignale, S.Kubicki, S.Ramel and G.Halin, "A model-based method for the design of services in collaborative business environments", *Lecture Notes in Business Information Processing. Proceedings of the 2nd International Conference in Exploring Service Science (IESS)*, Geneva, Switzerland, 2011, vol. 82.
- [6] J.O'Sullivan, D. Edmond, and A. ter Hofstede, "What's in a Service? Towards Accurate Descriptions of Non-Functional Service Properties", in *Distributed and Parallel Databases*, vol. 12, 2002, pp. 117–133.
- [7] J. O'Sullivan, *Towards a precise understanding of service properties*, PhD dissertation, 2006.
- [8] *UML Profile for modeling quality of service and fault tolerance characteristics and mechanisms v1.1*, formal/2008-04-05, <http://www.omg.org/spec/QFTP/1.1/PDF/>
- [9] *Oasis Web Services Quality Model draft v2.0*, [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wsqm](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsqm), September 2005
- [10] *IBM Web Service Level Agreement (WSLA) Language Specification v 1.0*, <http://www.research.ibm.com/wsla/WSLASpecV1-20030128.pdf>
- [11] *S-Cube Quality Reference Model for SBA*, [http://www.s-cubenetwotk.eu/results/deliverables/wp-jra-1.3/Reference\\_Model\\_for\\_SBA.pdf/at\\_download/file](http://www.s-cubenetwotk.eu/results/deliverables/wp-jra-1.3/Reference_Model_for_SBA.pdf/at_download/file), 2009.

- [12] A. Parasuraman, V. Zeithaml, and L.L. Berry, "A Conceptual Model of Service Quality and Its Implications for Future Research", in *Journal of Marketing*, vol. 49, 1985, pp. 41-50.
- [13] A. Parasuraman, V.A. Zeithaml, and L.L. Berry., "SERVQUAL: A multiple-item scale for measuring consumer perceptions of service quality", in *Journal of Retailing* 64 (1), 1988, pp.12—40
- [14] A. Parasuraman, V.A. Zeithaml, and L.L. Berry, "Reassessment of expectations as a comparison standard in measuring service quality: implications for further research", *The Journal of Marketing*, 1994, pp. 111-124
- [15] A. Parasuraman, V.A. Zeithaml and A. Malhotra, "E-S-QUAL: a multiple-item scale for assessing electronic service quality", in *Journal of Service Research*, 7(3), 2005.
- [16] N. Zarvić, R.J. Wieringa, and M. van Daneva, "Towards Information Systems Design for Value Webs", in *Proceedings of Workshops of CAiSE 2007*, Tapir Academic Press, Trondheim, Norway, 2007, pp. 453-460.
- [17] H. Kang and G.Bradley, "Measuring the performance of IT services: An assessment of SERVQUAL", in *International Journal of Accounting Information Systems* 3(3), 2002, pp. 151—164.
- [18] B.Batouche, Y.Naudet, and F.Guinand, "Semantic Web Services Composition Optimized by Multi-Objective Evolutionary Algorithms", in *Proceedings of the 2010 Fifth International Conference on Internet and Web Applications and Services (ICIW'10)*, 2010, pp.180-185.
- [19] K.Kritikos, and D. Plexousakis, "Requirements for QoS-Based Web Service Description and Discovery", *IEEE T. Services Computing* 2(4), 2009, pp. 320-337.

## Towards an Interdisciplinary View on Service Science— The Case of the Financial Services Industry

Michael Fischbach  
University of St. Gallen,  
Müller-Friedberg-Str. 8,  
9000 St. Gallen, Switzerland  
Email: michael.fischbach@unisg.ch

Thomas Puschmann  
University of St. Gallen,  
Müller-Friedberg-Str. 8,  
9000 St. Gallen, Switzerland  
Email: thomas.puschmann@unisg.ch

Rainer Alt  
University of Leipzig  
Grimmaische Str. 12  
04109 Leipzig, Germany  
Email: rainer.alt@uni-leipzig.de

**Abstract**—In the last decade service science has received considerable attention in the research community. Most research regards services either from a business or a technical perspective. This paper argues that existing approaches still lack detailed models for the application of the inter-disciplinary nature of Service Science as well as an application of these concepts in practice. This paper describes a first attempt to apply the characteristics of service-oriented architectures from the information systems discipline to the business domain. It depicts autonomy and modularity, interoperability and interface orientation as major design principles that promise potentials when transferred to the business domain. The proposed inter-disciplinary approach was applied at the case of Zürcher Kantonalbank in Switzerland that realized a company-wide Service Management concept according to the presented design principles.

### I. INTRODUCTION AND RESEARCH QUESTIONS

THE service sector has the largest share of value creation in almost all developed industrial nations [1, 2]. The employment figures draw a similar picture: meanwhile in Germany more than 60% of the staff is employed in the service sector, in the US even more than 70% [3]. Due to the steadily growing importance of services, Service Science emerged as a new discipline that explicitly focuses on the research of services. The main justifications for Service Science are the specific characteristics of services. Contrary to the traditional notion of products, services are intangible, mainly based on information and produced and consumed simultaneously [4, 5]. Additionally, they are predominantly based on the usage of resources, instead of their ownership.

Since its first discussion in literature in 2006 [4], many research activity has been dedicated to this area [6]. The fact that designing, specifying, developing, implementing and managing services significantly differs from traditional product-oriented approaches has led to many theoretical contributions in Service Science [6]. Most consider services either from a business or a technical perspective, which might primarily be explained from the disciplinary focus of the authors. This paper argues that the existing approaches either fall short of addressing the interdisciplinary nature of Service Science or applying it to practice. Although interdisciplinarity is seen as a major instrument of innovation [7] and also a

key idea of Service Science, most approaches lack in detailing how it can be realized. In the following an integrated approach which is applied in practice using case study research is suggested. The financial services industry seems suitable for this as products are immaterial in nature and information technology (IT) has a long tradition in this industry. Three distinct research questions are pursued in this paper:

- *What are the elements of a model that fosters the interdisciplinary transfer of concepts from one domain (computer science / IS) to another (business and management)?*
- *What are the characteristics of the technical view on services and how can these characteristics be transferred to business-oriented services in banking?*
- *How might the findings be implemented in practice and what are the business benefits for using a Service Science approach in practice?*

Chapter 2 outlines the research approach. Chapter 3 introduces Service Science as a super-discipline covering a broad body of service-related academic research, its understanding of services, the research model (research question 1) and commonly accepted technically-oriented characteristics attributed to a service. Based on this, the concepts behind these characteristics are transferred to a business context and their likely impacts are evaluated (research question 2). A case study at the Swiss Zürcher Kantonalbank (ZKB) in chapter 4 describes how these Service Science artefacts are adapted in practice in order to overcome major market challenges (research question 3). Chapter 5 concludes and identifies future research opportunities.

### II. RESEARCH APPROACH

This contribution resulted from the consortium research program “Sourcing in the Financial Industry” – short “CC Sourcing” (see [8-11]). In June 2010 CC Sourcing went into its fourth phase. The research objective of this particular phase of the project is the design of artefacts (architectures, methods, reference models and tools) that help solving problems regarding the customer- and service-oriented design of



financial institutions. The basic principle of consortium research is the collaboration between academic institutions and companies, ensuring both an academic and a practice oriented view the problems. Both parties are engaged in the definition of the problems and objectives as well as in the design, development, evaluation and diffusion of artefacts. The chosen consortium research method is based on a process model for Design Science Research [12] and the corresponding guidelines [13].

The collaboration between practitioners and academics basically can either occur in a bilateral or a multilateral setting. An example for the latter are workshops conducted with all participating partners. Bilateral arrangements include dedicated projects in which the academic institution(s) work(s) on a specific problem of one industry partner. However, despite the bilateral orientation of such projects, the ultimate goal is to extract knowledge and share it within the consortium. Furthermore, existing knowledge gained in the consortium is used for creating an individual company's utility in such projects. So there is a knowledge input from the consortium, a knowledge creation during the bilateral project and a knowledge output to the consortium. This paper partly resulted from a bilateral project with Zürcher Kantonalbank (ZKB) where the CC Sourcing supported the ZKB in introducing a Service Management, and partly from research activities in the whole consortium. Fig. 1 sums up the overall research setup.

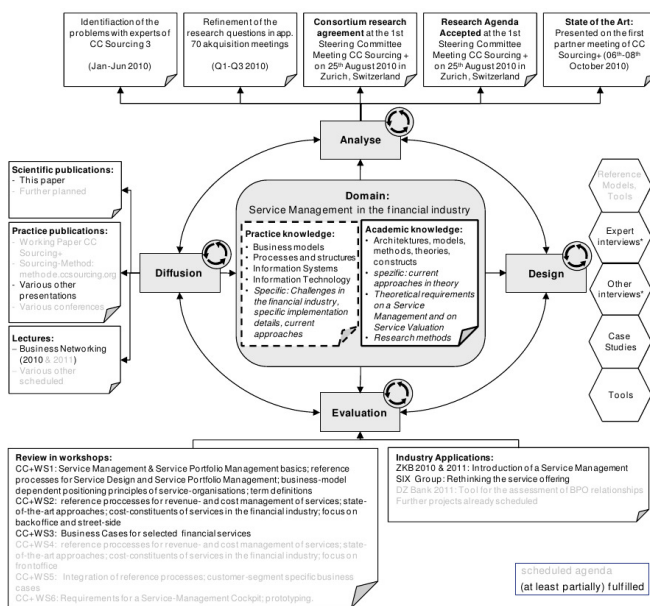


Fig. 1. Consortium Research CC Sourcing+ [10].

### III. SERVICE SCIENCE

#### A. Definitions and Objectives

As Service Science is a relatively new research discipline, it still lacks a commonly accepted definition of the term Service Science [14]. Generally, definitions may be negative (i.e. saying what the term is not about), enumerative (i.e. listing of examples) or constitutive (i.e. naming the characteris-

tics of the term) [15]. The latter is regarded the most suitable one from an academic point of view, as it allows detailed insights into the essences of the term Service Science [16]. Based on this definition, certain overlapping definitional parts can be extracted from literature [4, 17, 18]. Generally agreed upon are the objectives of Service Science, the research objects and the interdisciplinary nature [19].

A commonly accepted objective of Service Science is the development of innovative services by means of suitable methods and formal models. Services need to be developed as systematically as physical goods. Another objective is the continuous improvement of services. A service is the main research object in Service Science and at least some authors emphasize the link to IT. Another research object are service systems, which are dynamic, value creating structures, whose entities are humans, organizations, technologies and/or information [6, 14]. [3] mention call centres or the educational sector as examples of service systems. Finally, to a large extent there is consensus about the interdisciplinary nature of the research field. Service Science bases on models and theories developed in computer science, management science, the engineering sciences and organisational science, besides other disciplines such as psychology, liberal arts, law and sociology.

#### B. Putting an Interdisciplinary View into Practice

As mentioned earlier, despite its interdisciplinary goals, the contributions to Service Science still feature the origin of the researchers. For example, computer scientists regularly pursue a technical view on services, as in the context of web-services and service-oriented architectures (SOA) [20-25]. In contrast, [26] and [27] propose a (relatively wide and un-specific) business-oriented definition: in their view, a service is defined as the application of competency and knowledge in order to create value between supplier and customer. However, Service Science in its original intentions aims at overcoming such unidirectional views by taking a really interdisciplinary view on problems (e.g. [16]). Thereby it qualifies as a valid research object for the IS domain to assume an integrator role between management and computer science [28].

The approach followed in this paper applies characteristics (or design guidelines) from technical services to business-oriented services, thus showing the potential of taking advantage in one discipline from findings of another discipline. Establishing this link, i.e. taking an interdisciplinary view on services, is not a straightforward undertaking. The new definition of a service will not just be a blend of existing definitions, but rather a re-interpretation of them, resulting in an integrated common understanding. It has to fulfil the function of a bridge between business-, IT- and other disciplines [29-33].

The subsequent transfer is grounded on the concept of "conclusion by analogy" (see e.g. [34]): if concepts that proved successful in one domain are applied to another domain that is similar to the first domain, these concepts are likely to be successful in this domain as well. Thus, as technical SOA-services (such as web services) and business services (such as a payments transaction processing service) can

reasonably have an analogical relationship<sup>1</sup>, characteristics that were found to be successful in the SOA domain, as for instance modularity or interface orientation, could possibly also be helpful when applied to business services.

In this sense, Fig. 2 shows the research approach, which does not investigate the business impact of designing IT according to SOA paradigms (as e.g. [35]), but rather investigates the impact of applying the concepts behind technical service characteristics (stemming from a SOA perspective) onto the business context.

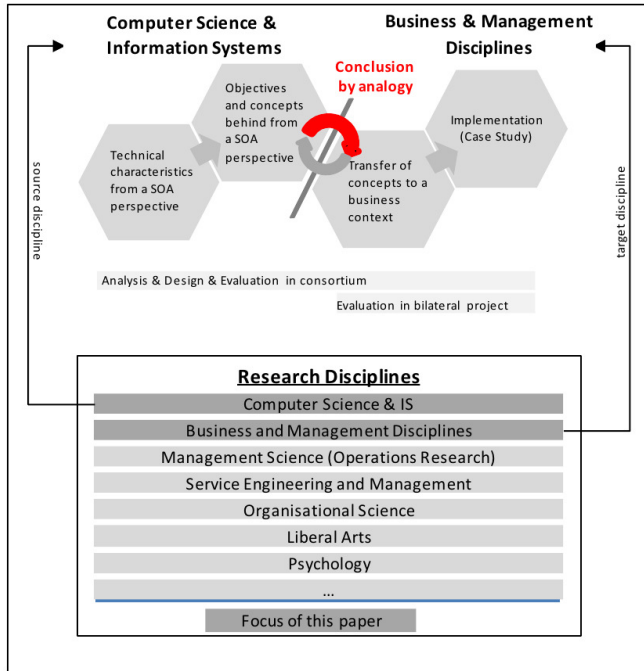


Fig. 2. Conception of the research model

For instance, from a technical point of view, loose coupling means that services are logically independent by “using standardized, dependency-reducing, decoupled message-based methods such as XML” [36], see further [37, 38]. From an interdisciplinary view, the concept would mean, that a bank now has the possibility to dynamically fulfil customer’s needs by arbitrarily putting together various services on a case-to-case basis. Consequently, loose coupling would mean that banking services are designed in a way that allows for a flexible orchestration.

An example for such a flexible, modular service is depicted in Fig. 3. It shows a service bundle that is comprised of various other modules and including different “service levels”, with the modules being interoperable, i.e. they can be combined arbitrarily.

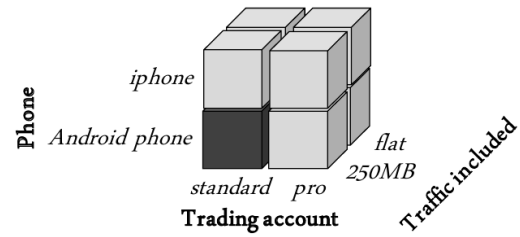


Fig. 3. Exemplary banking service „Mobile trading”

C. Characteristics of Service Orientation

Table I gives an overview over several technically-oriented characteristics of a service.<sup>1</sup> According to a scheme proposed by [39], the identified characteristics are aggregated into three major characteristics. After a short description of the concepts behind, it re-interprets them and derives implications from an inter-disciplinary point of view. With this, it lays the foundation for applying the concepts to a purely business-oriented implementation at ZKB.

TABLE I: IDENTIFIED CONSTITUTIVE CHARACTERISTICS OF A SERVICE FROM A TECHNICAL PERSPECTIVE

Identified characteristic	Exemplary details	Prior re- search
Use of standards (interoperability)	Technical interface standardization, conceptual interface standardization, use of open and common industry standards	[4, 6, 16, 21, 37, 40-44]
Loose coupling (autonomy and modularity)	High service cohesion and weak logical coupling, loosely coupled communications	[6, 21, 37, 40-44]
Platform-independence/abstraction from service-implementation (interface orientation)	Unified service specification, stable and managed service contracts	[21, 37, 42, 43]

Because a transfer of terms and concepts from one domain to another almost always leaves some space for subjectivity (i.e. differing interpretations), the following characteristics are not necessarily inter-subjectively accepted, but should suggest a possible solution towards the realization of an inter-disciplinary perspective. Additionally, the re-interpretation process cannot be proven rigorously as it requires creative design thinking [45]. Therefore, the outlined research approach seems to be the most suitable.

Characteristic 1: Interoperability

Interoperability results from employing standards on different levels. The technical service should offer interfaces that are based on widespread standards in order to ensure interoperability. Examples for such standards are WebService Description Language (WSDL) for metadata specifications, WS-Trust for security specifications, messaging specifications like SOAP and many others. All efforts in this area are more or less purely technically-oriented. According to [46], interoperability efforts have certain objectives: a platform-spanning integration, re-usability (for a detailed discus-

<sup>1</sup> [34] defines an analogy as: “...a common abstraction or explanation between the source and target analogy components and that common abstraction is known to the analogizer”. The mentioned types of services at least exhibit similarity with respect to the name (“service”), the non-transfer of possession, non-storability etc. (see e.g. ([19]) and can thus be said to be in an analogical relationship.

sion of re-usability see [6, 16, 37, 44] and a reduction of redundancies.

These objectives should be pursued in similar forms when looking from a business-oriented view on the banking services offering: by designing banking services in line with the concept of interoperability, the service-portfolio could be much more flexible, as the number of possible combinations rises significantly, without much additional complexity (as the example in section B showed). A standardized, clearly structured and well-described banking service would enhance re-usability as e.g. other business units in the bank could evaluate the functionality of the service and on request include them into their offered customer service portfolio. This enhances an integrative, company-wide view on the service offering and avoids possible parallel developments of the same service in different organizational units (reduction of redundancies).

A means to provide a standardized description of a service's features is a (standardized) service profile description sheet, a human-readable document that contains all decision-relevant information about a service, as e.g. name, service levels, responsibilities and others<sup>1</sup>. An example is a service that provides the processing of payments transactions. The service performs all relevant tasks from the scanning of transfer slips to the client transaction confirmation. The service profile description sheet of this service would include the information displayed in Table II.

By providing (potential) customers such a standardized sheet, they are able to assess the services' characteristics and to compare it to other (eventually similar) services. Further, it ensures a certain degree of standardization and thus diminishes quality uncertainty when re-using the service. Such clear and standardized service documentation also aids in revealing redundancies among services and unambiguously points out which processes are supported and with which services it can potentially be combined. Consequently, applying the principle of interoperability to the business domain seems promising.

TABLE II  
EXEMPLARY CONTENTS OF A BUSINESS-SERVICE PROFILE  
DESCRIPTION

Category	Exemplary Contents
Master Data	e.g. Service-ID, Service-Manager
Service Descriptions	e.g. Utility provided, included sub-services, supplier, service-levels (support times, operating time, volume, availability, max. down-time)
Financials	e.g. accounting unit, amount per acc. unit, fixed costs
Processes & Services	Supported business processes, services with which it can possibly be combined

#### *Characteristic 2: Autonomy and modularity*

Autonomy and modularity means a high cohesion within a single service and concurrent low logical coupling and loose-

ly coupled communication between services. SOA's goal is to group application logic and corresponding data into a set of services [47]. These services are structured in a way that logic and data which are highly dependent on one-another are combined in a service, while keeping dependencies between different services as low as possible (i.e. they exhibit low dependencies on other sub-systems) [48]. Cohesion measures the extent to which the encapsulated functions and data serve the same higher purpose [49]. According to [50], weak logical coupling means that business-requirements that cause a change in one service are not causing a change in the other service as well.

As an example, this concept can be applied to the (business-)service bundle of payments transactions processing, which consists of various process steps including the digitization of payment slips, routing the payments transaction, performing various validity checks such as account balance, black listings etc. Without defining services for each of the process steps, it is almost impossible to consider a (partial) out-sourcing. However, once the service is split into several smaller services, each of which assumes a certain logically enclosed task, parts of the value creation can be outsourced. Thereby, in order to fully reach the objectives that are commonly associated with (out-)sourcing, such as economies of scale, it is important to find a well-considered service cut that enables modularity and autonomy. In the mentioned example, the digitization of the payment forms should be encapsulated in one service. Buying this service from an external supplier lets the firm take advantage of scale economies, as the scanning hardware is rather capital intensive. Further, the digitization comprises a logically encapsulated task. Another benefit of autonomy and modularity in connection with interoperability is the enhanced ability to re-use certain services: eventually, the provider of the digitalization service could also offer it to external customers (i.e. insourcing).

#### *Characteristic 3: Interface orientation*

With interface orientation, services abstract from implementation details and service interfaces provide metadata specifying the outcome to be expected and how the service can be used. However, it does not specify the way in which the service produces the output [51]. By providing a programming language-, platform- and middleware-neutral service description [52], service interfaces also abstract from the technical implementation of the services [41].

A bank client advisors' main task is the sale of banking services such as advisory and bank accounts. Possibly she demands different service levels to be provided by the service-producer (i.e. the bank's backoffice), such as different banking accounts whose reporting functionalities and prices vary with the service levels. End customers can thus be provided with price-adequate solutions that fit their needs, i.e. right-sized solutions. But besides these functional properties of the service (depicted by the already mentioned profile description sheet), the customer advisor does not need detailed knowledge about how the service is produced. This abstraction from implementation details enables both parties, the client advisor and the service producer (i.e. the backoffice in this case) to concentrate on their respective core competen-

cies, while not being confused by domain-specific technical terms and explanations that would be of no use for the other party but rather cause confusion. Applying this principle in a business context poses great challenges on the service provider and thus requires him to closely co-operate with the service-buyer (i.e. the bank's frontoffice): the producer has to clearly and understandable formulate the utility the service brings, without getting lost in details about the way the service is produced. Thus, applying the interface-orientation to business-contexts also helps in overcoming discrepancies between different domains of the company; especially it fosters an end-customer-oriented thinking (in terms of end-customer utility) even from non-frontoffice units.

In the payments transactions example, interface-orientation requires the service description sheet to only state information that is understandable for the service requestor, without getting lost in domain-specific details (such as transaction routing algorithms or the exact correspondent banks the bank cooperates with).

#### Summary

Table III summarises the technically-oriented service characteristics, their application to business-oriented banking services as well as likely effects. In case these effects materialize, undoubtedly at least some of the current challenges in the financial industry could be (partially) solved. The stated re-interpretation examples as well as their likely effects were broadly discussed and validated in several interviews as well as in the consortium (design phase). The accompanying case study shows how the mentioned as well as other concepts were employed by a large Swiss cantonal bank. It further points out first experiences with the approach (evaluation phase).

TABLE III.  
SUMMARY OF THE TRANSFER AND LIKELY EFFECTS

Technically-oriented characteristic	Exemplary application on a business service	Effects
Interoperability	Provision of a standardized service profile description sheet that explicitly points out e.g. which processes are supported.  See Table II.	Fostering standardization, reducing quality-uncertainty, aid in revealing redundancies, fosters re-combination of services
Autonomy and modularity	Breaking down the payments transactions service into several sub-services in order to enable sourcing-decisions, as e.g. to outsource the digitalization of payment slips.	Enable the building of value creation networks; fostering re-usability.
Interface orientation	No exchange of production details between front- and backoffice; only exchange functionalities and other properties by means of the service profile description sheet.	Enable all parties to concentrate on their core-competencies.

## IV. SERVICE SCIENCE IN PRACTICE – THE CASE OF ZÜRCHER KANTONALBANK

### A. Challenges of Zürcher Kantonalbank

Zürcher Kantonalbank (ZKB), headquartered in Zurich, is the largest cantonal bank in Switzerland with about 5000 employees, more than 100 branches and a balance total of 117bn CHF. Customer-proximity and regional anchoring are core elements of the bank's market appearance. The bank concentrates on the region of Zurich (canton), serving retail-, private- and corporate customers. It is internationally active only in the segments Private Banking and Institutionals.

A multi-years project focuses on the introduction of a company-wide Service-Management. Service-Management is a business area-spanning discipline aiming at improved integration of internal service creation activities into the company as a whole and thus to achieve improved bank management. The implementation includes a transformation of the business unit "logistics" from a cost- to a service-centre, which has far reaching organizational implications.

The business unit "logistics" supplies all other business units with various services. Sales teams and specialists departments (referred to as "frontoffice") order services from the logistics, reaching from the execution of payments transactions to the installation and maintenance of workplaces or the real estate valuation. Logistics does not have any end-customer contact. Prior to the introduction of a Service Management the integration of the internal service production activities into the company as a whole was unsatisfactory with respect to transparency, cost allocation and standardization:

- *Missing transparency: Capacities and resources employed and the steps during production were known in detail neither by the frontoffice nor by the logistics itself. Thus, the possibility to calculate exact costs and time expenditures was rather limited. These issues were largely due to intransparent or even missing service descriptions or any other source of information (such as repositories) and no clear service cuts. Redundancies in the service offering were the consequence.*
- *Undifferentiated cost allocation: missing transparency of the (internal) production activities inevitably had an impact on internal cost allocation: if the amount of input (manpower, material, used infrastructure etc.) is unknown, costs cannot be allocated adequately. Thus, the logistics unit charged the frontoffice lump-sums that could not be verified on a calculatory basis.*
- *Non-uniform Service Level Agreements (SLA): SLAs are the core elements of all agreements between the logistics unit and its purchasers. Despite the vast number of different existing SLA, there were no standards regarding their contents. SLA varied depending on the underlying individual agreements and thus were not comparable. For instance, real estate valuation services differed hugely from case to case. Due to their heterogeneous nature the different SLA were not comparable at all. Additionally, there was a mass of heterogeneous SLAs, several people responsible*



*for the same type of service and no structured communication channels between the logistics and other units.*

### *B. Introduction of Service Management*

ZKB's Service Management approach follows the three technically-oriented service characteristics, namely interoperability, modularity & autonomy and interface orientation and their application to the business domain:

#### *Interoperability of services*

All service managers are obliged to provide a service profile description sheet that adheres to well-defined standards and consist of modular constituents. Supported processes as well as "compatible" services have to be mentioned. Sub-services are stated as well. The service description sheet is part of a repository containing all services and service bundles offered to the frontoffice. Service Managers are advised to capture as much individuality as possible while keeping the number of different service levels at a maximum of 3. Up to now, frontoffice staff reports higher transparency regarding the service offering, i.e. by means of the repository they can check whether a planned service already exists in other areas of the bank and exactly which properties and features it exhibits. Further, combination of different services became easier, due to information transparency as well as the interoperable design of services. Consequently, the frontoffice is enabled to increasingly offer individualized solutions consisting of several services put together on a case-to-case basis, thus increasingly aligning the offering to customer needs while still ensuring a certain level of standardization.

#### *Autonomy & modularity of services*

In addition to the repository that contains all service profile description sheets, a service catalogue provides information in much finer granularity, down to single activities of sub-services. All services are cut according to the principles of autonomy & modularity, namely high cohesion within and weak coupling among services. By cutting services according to business logic (as for instance defining a new sub-service for the digitization of payment slips within the payments transaction processing service) enables ZKB to consider partial sourcing (in-/and out-sourcing) of certain services, which lays the foundation for ZKB to concentrate on core competencies and lay out non-core activities into the partner network. Currently, ZKB is considering outsourcing the digitization and certain plausibility checks of domestic as well as international payment slips to specialized providers in order to take advantage of economies of scale. Another effect of the stringent definition of cohesive and loosely coupled business services is that the logistics itself gains a transparent view on its' offering, thus creating potential for improved cost allocation and higher production efficiency.

#### *Interface orientation of services*

Due to rigorously applying the concept of interface orientation, the service profile description sheet (which is the central information source for potential demanders) only contains information relevant for a "buying" decision (see Table II). Production details are undisclosed. Thus, potential and

actual demanders only get the information relevant for their decision, without being confronted with specific realization details. This enables them to concentrate on their core competencies and leaves all implementation detail to the supplier, i.e. the logistics department. However, although the service offering becomes more transparent (i.e. less complex), now it is no longer possible for the frontoffice to make a judgement about the efficiency with which the backoffice is producing. This problem is solved by giving the frontoffice the right to possibly buy services from outside the company. So the backoffice is put under competitive pressure, which also fosters the development towards the creation of networks.

## V. CONCLUSION

Although the interdisciplinary nature of Service Science is accepted in current research, only few provide more detailed guidelines on how to realize it in practice. This paper drafted an integrated approach that was applied at the case of ZKB. For this purpose, three service characteristics from the IS perspective were identified and applied to the business domain, i.e. banking services. The findings are promising in three areas:

- *Innovation and business benefits through an interdisciplinary approach: The presented interdisciplinary approach to transfer artefacts between two domains. SOA concepts from the IS discipline may be used to design business services and thus enable innovation for the design of new services (e.g. mobile trading services). The case of ZKB showed that current business challenges, such as missing transparency, undifferentiated cost allocation and non-uniform SLAs, can be addressed with these concepts.*
- *Improved business – IT Alignment: The linkage of business and IT remains a major challenge in many businesses in general and especially in service-oriented business as the main object is information. The presented approach supports the linkage between the technical architecture approaches of computer science and IS to the business domain.*
- *Practical relevance of Service Science: Much research in the area of Service Science has been provided for theories and concepts. However, there is still a need for practical applications of Service Science. This includes business benefits and experiences of companies that re-organized towards service-orientation. The case of ZKB provided insights although the company is still in a state of transformation.*

Despite these results, several other opportunities for future research arise. Sustainable practice success of the outlined approach will depend on many additional factors, such as cultural transformation, acceptance, communication to employees etc. Implementing the findings requires to operationalise them in clear guidelines and well-established principles. The presented approach is only an initial step towards a more comprehensive concept for the inter-disciplinary transfer of artefacts between scientific disciplines. To live up to the idea of Service Science the characteristics need to be ver-

ified and enhanced with more disciplines, especially the domain of service engineering, and the benefits of using these design guidelines for practice need to be thoroughly evaluated.

## REFERENCES

- [1]. Spohrer, J. and P. P. Maglio, *Fundamentals of Service Science*. Journal of the Academy of Marketing Science, 2008. **36**(1): p. 18-20.
- [2]. Spohrer, J. and P. P. Maglio, *The Emergence of Service Science: Toward Systematic Service Innovations to Accelerate Co-creation of Value*. Production and Operations Management, 2009. **17**(3): p. 238-246.
- [3]. Maglio, P. P., et al., *Service Systems, Service Scientists, SSME, and Innovation*. Commun. ACM, 2006. **49**(7): p. 81-85.
- [4]. Chesbrough, H. and J. Spohrer, *A Research Manifesto for Service Science*. Communications of the ACM, 2006. **49**(7): p. 35-40.
- [5]. Sasser, E., R. P. Olsen, and D. D. Wyckoff, *Management of Service Operations*. 1978, Boston: Allyn and Bacon.
- [6]. Bardhan, I. R., et al., *An Interdisciplinary Perspective on IT Services Management and Service Science*. Journal of Management Information Systems, 2010. **26**(4): p. 13-64.
- [7]. Klein, J. T., *A Conceptual Vocabulary of Interdisciplinary Science*, in *Practicing Interdisciplinarity*, P. Weingart and N. Stehr, Editors. 2000: Toronto.
- [8]. Back, A., G. v. Krogh, and E. Enkel, *The CC Model as Organizational Design Striving to Combine Relevance and Rigor*. Syst Pract Act Res, 2007. **20**:91: p. 91-103.
- [9]. Österle, H. and B. Otto, *Consortium Research: A Method for Relevant IS Research*. Business & Information Systems Engineering, 2010(5): p. 1-24.
- [10]. Österle, H. and B. Otto. *A method for consortial research*, in: *Proceedings of the 18th European Conference on Information Systems (ECIS 2010)*. in *ECIS 2010*. 2010. Pretoria.
- [11]. Österle, H. and B. Otto. *Relevance through Consortium Research? Findings From an Expert Interview Study*, in: *Proceedings of the 5th International Conference on Design Science Research in Information Systems and Technology (DESRIST 2010)*. in *DESRIST 2010*. 2010. St. Gallen.
- [12]. Peffers, K., et al., *A Design Science Research Methodology for Information Systems Research*. Journal of Management Information Systems, 2008. **24**(3): p. 45-77.
- [13]. Hevner, A. R., et al., *Design Science in Information Systems Research*. MIS Quarterly, 2004. **28**(1): p. 75-105.
- [14]. Spohrer, J., et al. *The Service System is the Basic Abstraction of Service Science*. in *Proceedings of the 41st Annual Hawaii International Conference on System Sciences*. 2009. Waikoloa (HI).
- [15]. Bullinger, H.-J., Scheer, A.-W., *Service Engineering - Entwicklung und Gestaltung innovativer Dienstleistungen*. 2 ed. 2005, Berlin, Heidelberg: Springer.
- [16]. Alt, R., *Innovation durch Services Science*. Managementkompass: Industrialisierungsmanagement, 2009: p. 9-11.
- [17]. Abe, T., *What is Service Science*. 2005, Fujitsu Research Institute.
- [18]. Maglio, P. P., et al., *Service Systems, Service Scientists, SSME, and Innovation*. Communications of ACM, 2006. **49**(7): p. 81-85.
- [19]. Buhl, H. U., et al., *Service Science*. Wirtschaftsinformatik, 2008. **50**(1): p. 60-65.
- [20]. Bieberstein, N., et al., *Impact of Service-oriented Architecture on Enterprise Systems, Organizational Structures, and Individuals*. IBM Systems Journal, 2005. **44**(4): p. 691-708.
- [21]. Papazoglou, M. P., *Service-Oriented Computing: Concepts, Characteristics and Directions*. WISE, 2003: p. 3-12.
- [22]. Piccinelli, G. and L. Mokrushin, *Dynamic Service Aggregation in Dynamic Marketplaces*. 2001, Hewlett Packard Laboratories: Bristol (England).
- [23]. Sahai, A., V. Machiraju, and K. Wurster, *Managing Next Generation E-services*. 2000, Hewlett-Packard Laboratories: Palo Alto (CA, USA).
- [24]. Singh, M. and M. Huhns, *Service-Oriented Computing: Semantics, Processes, Agents*. 2004, New York: John Wiley.
- [25]. W3C, *Position Papers for the World Wide Web Consortium (W3C) Workshop on Web Services*. 2001, Hewlett Packard Laboratories: Palo Alto (CA, USA).
- [26]. Spohrer, J., et al., *Steps Toward a Science of Service Systems*. IEEE Computer, 2008. **40**(1): p. 71-78.
- [27]. Vargo, S. L. and R. F. Lusch, *Evolving to a New Dominant Logic for Marketing*. Journal of Marketing, 2004. **68**: p. 1-17.
- [28]. Leyking, K., F. Dreifus, and P. Loos, *Serviceorientierte Architekturen*. Wirtschaftsinformatik, 2007. **49**(5): p. 394-401.
- [29]. Baida, Z., J. Gordijn, and B. Omelayenko. *A Shared Service Terminology for Online Service Provisioning*. in *6. International Conference on Electronic Commerce*. 2004. Delft: ACM Press, New York (NY).
- [30]. Laartz, J., *SOA revolutioniert das Management*. Wirtschaftsinformatik, 2008. **50**(1): p. 72-73.
- [31]. Rust, R. T. and P. K. Kannan, *E-Service: A New Paradigm For Business in the Electronic Environment*. Communications of the ACM, 2003. **46**(6): p. 37-42.
- [32]. Stafford, T. F., *E-Services*. Communications of the ACM, 2003. **46**(6): p. 27-28.
- [33]. Weigand, H., et al. *Value-based Service Design Based On A General Service Architecture*. in *3. International Workshop on BUSINESS/IT Alignment and Interoperability*. 2008. Montpellier: Sun SITE Central Europe, Aachen.
- [34]. DeJong, G., *The role of explanation in analogy; or, the curse of an alluring name*, in *Similarity and Analogical Reasoning* S. Vosniadou and A. Ortony, Editors. 1989, Cambridge University Press: New York. p. 346-365.
- [35]. Mueller, B., G. Viering, and F. Ahlemann, *Toward Understanding the sources of economic potential of service-oriented architecture: findings from the automotive and banking industry*, in *ECIS 2007*. 2007: St. Gallen.
- [36]. Brown, A., S. Johnston, and K. Kelly, *Using Serviceoriented Architecture and Component-based Development to Build Web Service Applications*. 2002, Rational Software Corporation.
- [37]. Baskerville, R., M. Cavallari, and F. Virili. *Extensible Architectures: The Strategic Value of Service-Oriented Architecture in Banking*. in *13th European Conference on Information Systems*. 2005. Regensburg.
- [38]. Erl, T., *Service-Oriented Architecture (SOA): Concepts, Technology, and Design*. 2005: Prentice Hall.
- [39]. Legner, C. and R. Heutschi. *SOA adoption in practice - findings from early SOA implementations*. in *ECIS 2007*. 2007. St. Gallen.
- [40]. Bettag, U., *Web-Services*. Informatik Spektrum, 2001. **24**(5): p. 302-304.
- [41]. Legner, C. and T. Vogel. *Design Principles for B2B Services - An Evaluation of Two Alternative Service Designs*. in *IEEE International Conference on Service Computing (SCC 2007)*. 2007. Salt Lake City (USA).
- [42]. Papazoglou, M. P., et al., *Service-Oriented Computing: A Research Roadmap*. Service Oriented Computing (SOC), 2006(05462).
- [43]. W3C. *Web Services Glossary*. 2004 [cited; Available from: <http://www.w3.org/TR/ws-gloss/>].
- [44]. Winkler, V., *Identifikation und Gestaltung von Services: Vorgehen und Beispielfolle Anwendung im Finanzdienstleistungsbereich*. Wirtschaftsinformatik, 2007. **49**(4): p. 257-266.
- [45]. Hey, J., et al., *Analogies and Metaphors in Creative Design*. International Journal of Engineering Education, 2008. **24**(2): p. 283-294.
- [46]. Heutschi, R., *Serviceorientierte Architektur: Architekturmodell und Umsetzung in der Praxis*. PhD Thesis. 2007: University of St. Gallen.
- [47]. Klesse, M., Wortmann, F., Schelp, J., *Erfolgsfaktoren der Applikationsintegration*. Wirtschaftsinformatik, 2005. **47**(4): p. 259-267.
- [48]. Simon, H. A., *The architecture of complexity*, in *Managing in the modular age*, R. Garud, Kumaraswamy, A., Langlois, R. R., Editor. 2002, Blackwell Publishers: Malden. p. 15-38.
- [49]. Papazoglou, M. P., Yang, J., *Design Methodology for Web Services and Business Processes*, in *Technologies for e-services: third international workshop, TES 2002*, A.C. Buchmann, Fiege, L., Hsu, M.-C., Shan, M.-C., Editor. 2002, Springer: Berlin. p. 54-64.
- [50]. Gall, H., Hajek, K., Jazayeri, M. *Detection of logical coupling based on product release history*. in *International Conference on Software Maintenance (ICSM)*. 1998.
- [51]. Balzert, H., *Lehrbuch Grundlagen der Informatik - Konzepte und Notationen in UML, Java und C++*. 1999, Heidelberg, Berlin: Spektrum Akademischer Verlag.
- [52]. Dodd, J., *Practical Service Specification and Design Part 3: Specifying Services*. CBDi Journal, 2005(Juli/August): p. 11-20.





# Services Composition Model for Home-Automation peer-to-peer Pervasive Computing

Juan A. Holgado-Terriza, Sandra Rodríguez-Valenzuela  
Software Engineering Department  
University of Granada, Granada 18071, Spain  
Email: jholgado@ugr.es, sandra@ugr.es

**Abstract**—Collaborative mechanisms between services are a crucial aspect in the recent development of pervasive computing systems based on the paradigm of service-oriented architecture. Currently, trends in development of services computing are taking into account new high-level interaction models founded on services composition. These services make up their functionalities with the objective of creating smart spaces in which services with different purposes can collaborate to offer new and more complex functionalities to the user transparently. This leads to the creation of collaborative spaces with value-added services derived from the composition of existing ones. However, there are many aspects to consider during the development of this type of systems in pervasive spaces, in which the extensive use of embedded devices with limited characteristics of mobility, computing resources and memory, is a large handicap. This paper describes a model of services composition based on a directed acyclic graph used in a services middleware for home-automation, in which we work with loosely coupled services-oriented systems over the peer-to-peer technology JXTA. The presented composition model guarantee the acyclicity of the composition map between services as well as favours the building of collaborative light services using peers as proactive entities, which could be executed on embedded devices. These ones are capable of establishing dynamic intercommunications, synchronizing with others and form coalitions to cooperate between theirs for a common purpose.

## I. INTRODUCTION

**T**HE PROLIFERATION of smart communication devices and the extensive use of the Internet in any device (e.g. mobile device) have brought the need of integrating business process models into any kind of system [1]. Business processes provide complex and sophisticated services and products as consequence of the massive penetration of data from internet-enabled devices, the data management capabilities of mobile devices and any other wearable and embedded devices, the context information depending on its location, its physical and computing environment, and even its human users. Services such as videoconferencing, VoIP, ambient assisted living appliance, distance education or learning, smart security home are now possible. Moreover the integration of more resource-full computing devices into the home environment may assist us by means of autonomous decision making based on the context or available data which is part of the Smart Home concept [2]. The convergence between the operation given by in-house devices and the business process, require computing models where the interactions between the device operation and the business process should be more natural. The technological advances necessary to build a pervasive computing

environment fall into four broad areas: devices, networking, middleware and applications [3].

Many years ago, Weiser defined the main lines of pervasive computing research that are focused initially on hardware issues such as the reduction in size and power consumption, the processing power, the wireless communication protocols and the invisibility, transforming the devices in invisible objects [4]. Other key features of a ubiquitous environment are the dynamic reconfigurability, modularity, extensibility and portability. Thus, the middleware platform for pervasive systems must solve problems such as interoperability, heterogeneity and transparency with respect to devices, as well as dynamic discovery, selection, composition and adaptation of components [5]. Today, the principles of the SOA (Service Oriented Architecture) paradigm are the most widespread and used for the development of pervasive computing systems [6].

However, the development of services-based software architecture may become very complex in ubiquitous computing when the interaction space includes embedded devices of small size. The resources of these devices are often too limited to run certain processes and store information. Besides, they possess limited capabilities in terms of processing power, memory, time battery life and bandwidth [7]. These characteristics mean that the application development and business processes in ubiquitous environments requires new specific computer models, and therefore, new software infrastructures that support these applications, taking into account that they must be integrated into embedded execution environments, i.e., devices with limited resources.

By means of services composition a service can make up its functionality in base of the collaboration with the rest of services. Hence, a collection of collaborative services can provide a way of creating smart spaces that offer new and more complex functionality transparently to the user [8]. The most common service collaboration models are based on the orchestration and choreography of services. In the service orchestration the execution flow control is always responsibility of one of the parties involved in the collaboration, while in the service choreography any of the entities involved in the collaboration can take part in the interaction [9].

There are several service composition approaches which study how to make a composition from existing services in order to achieve a more complex functionality that typically is not provided by a single available service [10]. This paper

presents a model of services composition based in a directed acyclic graph and used in a services middleware for home-automation. The middleware is based on the SOA paradigm and it has been built over the peer-to-peer framework JXTA, which has been selected by the good scalability and the decentralized nature achieved by the P2P systems [11]. The service composition model is based on the orchestration principles in which the interaction control is responsibility of each service, as well as the execution order of operations and flow of messages and transactions required in the collaboration. The composition model in the service middleware often differentiates between simple and composite operations. The execution of a composite operation involves the execution of a set of requested operations on several collaborative services. The presented model of services composition favours building of collaborative decentralized lightweight services, which could be executed on embedded devices, using peers as distributed and dynamic entities.

The rest of the paper is organized as follows. Section 2 introduces the most representative aspects of the SOA-based middleware which has been developed taking into account the constraints imposed by the use of devices with limited resources. Section 3 specifies the details of the service composition model designed over the platform. Section 4 introduces the communication patterns involved in the service composition model. Section 5 explains the behaviour of a service when a composite operation is executed. In section 6 we will show an example which involves several services with composite operations. Finally, section 7 reviews the related work and shows the future research, before concluding in section 8.

## II. SOA-BASED MIDDLEWARE FOR PERVASIVE COMPUTING

DOHA (Dynamic Open Home-Automation) is a services platform for the access, control and management of home-automation systems that facilitates the construction of dynamic, scalable and pervasive applications, based on a set of lightweight and independent services. The DOHA services platform is based on the SOA paradigm and uses the peer-to-peer middleware JXTA as the support platform.

DOHA abstracts the physical distribution of devices and its management by a set of high-level collaborative services, as shown in Fig. 1. The platform promotes the collaboration between services which involve communication between peers at a lower level, and the interconnections between devices across different networks placed on diverse subnets at the lowest level.

DOHA is supported by the peer-to-peer framework JXTA, which allows any device connected to the network to collaborate and communicate as a peer, providing positive features such as interoperability, platform independence and ubiquity. This enables integration in the same network of nodes that can represent services, physical devices, applications requiring the use of services, etc., leading to a system easily and naturally scalable, where new features and new devices and services

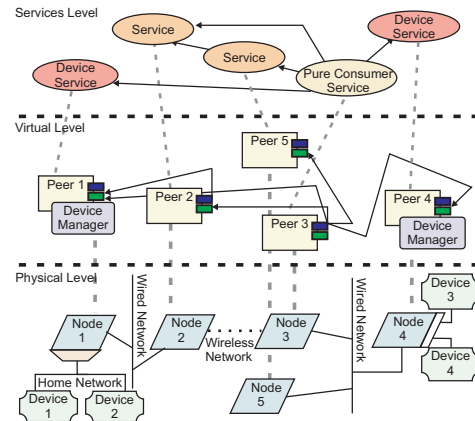


Fig. 1. Levels of abstraction of DOHA platform.

can be added in a more flexible manner. There are several approaches which implement a distributed collaborative model based on P2P technologies. For example, the JXTA-Overlay is a JXTA-based P2P platform designed with the aim of leveraging the capabilities of Java, JXTA and P2P technologies to support distributed and collaborative systems [12]. As in the Barolli et al. model, the DOHA platform has been successfully applied to large-scale systems with powerful embedded devices [13]. However, when the memory resources of embedded devices are scarce, there is no space for the full JXTA middleware. In this case a variation of JXTA for J2ME (CLDC-MIDP2) is deployed on embedded devices [14].

The DOHA platform takes into account another important aspect in pervasive systems, the large number of heterogeneous hardware devices that may be part of a network, and how the hardware interaction is managed from the service level (e.g. HVAC, temperature sensor, alarm clock). The JavaES (Java Embedded System) framework is used to enable the operation with different types of physical devices (e.g. appliances, sensors, actuators) in the environment. The access to the hardware is carried out in a standardized fashion, since JavaES abstracts the specific hardware capabilities of each embedded device [15].

A service in the context of DOHA is an autonomous self-contained component capable of performing specific activities or functions independently, that accepts one or more requests and returns one or more responses through a well-defined, standard interface. There are two special types of services in DOHA: the *Device Service* and the *Pure Consumer Service*. The *Device Service* can interact with the physical devices of the environment and it provides physical device control. The *Pure Consumer Service* does not provide access to other services; it often provides access to end users applications, usually with a graphical user interface, and it does not offer functionality to the rest of services.

A DOHA service has an internal structure organized in a set of software layers. The multilayer structure facilitates the decoupling of tasks performed by a service in cohesive components; e.g., separating the task of requesting services

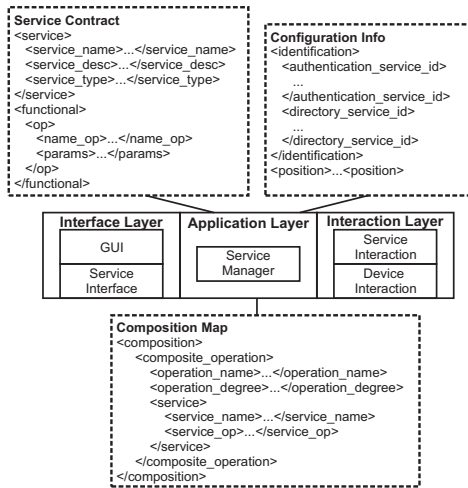


Fig. 2. Anatomy of a DOHA service.

from the task of providing services. Fig. 2 shows the anatomy of a DOHA service which provides a model to design and implement services and also allows managing the behaviour flow of the service in each step of its execution.

The *Interface Layer* is the public access point of the service which provides the functionality of the service in terms of operation (simple or composite) that can be invoked from any other consumer service. The layer is responsible of receiving the requests, executing the service operations by forwarding them to the *Application Layer* and finally returning a response. Services may have a graphical interface to allow the user’s access. In this case, the *Interface Layer* should include the *GUI component*.

The *Application Layer* is the real core of the service and it is in charge of processing the operations from the received requests given by the *Interface Layer*, supervised by the *Service Manager* component. The *Service Manager* handles the execution of an operation in the context of the service, including the necessary invocation of operations of any collaborative service, and provides an adequate response to the service requesters. The decoupling of the *Application Layer* with respect to the *Interface Layer* allows us a way to control the "stateless" feature of the service, since the use of state information may adversely affect its availability and scalability.

All DOHA services are stateless from an external point of view, since the *Interface Layer* always provides a response to any request carried out by any service at any given time. But in many occasions the service could be required to maintain state information; for example, when the service virtualizes the state of a physical device (e.g. temperature sensor, illumination sensor) with a logical state enclosed in the service. The state-dependent part is bounded and limited by the *Application Layer* and it is addressed by the *Service Manager*. Therefore, it is imperceptible to the outside.

Finally the *Interaction Layer* contains the logic necessary to make possible the communication, the invocation of operations

to other services or directly to devices (e.g. sensors or actuators), and the recovery of responses which are delivered later to the *Application Layer* in order to complete the execution of the running service operation. The *Service Interaction component* is responsible of managing the collaboration with other services, acting as a client of consumer of these services. On the other hand the *Device Interaction component*, which only appears in the *Device Services*, interacts with JavaES which gives a hardware abstraction to access the physical devices in the environment.

The deployment, start-up and execution of the service require knowing additional information that is managed by the service during its life-cycle at runtime. This information is enclosed into three descriptive documents which form the base of the service specification. These ones are the *Service Contract*, the *Service Composition Map* and the *Service Configuration*. Each of them abstracts a fundamental aspect of the service within the platform and contextualizes his collaborative behaviour with other services. The *Service Contract* is a public resource exchangeable between services, containing a description of the requirements, restrictions and functionality of a specific service. This information will be exploited by the rest of services in order to be aware of the functionality offered by the service and later make use of it. The *Service Composition Map* establishes the relations among services and the operations they are to perform, in order to carry out composite operations. This information is private and only accessible by the service itself, which is the only one who knows with which other services it is supposed to interact, in order to carry out a composite operation. Finally, the *Service Configuration* is needed to initiate the execution of the service, and it contains configuration parameters related to the software infrastructure that provide support to the service such as JXTA and JavaES. Related to JXTA, the *Service Configuration* encloses the identifier of the *Authentication Service* (peer group id) and the identifier of the *Directory Service* (rendezvous peer).

### III. SERVICE COMPOSITION MODEL

The service composition model of the DOHA platform is based on the activities that a service should perform when it needs to collaborate with other services to complete a requested operation. Dynamic modelling of services, such as the execution flow of a service, can be shown from the point of view of the type of the operation to be carried out and what activities the service must perform for it. The types of operations that services can perform are simple or composite. A simple operation is a single transaction that the service can perform by itself, i.e., the service has all the resources necessary to carry it out and it does not require interaction with other services. In contrast, a composite operation involves the invocation of one or more operations in one or more services. The service that owns the composite operation is responsible of its execution and it must interact with the services involved in the operation, the *requested services*, to get the necessary functionality in order to complete the whole

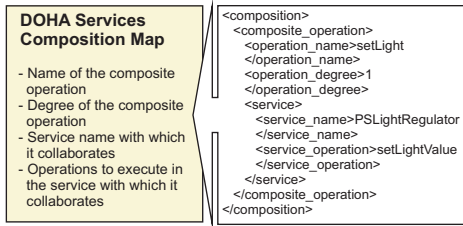


Fig. 3. Structure of the *Service Composition Map* and an example represented in XML.

operation. A service that only has simple operations is called a basic service; whereas a service that implements composite operations invoking other services is called a composite service. A *Device Service* is an example of a basic service that is composed only of simple operations. The composite operations are the base of the collaboration model of the platform and are listed by the *Service Composition Map*.

In the XML code of Fig. 3 we can observe the structure of a DOHA *Service Composition Map* that lists the requested services (using meta-data enclosed into `<service_name>` tags) and the corresponding invoked operations required for the execution of a composite operation available on the service. In this case, the *Service Composition Map* belongs to the *PSLight* service, which make use of the *PSLightRegulator* service to manage the composite operation *setLight()*.

Once a service is running, it can interact with other requested services of the platform to perform composite operations. The service is only aware of these requested services, allowing the collaboration with them to be carried out without user intervention, creating autonomously collaborative applications at service level. However, the service is not aware of the rest of the services available in the platform.

Service composition can be modelled using graph theory. The composition map of a service is formed by composite operations. Each composite operation  $op$  of a service  $S$  can be defined by a directed graph  $G_{op_S} = (o_{op_S}, V(G), L(G), E(G))$  where:

- $o_{op_S}$  is the main vertex of the graph, and corresponds to the origin service  $S$  of the composite operation  $op_S$ .
- $V(G)$  is the set of vertices of the graph where each vertex represents a service invoked from the operation  $op_S$ , i.e. a required service.
- $L(G)$  is the set of labels where each label embodies an invoked operation in a required service.
- $E(G)$  is the set of edges related with two vertices where the origin vertex is  $o_{op_S}$ , the destination vertex is an element of  $V(G)$  and the label of the line is an element of  $L(G)$ . Accordingly, each element of this set is defined by the function  $edge_i(o_{op_S}, op_i, v_i)$ , where  $op_i$  must be an invoked operation of a required service  $v_i$ , verifying that  $E(G) \subseteq o \times L(G) \times V(G)$ .

The composite operation graph is directed because the arcs or operations between vertices always have a sender and a receiver. The former is the service that starts or invokes a

composite operation and which performs the request, and the latter is the requested service which is the owner of the invoked operation. Thus, we can define a syntax based on graph theory to model the composition between services. By means of these premises, the full composition map of the service  $map(S) = G_{op_{1_S}} \cup G_{op_{2_S}} \cup G_{op_{3_S}} \cup \dots \cup G_{op_{n_S}} = \cup_i G_{op_{i_S}}$  is a directed super-graph formed by the union of all the composite operations of the service  $S$ .

The composition graph of a given service may contain calls to several services operations in the same composite operation, but these operations are performed in a sequential fashion, not nested. The depth of the composition graph of a service is always of one level, however, we must take into account that when a service starts a composite operation, it does not know if one of the operations in the requested services is composite too. In this case, we could have a chain of composite operations. Because of this, we define the function  $degree(G_{op_S})$  as the complexity degree of an operation which establishes the depth of its graph. A simple operation has complexity degree 0, i.e.  $degree(op) = 0$ . The complexity degree of a composite operation  $G_{op_S}$  is defined by  $degree(G_{op_S}) = \max(degree(op_i)) + 1, \forall op_i \in L(G)$ . We can say that the degree of a service is the maximum degree of all the composite operations of its composition map, i.e.  $degree(map(S)) = \max(degree(G_{op_{i_S}})), \forall G_{op_{i_S}} \subseteq map(S)$ .

The function  $degree()$  is used to ensure the acyclicity of the service composition graph. We have established the following restrictions on the construction of composite operations:

- All composite operation must finish with a simple operation.
- A composite operation can only invoke another composite operation with lower complexity degree. A composite operation with degree 1 can invoke only simple operations, i.e. operations with degree 0. A composite operation with degree  $n$  can invoke another composite operations with maximum degree  $n - 1$ .

We can see an example of composite operations degree in Fig. 4. The degree of the simple operation, *setTemperatureValue()* in the service *Temperature Regulator Service*, is equal to 0. This operation is invoked by the composite operation *setTemperature()* of the service *Temperature Service*, therefore this operation has degree 1. Also, the operation *setTemperature()* is invoked by the composite operation *setProfile()* of the service *Comfort Service*. The operation *setProfile()* also invoke the simple operation *tvOn()* of the service *TV Control Service*. Hence, we can say that the degree of the composite operation *setProfile()* of the service *Comfort Service* is 2, because that is the maximum degree of its invoked operations plus 1.

Based on the restriction imposed on the invocations between composite operations for the function  $degree(op)$  we have defined the axiom  $edge(o_{op_{v1}}, op_{v2}, v2) \rightarrow degree(op_{v1}) > degree(op_{v2})$ . This axiom imposes that the degree of a composite operation  $op_{v2}$  of  $v2$  must be strictly less than the degree of the composite operation in  $o$  which invokes it,  $op_{v1}$ . Using



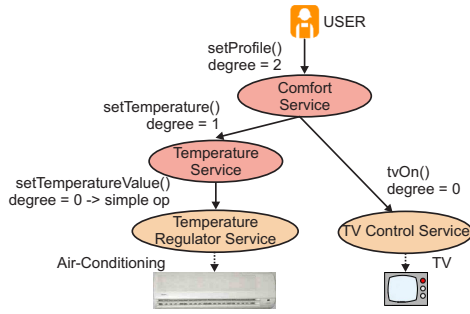


Fig. 4. Degree in composite operations.

this axiom we can prove the property of acyclicity of the *Services Composition Map*. By construction, the model verifies the acyclicity of the service composition graph with degree 1, a composite operation invokes only simple operations. We could consider a more complex composition graph with several services with composite operations interacting,  $v1$ ,  $v2$  and  $v3$ . This example must verify that  $edge(o_{op_{v1}}, op_{v2}, v2)$  and  $edge(o_{op_{v2}}, op_{v3}, v3)$ , which according to the axiom defined above involve that  $degree(op_{v1}) > degree(op_{v2})$  and  $degree(op_{v2}) > degree(op_{v3})$ . By transitivity it could also be verified that  $degree(op_{v1}) > degree(op_{v3})$ . Therefore we can affirm that no matter how complex is the composition map of a service built on DOHA, the graph associated with it is a directed acyclic graph.

#### IV. COMMUNICATION MECHANISMS IN SERVICES COMPOSITION

The DOHA communication mechanisms are based on the SOA scheme of publication, discovery and invocation of requests, which is implemented using JXTA primitives and protocols. A DOHA service takes the peer as the entity that provides communication capability. JXTA shares some of the SOA principles. The JXTA Peer Discovery Protocol is used for the publication and discovery of services using the concept of advertisement as exchanged information between peers. In contrast, for inter-peer communication JXTA provides three basic transport mechanisms, each one providing a different level of abstraction. The endpoint is the lowest level transport mechanism, followed by the pipe, and then finally, at the highest level, there are JXTA sockets [16]. DOHA exploits the pipe as the basic facility in order to provide a communication channel between peers in DOHA.

The DOHA services as peers publish their presence making use of advertisements to allow other services to discover it. Therefore, the services need discovery mechanisms that allow communication among services with different locations and functionalities. In JXTA special peers exist, the Relay Peer and the Rendezvous Peer, which provide remote discovery of advertisements between peers in different networks. DOHA treats these types of peers in the implementation of a Directory Service.

From a purely P2P point of view, JXTA does not require a specific service node to provide registry services. However,

JXTA is versatile enough to accommodate a brokered mode of operation as well, whereby Rendezvous/Relay nodes can take over the role of the Registry and Lookup servers. The Rendezvous/Relay peer nodes can manage requests and responses to facilitate communication between pairs of peers [17].

When a service discovers new advertisements in the network, it stores them in the local cache of its peer. If this service is also connected to the Service Discovery (Rendezvous peer), it can also perform a remote search and discover more advertisements. A Rendezvous peer has the responsibility of coordinating all peers in the JXTA net and propagating the messages and advertisements remotely. If the peers are in separate subnets, we can use any one Rendezvous peer to manage the reception of remote messages and broadcast those within its local net. If the local net has a firewall or NAT (Network Address Translation), the peer can use a Relay peer to surpass it and allow remote discovery of its advertisements by peers of other external networks.

Each peer of any DOHA service has a pool of pipes with a name, an unambiguous identifier, and a pipe advertisement for each one. This information is known by the rest of the peers in the peer group due to its publication into a pipe advertisement, and in addition it could be delivered to remote networks through the Rendezvous Peer. When a service wants to establish communication with another, first it seeks one of those announcements in the cache of his peer or from the Directory Service, and then it uses this information for the pipe creation.

The peer associated with each service have a special pipe, named the interaction pipe or *PipeI*, to claim the use of collaborated services as an input pipe in a customer service. This pipe allows the peers to act as consumers in the network.

#### V. BEHAVIOUR FLOW OF THE SERVICE COMPOSITION MANAGEMENT

The behaviour of a service can be represented using an UML activity diagram that models how the services interact with each other. The execution flow in a service can be distinguished in function of the abstraction layer where the activity takes place. Therefore, we have partitioned the set of activities of a service according with the layer in which these operations take place: Interface, Application and Interaction. Each partition has a group of actions with respect to their responsibility as we show in Fig. 5.

The services execute the activity of *waiting for messages* in the partition associated with the *Interface Layer*. During this activity the services listen to their input pipes waiting to receive requests from other services. Once a message arrives, it is delivered to the *Application Layer*, which will be responsible for processing the message.

The bulk of the operations in a DOHA service is composed by the group of activities associated with the *Application Layer*. When a request is received on the interface, the flow passes to this layer in order to determine the type of the request. The types of request that a service can receive are related with the next activity to perform. The main activities

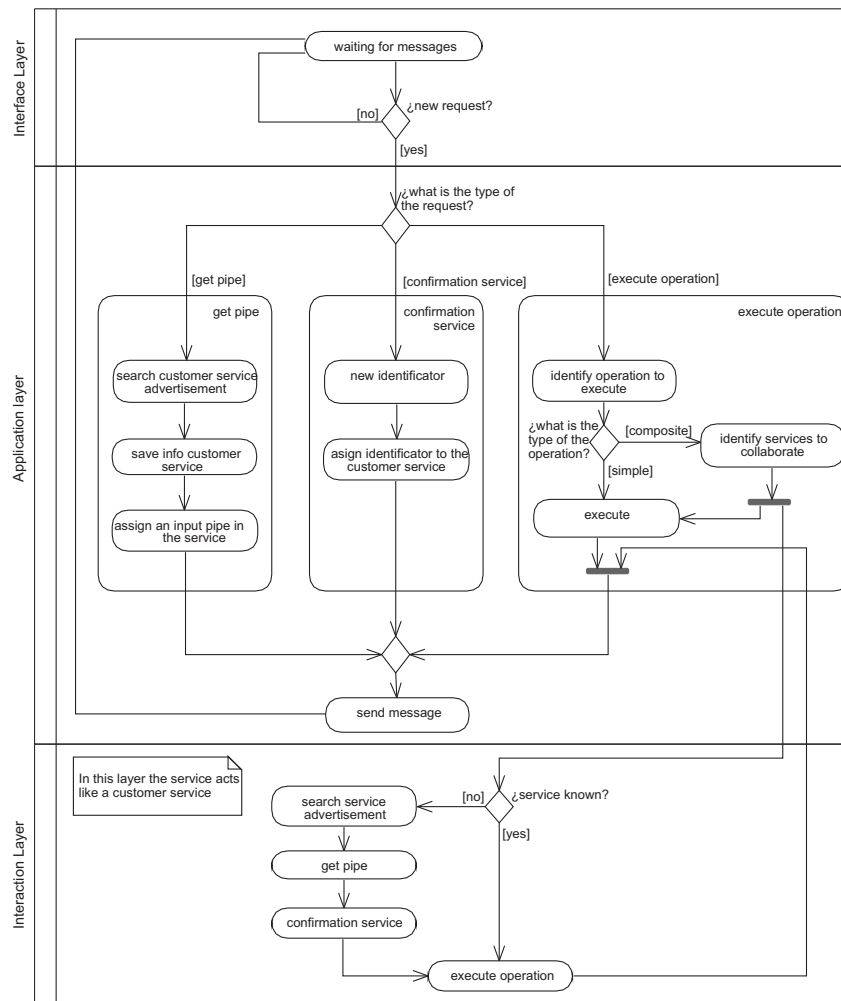


Fig. 5. Behaviour of the execution flow of a service based on their activities.

in a service are grouped as a subgroup of activities: *get pipe*, *confirmation service*, and *execute operation*. Depending on the type of the received request, the execution flow will continue for one or a subset of activities in this partition. All the groups of activities in the application partition finalize by sending a message to the service customer with information about the request. Fig. 6 shows the messages involved in the communication between a Customer Service and Provider Service by means of an UML sequence diagram.

The sub-activity *get pipe* has the function of associating an input pipe with the service customer in the service. The first activity of this group is *search a customer service*. This activity locates the customer advertisement in the network. Then, the activity of *save info customer* determines who is the service customer and what is his pipe of communication from the information contained in the customer advertisement. Finally, in the activity *assigns an input pipe* a pipe service is associated with this customer.

The *confirmation service* activity allows the customer service to know if the service is available before communicating

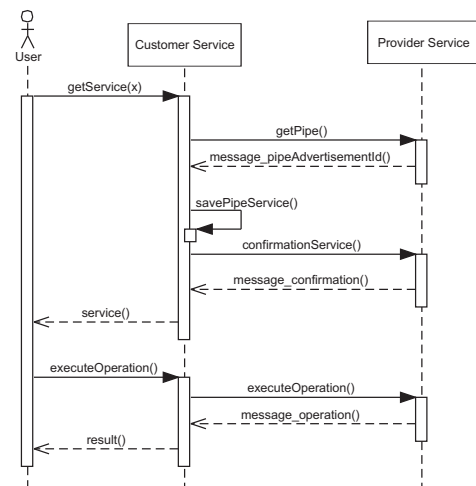


Fig. 6. Communication flow between services.

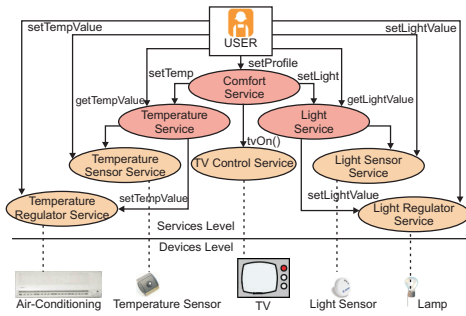


Fig. 7. Example of services composition.

with it. This activity assigns identifications to each customer and saves this information about what customer services are communicating with it.

Finally, the *execute operation* activity performs the functionality of the service executing its operations. The first activity in this group determines what operation of the service is the one that the service customer wants to execute. If the operation is simple, the service can execute it itself. Otherwise, when the operation requires the collaboration with other services, the activity of identifying services to collaborate is performed inspecting the *Service Composition Map*. Remember that the service in the execution of a composite operation can only know the required operations from other services of lower complexity degree with a depth of one level according to the service composition graph. Subsequently, the control flow must pass to the interaction partition, because the service must act as a customer of other services.

The last partition of activities is the *Interaction Layer*. The activities of this partition are fundamental in the behaviour of the DOHA services and in the composition model. This partition contains the activities that enable the service to communicate with others services acting as a consumer and to manage a composite operation. In this partition the service performs the activities as if he was a customer, *get pipe*, *confirm the service* and *execute operation*. Finally, returns the execution flow to the partition of the *Application Layer*.

## VI. AN EXAMPLE OF APPLICATION TO A COLLABORATIVE SERVICES SYSTEM

To illustrate the model of service composition in DOHA platform we are going to show an example scenario where different services implement their functionality using composite operations which involve the collaboration between services. The goal of the example application is to develop a large enough number of services to cover the user needs for home-automation. The implemented services are *Comfort Service*, *Light Service*, *Light Sensor Service*, *Light Regulator Service*, *TV Control Service*, *Temperature Service*, *Temperature Sensor Service* and *Temperature Regulator Service*, as we show in Fig. 7.

From the point of view of the owner of the composite operation, the service composition process starts when it receives a request for this operation from a service consumer.

After receiving the composite operation request the service searches for the services involved in the operation. When the communication between services can be carried out, the service owner of the composite operation acts as customer of the rest of the services. The users that request for a composite operation do not know that it involves the collaboration between services and the composition of their operations. The users only make the request and receive a response from the service owner of the composite operation.

Each service has its own directed graph which models its composite operations by means of  $map(S)$ . The set of all composition graphs in an application form the total collaborative execution flow of the application based on composite operations. By construction, we can define the system composition graph of an application as the union of individual composite operation graphs of the services of the application  $app$ , i.e.  $map(app) = \cup_i map(S_i) = \cup_{i,j} map(G_{op_{jS_i}}) = \cup_{i,j} (o_{op_{jS_i}}, V_{jS_i}, L_{jS_i}, E_{jS_i})$ . The system composition graph can be built dynamically in order to know the *Service Composition Map* of the running services at any time in contrast with other works that require a static definition of the system composition graph [18]. This information could be relevant when a study of performance, reliability, workload, end-to-end delay or any other QoS parameter should be performed.

## VII. RELATED WORK

According to Peltz, the language used to describe the flow of collaborative processes must satisfy the requirements of i) asynchronous invocation and ii) exception handling and integrity in transaction [9]. These requirements are met on the DOHA platform, because the invocation of the operations is asynchronous and is managed by the invoker service which is responsible for handling any exceptions thrown by the specific operation.

A research work with some similarities with the presented in this paper is shown in [19], which uses the P2P technology to support the design of a services middleware. It distinguishes between the actions of peers as service providers, consumers, or both at once. The main objective of the author's research is to present a services collaboration model based on the properties of the services which are crossed semantically to obtain services with more complex features. To achieve this, the authors introduce a formal model for services composition based on the semantic properties of the services, which is similar to the collaboration model used in DOHA, that also is represented using a formal model based on the graph theory.

The high complexity of pervasive systems makes the development of applications with the non-functional properties required for the SOA paradigm difficult. A well-know strategy to overcome this complexity is the use of a centralized architecture based on a gateway [18]. Nakamura proposes a service collaboration model based on behaviour profiles created from a collaboration graph and stored in a centralized configuration file (SMI definition - Service method invocation). The collaboration between services in DOHA is also based on a file which stores services and operations related



with the composite operations of each service. However, it is private and known only to the service which is associated, other services do not know how it is or how the service works to carry out their composite operations. The main difference between the composition model designed by Nakamura et al. and the presented in this research is the scope of the composition map. In the first case, the composition model is defined at system level and the composition model can be defined previous to the definition of the services. In our case, the composition model is defined at service level and requires the knowledge of the *Service Contract* of the services with which the collaboration will be performed to form the composition map.

Ontologies are a knowledge representation model being used in a wide range of research works to manage the services collaboration, as in the case of *SOAM* of Vazquez et al. [20]. *SOAM* is an experimental model for the creation of smart objects using ontologies on the web, i.e. Semantic Web technologies to enable communication between the semantic context and reasoning processes in order to provide an adaptation of environment to user preferences. It also uses behavioural profiles, based on which it provides a service collaborative model between different semantic objects in the environment. The main short-term objective in the development of DOHA is to become a platform for context-sensitive services, thus linking the development of pervasive computing applications with the development of ambient intelligence applications using ontologies as the method for the representation of the context information and its relation with the services.

### VIII. CONCLUSIONS

Development and deployment of service-based applications on embedded devices are highly complex tasks. These devices have very limited resources, little processing power and memory. Hence, the programs developed on them must optimize the use of scarce resources. For this project we choose to implement an engineering process to model a services architecture based on SOA principles and use the capabilities of the peer-to-peer JXTA platform, with a well-defined model of communication, behaviour and collaboration. Using SOA as the design philosophy can improve relations between technology and services development that support the needs of users, broadening the range of possibilities that the various applications built on the platform can offer. The ability of optimally achieving collaboration between services is a key factor for the competitiveness and growth of a services platform. The use of lightweight services composition maps based on directed acyclic graph and distributed in each service, can create new added-value services that release the potential of the applications and resources used by the platform, raising user satisfaction with them. Furthermore, the proposed composition model allows the services to define their individual composition maps, which are smaller and easy to manage and store in embedded systems than the models which define the composition flow of the whole application. A further advantage of the model presented with respect

of others is its distributed nature, being a model based in individual composition maps, each service controls its own map of collaboration, without requiring a centralized node holding the information of system-wide collaboration.

### ACKNOWLEDGMENT

This research is partially supported by the Spanish Ministry of Education and Science through a pre-doctoral FPU grant.

### REFERENCES

- [1] P. Lalanda, L. Bellisard, and R. Balter, "Asynchronous mediation for integrating business and operational processes," *Internet Computing*, vol. 10, no. 1, pp. 56–64, 2006.
- [2] A. Yachir, K. Tari, Y. Amirat, A. Chibani, and N. Badache, "Mdp and learning based approach for ubiquitous services composition," in *2010 IEEE Globecom Workshops, GC'10*, 2010, pp. 1668–1673.
- [3] D. Saha and A. Mukherjee, "Pervasive computing: A paradigm for the 21st century," *Computer*, vol. 36, no. 3, pp. 25–31+4, 2003.
- [4] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 256, pp. 94–104, 1991.
- [5] G. Banavar, J. Beck, E. Gluzberg, J. Munson, J. Sussman, and D. Zukowski, "Challenges: An application model for pervasive computing," in *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, 2000, pp. 266–274.
- [6] Z. Stojanovic and A. Dahanayake, *Service-oriented software system engineering: challenges and practices*. Idea, 2005.
- [7] S. L. Kiani, M. Riaz, Y. Zhung, S. Lee, and Y. . Lee, "A distributed middleware solution for context awareness in ubiquitous systems," in *Proceedings - 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, 2005, pp. 451–454.
- [8] J. Bronsted, K. M. Hansen, and M. Ingstrup, "A survey of service composition mechanisms in ubiquitous computing," in *Proceedings of UbiComp 2007 Workshop Innsbruck, Austria*, vol. 4717, no. 9, 2007, pp. 87–92.
- [9] C. Peltz, "Web services orchestration and choreography," *Computer*, vol. 36, no. 10, pp. 46–52, 2003.
- [10] R. M. Pessoa, E. Silva, M. Van Sinderen, D. A. C. Quartel, and L. F. Pires, "Enterprise interoperability with soa: A survey of service composition approaches," in *Proceedings - IEEE International Enterprise Distributed Object Computing Workshop, EDOC*, 2008, pp. 238–251.
- [11] J. F. Buford and H. Yu, "Peer-to-peer networking and applications: Synopsis and research directions," in *Handbook of Peer-to-Peer Networking*. Springer US, 2010, pp. 3–45.
- [12] L. Barolli and F. Xhafa, "Jxta-overlay: A p2p platform for distributed, collaborative, and ubiquitous computing," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 6, pp. 2163–2172, 2011.
- [13] S. Rodríguez and J. Holgado, "A home-automation platform towards ubiquitous spaces based on a decentralized p2p architecture," in *International Symposium on Distributed Computing and Artificial Intelligence 2008 (DCAI 2008)*. Springer Berlin / Heidelberg, 2009, pp. 304–308.
- [14] J. Holgado-Terriza and S. Rodríguez-Valenzuela, "Service oriented middleware for home-automation," 2011, submitted to Journal of Network and Computer Applications.
- [15] J. A. Holgado-Terriza and J. Viúdez-Aivar, "A flexible java framework for embedded systems," in *ACM International Conference Proceeding Series*, 2009, pp. 21–30.
- [16] L. Gong, "Jxta: A network programming environment," *IEEE Internet Computing*, vol. 5, no. 3, pp. 88–95, 2001.
- [17] R. L. McIntosh, "Open-source tools for distributed device control within a service-oriented architecture," *JALA - Journal of the Association for Laboratory Automation*, vol. 9, no. 6, pp. 404–410, 2004.
- [18] M. Nakamura, A. Tanaka, H. Igaki, H. Tamada, and K. . Matsumoto, "Constructing home network systems and integrated services using legacy home appliances and web services," *International Journal of Web Services Research*, vol. 5, no. 1, pp. 82–98, 2008.
- [19] J. Gerke, P. Reichl, and B. Stiller, "Strategies for service composition in p2p networks," in *E-business and Telecommunication Networks*. Springer Berlin Heidelberg, 2007, vol. 3, pp. 62–77.
- [20] J. Vazquez, I. Sedano, and D. López de Ipiúña, "Soam: A web-powered architecture for designing and deploying pervasive semantic devices," *International Journal of Web Information Systems*, vol. 2, no. 3, pp. 212–224, 2007.

# Violation of Service Availability Targets in Service Level Agreements

Loretta Mastroeni

Department of Economics  
University of Rome Tre

Via Silvio D'Amico 77, 00145 Roma, Italy  
E-mail: mastroen@uniroma3.it

Maurizio Naldi

Department of Computer Science  
University of Rome at Tor Vergata

Via del Politecnico 1, 00133 Roma, Italy  
E-mail: naldi@disp.uniroma2.it

**Abstract**—Targets on availability are generally included in any Service Level Agreement (SLA). When those targets are not met, the service provider has to compensate the customer. The obligation to compensate may represent a significant risk for the service provider, if the SLA is repeatedly violated. In this paper we evaluate the probability that a SLA commitment on the service availability is violated, when the service restoration time follows an exponential, Weibull, or lognormal distribution. For a two state model, where the service alternates between availability and unavailability periods, we show that such probability decreases as the variance of the restoration time grows, and that lengthening the time interval over which the service availability is evaluated reduces the risk for the service provider just if the compensation grows quite less than the length of that time interval.

## I. INTRODUCTION

**S**ERVICE level agreements (SLAs) define the contractual obligations of the service provider towards the customer, the mechanisms to enforce the delivery of the committed service quality, and the obligations of the service provider if the service level falls below the committed value [1], [2].

A key parameter in SLAs is the service availability, for which a target figure is declared. If the service is unavailable, the customer is not provided what it has paid for (an economical loss in itself), but suffers an additional larger loss due to the discontinuity in its business operations or social relationships. The latter category of losses may reach values of the order of 100-200 k\$ per minute of service interruption (see [3]).

The obligations of the service provider generally consist in the payment of a sum if the target availability figure is not met. If the service availability targets are not met repeatedly, the service provider is bound to suffer large losses, especially if that happens on a massive scale rather than for a few individual customers.

The service provider has to be able to evaluate the risk it incurs because of SLA violations, which in turn requires to evaluate the probability that the target figures are not met. Many SLA monitoring tools exist: HP OpenView Firehunter, CiscoWorks2000 Service Management Solution, and Lucent's CyberService, among those marketed in the recent past. However, SLA monitoring tools allow to perform an *ex-post* evaluation of the quality of service delivered, rather than the predictive evaluation that the service provider needs to properly set its commitment and negotiate a sustainable SLA.

In [4] the probability of violating availability SLAs has been evaluated by simulation, when the service restoration time follows an exponential distribution, but two more complex distributions are envisaged for the service restoration time, namely the Weibull and the lognormal. The same Weibull distribution is suggested in [5] as the best-fit model in the cases of grid computing services.

In this paper we provide a thorough examination of the risk of violating the SLA obligations, considering three models (exponential, Weibull, and lognormal) for the service restoration time. We provide an analytical expression for the probability of violating an availability SLA commitment for the exponential case, while previous results were based on simulation only. We provide simulation results for the same probability of violation, when the service restoration time follows instead a Weibull or lognormal distribution, which had not been dealt with in the literature. We show that the probability of SLA violation decreases as the variance of restoration times grows, and that lengthening the time interval over which the availability targets are examined is convenient for the service provider just if the compensation amount for each violation grows quite less than proportionally with the length of that time interval.

The paper is organized as follows. In Section II, we define the service model we adopt for our analysis. In Section III, we provide a formal definition of the service level agreement for availability and of the compensation policy. Finally, we provide in Section IV the results for the three models considered.

## II. SERVICE MODEL

In order to assess the violations of SLA obligations, we need a model for the service provided to the customer. In this section, we describe a simple model based on the alternation of ON and OFF states, and provide the definition of availability.

We consider the service to be either available or not. Though the customer could experience a graceful performance degradation, SLA commitments are sharp [6]. At time  $t$ , the state  $S_t$  of the service equals 1 if the service is available, and 0 otherwise. The service undergoes a sequence of alternating availability and unavailability (ON and OFF) states, whose average durations are respectively the *Mean Time To Failure* (MTTF) and *Mean Time To Repair* (MTTR). The durations of the OFF periods are represented by the sequence of positive

i.i.d. random variables  $\{B_1, B_2, B_3, \dots\}$ . We assume that the service starts in the ON state. The variable  $N_T \in \mathbb{N}_0$  represents the number of failures in the period  $(0, T]$  ( $N_T = 0$  means that the service works uninterruptedly in  $(0, T]$ ).

The service model is fully specified when we define the probability distribution for the duration of the ON and OFF periods. Here we assume that the duration of the ON period follows an exponential distribution, and the duration of the OFF period follows either an exponential distribution, or a Weibull, or a lognormal distribution. Two-parameter models (lognormal and Weibull) are used for more complex repair scenarios such as with significant travel time [4].

As the key service performance parameter for our model, we consider the availability. For our two-state model, the steady-state availability  $\Phi$  is defined as the expected value of the state variable [7], or, equivalently, as the probability that the service is ON, and can also be expressed through MTTF and MTTR:

$$\Phi = \mathbb{E}[S_t] = \mathbb{P}[S_t = 1] = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}. \quad (1)$$

### III. SERVICE LEVEL AGREEMENTS AND COMPENSATION POLICIES

In SLAs, the service provider commits itself to provide an adequate quality of service, and compensate the customer if that commitment is not honored. In this section, we review the definition of service availability targets, and describe the compensation policy considered in the following.

In SLAs, target values are indicated for service availability [8], [9]. In the basic definition (1), we must state what the object is whose availability we consider, and how we declare that object to be available or not. For example, in [10] a list of services and the associated definitions of availability are provided.

In order to check if the SLA obligations are met in an operational context, we set an observation interval  $T$  and measure the availability through the ratio of the cumulative outage duration  $X_T$  during the observation interval and the length of the observation interval itself:

$$\hat{\Phi} = \frac{T - X_T}{T} = 1 - \frac{\sum_{i=0}^{N_T} B_i}{T}. \quad (2)$$

If we fix the length of the observation interval, the SLA obligation  $\hat{\Phi} > z$  (the threshold  $z$  being a positive quantity) can be expressed as the constraint  $X_T < W = (1-z)T$  on the cumulative outage duration  $X_T$  over the observation interval. In particular, we can set a threshold  $z = \Phi$  equal to the declared steady-state availability. However, due to the random nature of the failure process, there is a non-zero probability that the SLA obligation is violated.

The compensation policy states what the service provider is to pay its customer when the service fails. We assume that the compensation is paid out for failures occurring over a period of time of extension  $T$  (the observation period), rather than on each single failure. In this paper, we consider a simple compensation policy based on the steady-state availability: a fixed amount of money is paid when  $X_T > W = (1 - \Phi)T$ .

### IV. PROBABILITY OF VIOLATION OF AVAILABILITY TARGETS

The risk for service providers, deriving from the unfulfilled commitments, depends on the probability of violating the SLA targets. In this section, we provide results for the cases where the service restoration times follow an exponential, a Weibull, or a lognormal distribution. For the exponential case, we obtain an approximate analytical expression. Instead, for the Weibull and lognormal cases, we resort to simulation.

**Exponential restoration times.** In services with high availability, we have  $\text{MTTR} \ll \text{MTTF}$ . In that case, the process of failure occurrences can be approximated by a Poisson process. Over a finite horizon  $T$ , the number  $N_T$  of failures follows approximately a Poisson distribution with average value  $\lambda T$ , where  $\lambda$  is the failure rate. By the total probability theorem, and recognizing that the number of failures and the duration of the outages are independent of each other, we can express the probability of violation as

$$\begin{aligned} \mathbb{P}[X_T > W] &\simeq \mathbb{P}\left[\sum_{i=0}^{N_T} B_i > W\right] \\ &= \sum_{k=0}^{\infty} \mathbb{P}[N_T = k] \cdot \mathbb{P}\left[\sum_{i=0}^{N_T} B_i > W \mid N_T = k\right] \\ &= \sum_{k=1}^{\infty} \mathbb{P}[N_T = k] \cdot \mathbb{P}\left[\sum_{i=1}^k B_i > W\right] \end{aligned} \quad (3)$$

Since the durations of outages are i.i.d. random variables with an exponential distribution, their sum follows an Erlang distribution. When there are  $k$  failures, the probability distribution of the cumulative outage duration is

$$\begin{aligned} \mathbb{P}\left[\sum_{i=1}^k B_i \leq x\right] &= \int_0^x \frac{\mu^k e^{-\mu v} v^{k-1}}{(k-1)!} dv \quad x > 0 \\ &= \frac{1}{(k-1)!} \gamma(k, \mu x), \end{aligned} \quad (4)$$

where  $\gamma(k, x)$  is the lower incomplete Gamma function [11].

By replacing the expression of the Poisson distribution and the result (4) in the probability of violation (3), we obtain the final expression

$$\begin{aligned} \mathbb{P}[X_T > W] &= \sum_{k=1}^{\infty} \frac{(\lambda T)^k}{k!} e^{-\lambda T} \left[1 - \frac{1}{(k-1)!} \gamma(k, \mu W)\right] \\ &= 1 - e^{-\lambda T} - \sum_{k=1}^{\infty} \frac{(\lambda T)^k}{k!(k-1)!} e^{-\lambda T} \gamma(k, \mu W) \end{aligned} \quad (5)$$

The resulting probability of violating the SLA depends on the failure rate  $\lambda$ , the observation interval  $T$ , and the threshold  $W$  for the overall duration of outages  $W = (1 - \Phi)T$ .

We report in Fig. 1 the probability of SLA violation for several values of the steady-state availability, from 95% to the excellent five nines case. We assume  $\text{MTTR}=4$  hours, a value typically adopted (see Chapter 2.2 in [12]). The range of values

TABLE I

EXPECTED NUMBER OF TARGET VIOLATIONS OVER ONE YEAR

Obs. int. $T$ [months]	Viol. prob. over $T$	Expected violations
1	0.140	1.679
2	0.224	1.345
3	0.277	1.109
4	0.312	0.936
6	0.353	0.706
12	0.401	0.401

for the observation interval goes up to 1 year, corresponding to 8640 hours. The probability of violating the targets grows

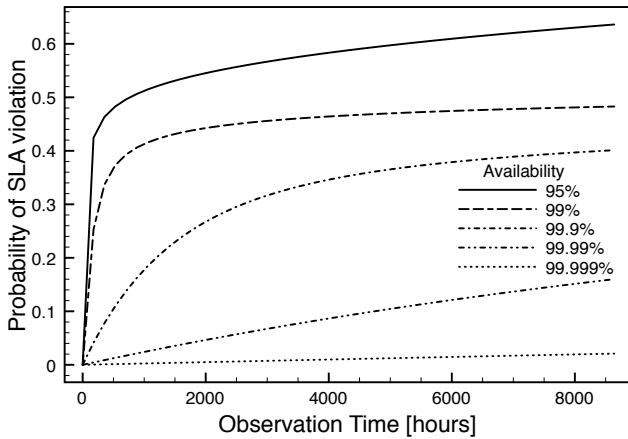


Fig. 1. Probability of violation under exponential restoration times

with the observation interval. This would seem contradictory with the generally held view that adopting long measurement intervals favors the service provider, since it allows to smooth out single events of protracted unavailability. Actually, we have to assess the overall number of violations taking place over an evaluation period of fixed length. If we reduce the length of the observation interval  $T$ , the probability of violation over  $T$  reduces, but the number of observation intervals included in the evaluation period increases. For example, for the case where  $A = 99.9\%$  and  $MTTR = 4$  hours, we see in Table I that the expected number of violations over a year actually decreases as we lengthen the observation interval: service providers may reduce their risk by lengthening the observation interval. However, we should consider the economical loss deriving from the application of the compensation policy. We expect the compensation sum to increase as the observation interval lengthens. We should therefore multiply the expected number of violations (third column in Table I) by the compensation paid for each violation. The data in the table show that lengthening the observation interval from one month to one year reduces the overall expected loss if  $C_{12}/C_1 < 1.679/0.401 \simeq 4.19$ , i.e., quite less than 12 times, the increase in the observation interval.

If the service provider adopts a threshold on the measured availability larger than the steady state availability, it may

bring the probability of violation down to acceptable values. For example, in [13] it is envisaged that the service provider may revise the performance objectives (e.g., by relaxing the constraint on the availability), if the SLA obligations are not being met. In Fig. 2, we see that the probability of violating the SLA obligation decreases when we raise the threshold over the steady-state availability (43 minutes for  $T = 1$  month and 2 hours 19 minutes for 6 months, when the steady-state availability is 99.9% and  $MTTR=4$  hours).

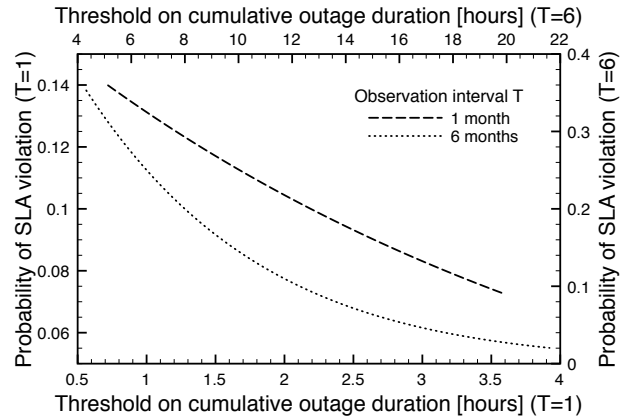


Fig. 2. Impact of limit outage duration on the probability of SLA violation

**Weibull restoration times.** The probability distribution of restoration times may differ from the exponential. For example, in [5] a Weibull distribution is proposed to model the duration of outages in grid computing.

Under the Weibull hypothesis, the probability distribution for duration of the generic  $i$ -th outage is

$$\mathbb{P}[B_i < x] = 1 - e^{-(x/\sigma)^\theta} \quad x \geq 0, \quad (6)$$

where  $\sigma$  is the scale factor, and  $\theta$  is the shape factor. When  $\theta = 1$  the Weibull distribution becomes the exponential one. For the case of grid computing, the shape factor should lie in the  $[0.6, 1]$  range [5]. When  $\theta < 1$ , the variance of the service restoration time increases with respect to the exponential case.

We determine the probability of violation by simulation, since no closed form exists for the distribution of the sum of i.i.d. Weibull random variables. We consider  $10^5$  instances of the observation interval, generating the number of failures according to a Poisson process, and the duration of each outage through a Weibull-distributed random number.

In Fig. 3 we show the probability of violating the SLA, when  $MTTR=4$  hours and  $\theta = 0.7$  (a standard deviation slightly lower than six hours, against the four hours of the exponential case). Despite the larger variance of the restoration time in the Weibull case, the violation probability is slightly lower than in the exponential case.

In Fig. 4, we examine in greater detail the impact of the shape factor, when the steady-state availability is 99.9%. Since the variance of the service restoration time grows as  $\theta$  decreases, the probability of SLA violation decreases as the

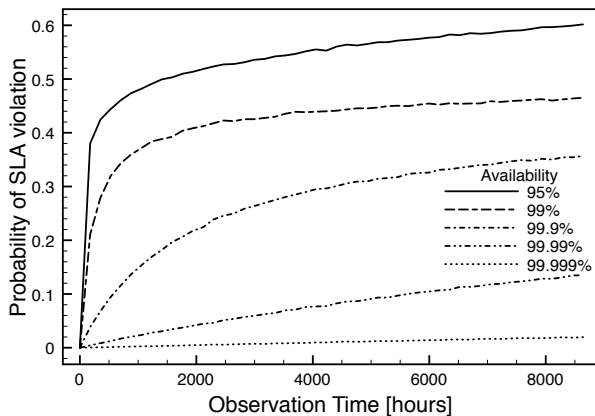


Fig. 3. Probability of violation under Weibull repair times ( $\theta = 0.7$ )

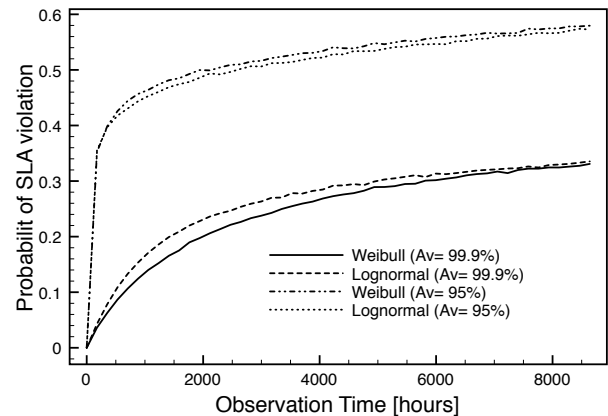


Fig. 5. Comparison between the Weibull and the lognormal case

variance grows. When  $\theta = 0.6$ , the violation probability is roughly 17.5% lower than in the exponential case.

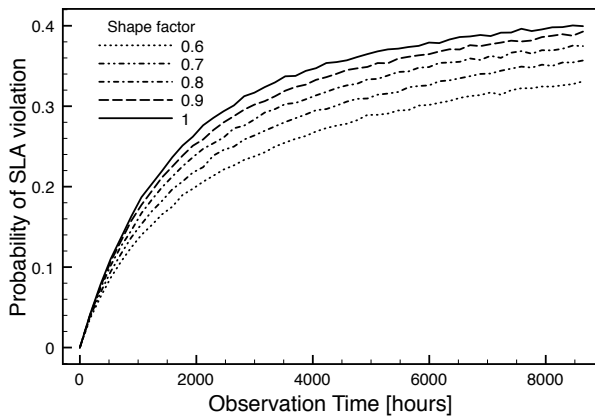


Fig. 4. Probability of violation under Weibull repair times ( $\Phi = 99.9\%$ )

**Lognormal restoration times.** In addition to the exponential and the Weibull case, the lognormal model has been proposed in [4] for the service restoration times.

Again, no closed form exists for the probability distribution of the sum of lognormal random variables, and we resort to simulation. We adopt a restoration time with mean value MTTR=4 hours, and a standard deviation ranging from 4 hours (as in the exponential case) to 7 hours (as in the Weibull case with  $\theta = 0.6$ ), and  $10^5$  simulation instances. In Fig. 5, we compare with the Weibull case for the largest standard deviation of the service restoration time (7 hours): the differences are larger for the high availability case ( $\Phi = 99.9\%$ ), but quite negligible when the steady-state availability is not very large, and smooth out as the observation interval lengthens.

## V. CONCLUSION

We have evaluated the probability that the availability commitments included in a Service Level Agreement are not met. The analysis has been conducted for a two-state service

model, with alternating periods of service availability and service restoration. Three probability models have been considered for the service restoration times: exponential, Weibull, and lognormal. We have shown that the availability target values are less likely to be violated as the variance of the service restoration time gets larger. If the service providers opts for longer evaluation intervals (for the assessment of SLA commitments), it must set compensations quite less than proportional to the length of the observation interval itself.

## REFERENCES

- [1] A. Keller and H. Ludwig, "The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services," *J. Network Syst. Manage.*, vol. 11, no. 1, pp. 57–81, 2003.
- [2] H. Ludwig, A. Keller, A. Dan, R. P. King, and R. Franck, "A Service Level Agreement Language for Dynamic Electronic Services," *Electronic Commerce Research*, vol. 3, no. 1-2, pp. 43–59, 2003.
- [3] M. Pesola, "Network protection is a key stroke," *Financial Times*, FT Business Continuity, March 9, 2004.
- [4] A. P. Snow and G. R. Weckman, "What Are the Chances an Availability SLA will be Violated?" in *Sixth International Conference on Networking (ICN 2007)*, 2007, p. 35.
- [5] R. Alsoghayer and K. Djemame, "Probabilistic risk assessment for resource provision in grids," in *Proceedings of the 25th UK Performance Engineering Workshop*, Leeds, 6-7 July 2009, pp. 99–110.
- [6] A. Michlmayr, F. Rosenberg, P. Leitner, and S. Dustdar, "Comprehensive QoS Monitoring of Web Services and Event-Based SLA Violation Detection," in *MW4SOC 09*, Urbana Champaign, Illinois, USA, 30 November 2009.
- [7] T. Aven and U. Jensen, *Stochastic Models in Reliability*. Springer, 1999.
- [8] M. Vogt, R. Martens, and T. Andvaag, "Availability modeling of services in IP networks," in *Design of Reliable Communication Networks, 2003. (DRCN 2003). Proceedings. Fourth International Workshop on*, October 2003, pp. 167 – 172.
- [9] E. Bouillet, D. Mitra, and K. Ramakrishnan, "The structure and management of service level agreements in networks," *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 4, pp. 691 –699, May 2002.
- [10] P. Cholda, J. Tapolcai, T. Cinkler, K. Wajda, and A. Jajszczyk, "Quality of resilience as a network reliability characterization tool," *IEEE Network*, vol. 23, no. 2, pp. 11–19, 2009.
- [11] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, Eds., *NIST Handbook of Mathematical Functions*. Cambridge University Press, 2010.
- [12] E. Bauer, *Practical System Reliability*. J. Wiley-IEEE Press, 2009.
- [13] D. Verma, "Service level agreements on IP networks," *Proceedings of the IEEE*, vol. 92, no. 9, pp. 1382 – 1388, September 2004.

# Orchestration of Service Design and Service Transition

Dr. Bernd Pfitzinger  
FOM Hochschule für Oekonomie &  
Management, Bismarckstraße 107, 10625  
Berlin, Germany  
Email: bernd.pfitzinger@gmail.com

Dr. Thomas Jestädt  
Toll Collect GmbH, Linkstrasse 4, 10785  
Berlin, Germany  
Email: thomas.jestaedt@toll-collect.de

**Abstract**—Standardized service management processes and organizations allow the implementation of changes to service offerings as part of an integrated and ISO 20.000 certified service management system. Two different models for the process-based orchestration of changes to services are presented addressing the Service Design and Service Transition phases of ITIL V3. The models are evaluated in a real life scenario and discussed in the context of a medium-sized company.

## I. INTRODUCTION

EVERY company offering services as their main product depends on a working set of business processes for the service management discipline – especially in the case of highly automated services.

A common framework to address the challenges of (IT) service management is available as IT Infrastructure Library (ITIL, [1], [2]) and since late 2005 it is possible to certify the management of IT services (i.e. the organization and its service management processes) according to the international standard ISO 20.000 [3]. In that way organizations ensure that all service management processes defined in the ITIL “good practice” framework are implemented. In addition to the well-known Service Operations processes this certification also requires a working implementation of the service management processes for the complete service lifecycle: Service Strategy, Continual Service Improvement, Service Design and Service Transition.

However, the detailed business process design needs to be developed by each organization. The individual processes vary vastly in the complexity of their respective tasks, e. g. repetitive well-structured processes (incident and event management) or highly creative tasks (e. g. the design of new services or major service changes). The safe and timely implementation of service changes is increasingly becoming a vital part of the service offering (and also a significant expense). This emphasizes the importance of the Service Design and Service Transition processes. Each service change includes not only technical changes but may also introduce changes to the supporting service operations processes and the work of the specialists involved.

In this paper we describe and evaluate two different models for the orchestration of changes to services. In sections II and III we describe each of the two implementations of the service design and service transition phases [4] of ITIL V3 as used in a ISO 20.000 certified service management system at Toll Collect GmbH. Since the initial implementation in 2007 we have gathered experience with more than 1000 service changes allowing us to compare and

evaluate both models using a case study approach [20] given in section IV.

Toll Collect GmbH provides the business services for the German electronic toll for heavy trucks involving more than 50 IT services ranging from standard IT applications (e.g. central billing processes, customer relationship management, and document management) to highly service-specific customer processes. The major IT services involve a 24 by 7 setup in a fault tolerant environment consisting of several hundred servers communicating with more than 650.000 units in the field. Regular updates due to changes in the road network are modelled and transferred via GSM networks. Overall the Toll Collect system is the 11<sup>th</sup> largest federal income source and collected 4.6 billion € in 2010 [5] at an overall quality level of better than 99.75% accuracy.

### A. Changing the Service

In the Toll Collect example the overall service is in one sense almost static (i. e. the collection of the German truck toll) yet many minor and some large changes have been incorporated into the system over the last years.

These *service changes* were implemented according to the established service management processes in an ISO 20.000 certified environment. Since the initial certification in 2007 more than 10 major releases, 1.000 medium sized service changes and more than 10.000 minor changes have been deployed, including completely new systems and requirements, updates to all parts of the technology stack and some bugfixes.

Since *changing the service* is in practice a daily recurring task it should not be necessary to implement it as a standalone project but rather according to a pre-defined process covering the service design and service transition phases of ITIL V3.

### B. Service Design and Service Transition

The service design phase covers all aspects of new or changed services: portfolio, architecture, processes and metrics [4], [7]. In technology-driven services it interfaces between the software development and the service operations processes via the service transition phase.

Major service changes typically have many properties of a project (e. g. involving a large number of people for a limited amount of time a unique and singular purpose, [6]) and are therefore typically organized as projects. However, within the service management processes themselves every service change triggers similar questions:

- does the scope of the processes change?
- should the staffing for the processes be changed?
- are there new requirements for data handling (e.g. concerning the configuration management database or service catalogue [10])?
- is the basic information complete (e.g. concerning the change advisory board, additional incident escalation rules, etc.)?
- are there changes to the processes' input or output?

In the Toll Collect example these questions are summarized in process-specific checklists – sufficient to handle medium-sized service changes without the involvement of the process specialists. However, large-scale service changes (releases) and some medium sized service changes will usually cover topics that can only be resolved by the process specialists, e. g. adapting the service catalogue in accordance to the configuration management database [9] or changing the underlying sourcing contracts [10], [11], [12]. Especially highly standardized processes (e. g. incident management) need to be adjusted in the case of service changes.

Designing the service management processes for the Toll Collect system first lead to two major choices:

- how to handle process changes resulting from service changes
- how to organize service changes to assure safe, timely and complete implementation.

With respect to possible process changes there are basically three distinct possibilities:

### 1) PROCESS CHANGES AS PROJECTS

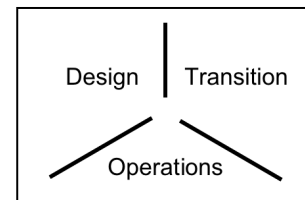
Any change to the existing service management processes is handled as a (independent) project, completely external to the service management processes and organization. This approach leverages the strengths of project management [6].

### 2) PROCESS CHANGE AS PROCESS

Since changes to existing processes are frequently occurring tasks in many quality management systems the implementation of process changes is in a strict sense rather a process in its own right than a series of independent projects. Both ITIL V3 and ISO 20.000 propose the implementation of a process dedicated to the improvement of the service management system (continual service improvement, [4],[7]) based on well-established quality management principles (e. g. the Deming-cycle [13], [14]).

### 3) PROCESS CHANGE AS INTEGRAL PART OF ALL PROCESSES

In addition to the classical quality management approach towards service improvements it is possible to separate changes to a process from mere “configuration” of the process due to changes in the services supported. To facilitate this distinction we choose to enhance each service management process to encompass a dedicated part for the service design and the service transition phase (see fig. 1).



**Figure 1:** Process design includes tasks for every service management phase

Accordingly changing the service becomes a common task within each process – albeit a specialist’s task.

Implementing the design and transition phase within each service management process has the advantage of keeping the process’ responsibility in a single place – comparable to the proper design of classes in object oriented programming [15]. Overall the ISO 20.000 compliant service management process model consists (in the given example) of 17 processes. Each process is focused to have a single responsibility and the ability to perform all tasks required to fulfil the responsibility (e. g. by collaborating with other processes by pre-defined interfaces).

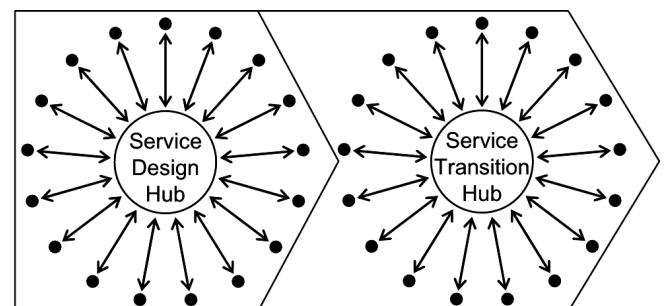
The orchestration of all processes and their respective activities within the service design and transition phases remains as the challenge in the design of a viable service management process model. In the example we decided against the implementation of cascaded interfaces, i.e. we do not treat service changes as “torch relay”. In the Toll Collect example two different orchestration models have been implemented subsequently within the ISO 20.000 certified service management system.

The following two sections give a description of the two orchestration models followed by a section evaluating the benefits and strengths of each model.

## II. HUB-AND-SPOKE MODEL

The model first implemented establishes two central hubs – one responsible for the service design phase, the other responsible for the service transition phase (see fig. 2).

The overall control of the service design phase resides with the service management process. It creates the design of the service change and delegates possible process changes and the configuration of the service management processes to each process via the service design hub.



**Figure 2:** Hub-and-spoke model with dedicated process hubs for the orchestration of the service design and service transition phase



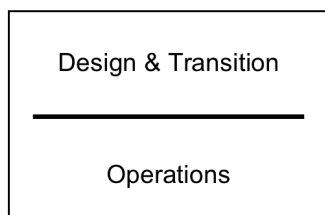
In practice most tasks regarding the service management processes can be resolved within a single process (e. g. enhancing the change advisory board). Therefore in most cases there is no need of two or more processes to collaborate (clearly a consequence of a successful separation of concerns [16]).

However, non-local changes to the processes can occur (e. g. changes to the service catalogue and configuration management database). Lacking the “torch relay” interfaces between processes these non-local changes need to be reflected back to the service design hub. Its responsibility is to resolve the non-local process change by involving all affected processes (and negotiating their requirements in the context of the service change).

In the Toll Collect example two separate hubs were implemented – one for the service design phase, the other for the service transition phase. This choice is obvious because the design phase emphasises different skills than the transition phase. As a consequence the organizations staffing each hub can make use of specialists very efficiently since each hub concentrates on one specialized responsibility. However, the resulting handover introduces an additional challenge and is a possible source of errors.

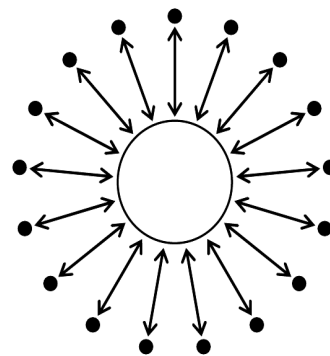
### III. ONE-STOP-SHOP MODEL

An alternative approach is also used in the Toll Collect example: the separation into two hubs with distinct responsibilities is abandoned in favour of a single point with overall implementation responsibility (for the whole service design and transition phase of a given service change). Accompanied by a minor change to combine the tasks of the service design and transition phases within each process (see fig. 3) this allows for a single hub within the service management process to orchestrate the complete service design and transition of a service change. The specialized tasks of each process remain unchanged (within each process).



**Figure 3:** Process design with combined activities from the service design and transition phases

As a consequence the responsibility for a service change remains continuously with a single orchestration hub (fig. 4) and therefore possibly with a single organization (or even person). However, this alternative service management model creates different demands for skill-set of the people involved. Rather than using specialists for the design phase and different specialists for transition phase, this model requires a combination of both in a single process role (and possibly organization or person).



**Figure 4:** One-stop-shop model with a single process hub for the service design and transition phases

## IV. MODEL EVALUATION

The scenario of more than ten major service changes and more than 1.000 medium-sized ones suggests using a service management system that treats service changes as a standard and frequently recurring transaction. Consequently in the Toll Collect example changes to a service and its related processes are implemented as an integral part of the service management processes. This follows well-established practices from object-oriented programming [17], e.g. focussing on clearly defined responsibilities and a separation of concerns.

The missing link is the orchestration of a single given service change across all service management processes and across the service design and transition phases to yield a new service offering smoothly rendered by the standard service operations processes. From a control point-of-view we decided against a “torch relay” approach in the design and transition phases: processes are not allowed to forward tasks. Using the experience of almost daily service changes since the initial certification in 2007 we have gathered extensive experience with the two different models of orchestration allowing us to evaluate the two models.

### A. HUB AND SPOKE MODEL

The main benefit of the “hub-and-spoke” approach is a further degree of abstraction allowing for an efficient assignment of specialists: Not only is the orchestration of service changes concentrated within one process (the one responsible for service offerings and service changes) but rather an additional separation into two different responsibilities (i.e. for a successful design phase and a successful transition phase) is directly implemented into the service management system. Depending on the size of the underlying service management organization this can lead to the introduction of too many process roles (e. g. a role each for resolving incidents, the design phase and the transition phase of the incident management process). Therefore even a clearly defined and simple process model can put too many process roles on a single person.

As a benefit, the separation and specialisation between the service design and service transition phases introduces an additional handover that allows for an improved planning (in

the sense of sub-projects and an intermediate milestone). It can of course also introduce the need for additional documentation and pose new communication problems.

#### B. ONE-STOP-SHOP MODEL

To address these issues we evolved the service management process model in the example to emphasize end-to-end responsibility. The responsibility stays within one place (even one process role and preferably one person) for the whole service change. At any given time there is a single responsible process role (person) for each given service change, the responsibility rests with the same role (person) until the successful completion of the service transition phase. This minor change to the underlying service management process model allows eliminating many process roles by combining the previously distinct roles for the service design and the service transition phase in each of the 17 service management processes. In practice this distinction was mostly theoretical since most often both roles were delegated to the same process specialist (e. g. the incident management specialist for service design and the one for service transition were two distinct roles delegated to the same person).

From the customers' point-of-view the "one-stop-shop" model produces better and more dependable results (as seen by internal customer reviews [18], [19]) – in accordance with theoretical expectations when introducing a "case worker" approach in a business process reengineering scenario [8].

In that way service changes as creative human tasks are transformed into a more generic combined task of design and transition. This is in contrast to the "hub-and-spoke" model which favoured the more efficient use of specialists within the design and transition phase of service changes.

#### V. CONCLUSION

The business of rendering service for a customer is (regardless of the level of technology involved) mostly a people business. The design of the underlying service management process determines the successful service operation. Real-life services routinely require many changes – even large-scale changes are no exception – often involving the cooperation or even change of the established service management processes necessary for to day-to-day service operations. We showed how to incorporate service and process changes into the service management system following simple lessons

from object-oriented software engineering. The challenge is to orchestrate the processes involved in the service change across all phases. Two different models of orchestration were shown in the context of a ISO 20.000 certified service management organization. Starting from the "hub-and-spoke" model with separate responsibilities for the design and transition phase we have shown a model tailored to fit small and medium sized organizations by emphasizing end-to-end responsibility.

#### REFERENCES

- [1] *Office of Government Commerce (OGC): ITIL – Service Strategy*, Norwich, 2007
- [2] *Office of Government Commerce (OGC): ITIL – Service Operation*, Norwich, 2007
- [3] *ISO/IEC 20000-1:2005, Information technology – Service management – Part 1: Specification*
- [4] S. Taylor, *ITIL Lifecycle Suite*, The Stationary Office Ltd. 2007
- [5] Bundesfinanzministerium, "Bundeshaushaltsplan 2010", Berlin 2010
- [6] Project Management Institute, "A Guide to the Project Management Body of Knowledge", PMI, 2008
- [7] James A. Fitzsimmons, Mona J. Fitzsimmons, *Service Management: Operations, Strategy, Information Technology*, Austin 2008
- [8] Michael Hammer, James Champy, *Reengineering the corporation*, New York, 2001
- [9] B. Pfitzinger, H. Bley, T. Jestädt, „Service Catalogue and Service Sourcing“ in W. Abramowicz, R. Alt, K.-P. Fähnrich, B. Franczyk, L. A. Maciaszek, "Informatik 2010, Business Process and Service Science – Proceedings of ISSS and BPSC", *Lecture Notes in Informatics*, Volume 177, Leipzig 2010
- [10] Peter Bräutigam (ed.), *IT-Outsourcing*. Berlin, 2004
- [11] Thomas Söbbing (ed.), *Handbuch IT-Outsourcing*. Heidelberg, 2006
- [12] Bernd Pfitzinger, Thomas Jestädt, Torsten Gründer. Sourcing decisions and IT Service Management, in Klaus-Peter Fähnrich, Rainer Alt, Bogdan Franczyk (ed.) Practitioner Track – *International Symposium on Services Science (ISSS'09)*. Leipziger Beiträge zur Informatik: Band XVI. Leipzig, 2009
- [13] W. Edwards Deming, *Out of the Crisis: Quality, Productivity and Competitive Position*, Cambridge University Press, 1986
- [14] Walter A. Shewhart, *Statistical Method: From the Viewpoint of Quality Control*, Lancaster Press, 1939
- [15] K. Beck, W. Cunningham, "A laboratory for teaching object oriented programming", *ACM Sigplan Notices* 24 (10): 1–6
- [16] E. W. Dijkstra, "On the role of scientific thought" in E. W. Dijkstra, "Selected Writings on Computing: A personal perspective", Springer Verlag, 1982
- [17] I. Sommerville: *Software Engineering*, Pearson Studium, München 2007
- [18] F. Hueber. *IT Service Management – Prozessausrichtung und Steuerung am Beispiel eines IT-Dienstleisters*. Diplomarbeit, TU Berlin, 2008
- [19] "Kundenzufriedenheitsumfrage", *internal report*, Toll Collect GmbH, Berlin 2010
- [20] R. K. Yin, *Case Study Research: Design and Methods*, Sage Publications, Los Angeles 2008

# Service Innovation Capability: Proposing a New Framework

Jens Pöppelbuß, Ralf Plattfaut, Kevin Ortbach, Andrea Malsbender,  
Matthias Voigt, Björn Niehaves, Jörg Becker  
European Research Center for Information Systems  
University of Muenster  
Leonardo-Campus 3, 48149 Muenster, Germany  
Email: {jens.poeppelbuss, ralf.plattfaut, kevin.ortbach, andrea.malsbender,  
matthias.voigt, bjoern.niehaves, joerg.becker}@ercis.uni-muenster.de

**Abstract**—Service organizations face the challenge of offering their customers continuously improved or completely new services and, hence, require service innovations to sustain themselves in the market. We interpret the design and implementation of new or enhanced service offerings as a dynamic capability because the service organization is required to sense impulses for innovation, seize meaningful ways for change, and to finally transform its operational capabilities to the desired state. Accordingly, we propose a new framework which structures service innovation capability into the areas of sensing, seizing, and transformation. We further identify and describe the key activities in all of these three areas based on an analysis of existing literature. With this conceptual paper, we contribute to a better understanding of service innovation capability by proposing a novel framework which is grounded in dynamic capability theory. This framework is beneficial to both practice and academia. It offers an overview of service innovation capability areas and activities against which service organizations can critically reflect their service innovation initiatives. As for academia, it stipulates promising directions for future research.

## I. INTRODUCTION

SERVICE organizations require service innovations in order to experience sustained growth, raise the quality and productivity levels of services, respond to changing customer needs and expectations, or stand up to superior competitive service offerings [1]–[4]. They face the principle challenge to “offer the marketplace continuously improved, if not new, services.” [5, p. 275] Service innovations are value propositions not previously available to the customer and result from changes made to the service concept and the delivery process [6]. Researching “the ways in which companies are innovating services” is considered to be a top priority for the science of services [7].

Several tools for service innovation or improvements have been proposed, including, e.g., service blueprints [8], [9], six sigma for service processes [10], and procedure models for service design (e.g., [2], [11]–[13]). Still, the development of new services is considered to be among the least understood topics in the service management and innovation literature [6]. What is lacking is a generic framework that depicts the constituents of service innovation capability [7], [14], [15].

---

This paper was written in the context of the research project KollaPro (promotional reference 01FL10004) funded by the German Federal Ministry of Education and Research.

In this paper, we develop a generic conceptual framework of service innovation capability. Thereby, we respond to the call in the field of service science for general frameworks of service innovation (see, e.g., [15, p. 181]). However, we do not aim at adding another normative process model for service innovation, but draw on dynamic capability theory [16] to describe what actually constitutes service innovation capability in an organization. Service innovation has recently been studied from a dynamic capability perspective [14], [17], [18] and we tie into this school of thought. The framework we propose abstracts from the many normative process models for new service development (NSD) by identifying three key dynamic capability areas and according activities needed for successful service innovation.

The remainder of this paper is structured as follows. We provide the theory background in the next section concentrating on both service innovation and the understanding of service innovation as a dynamic capability. In section 3, we develop our framework which outlines service innovation as a set of abilities clustered in the areas of sensing, seizing, and transformation. In the last section, we draw conclusions, show the implications for research and practice, and provide opportunities for future research.

## II. THEORY BACKGROUND

### A. Service Innovation

A service is the application of competences for the benefit of another [19]. It is “a time-perishable, intangible experience performed for a client who is acting as a coproducer to transform a state of the client.” [1, p. 240] Hence, the customer owns or controls inputs that the service provider is responsible for transforming according to mutual agreement [20].

The following characteristics are frequently mentioned when defining services or distinguishing services from manufacturing. Services are intangible and perishable [2], [21]. Furthermore, the production and consumption of services is not separable, i.e., both happen simultaneously because the customer is involved as a co-producer [2]. Finally, services are heterogeneous as they tend to differ in nature and quality from time to time due to different employees as well as varying customer needs and input [21]. In addition, a distinctive character of services is considered to be their process nature

[9], [21]. However, our understanding of service innovation is not limited by this perception. We agree with Vargo and Lusch that goods and services are not necessarily mutually exclusive [19].

Although early research on NSD frequently borrowed key concepts from the tangible product development literature [12], [15], [22], [23], it is argued that the development of a new service is at least different if not much more complex than the development of a new tangible product [13]–[15], [24]. To give an instance, changes to the service concept [25], i.e., the value proposition offered to the customer, and changes to the service process are mutually interdependent and considerably intertwined [26].

The management of service innovations comprises measures of both incremental (e.g., service enhancements or new constellations of existing service characteristics) and radical change (e.g., introduction of totally new services) [26]–[29]. Service concept and process changes can be driven by different causes, which include arisen or anticipated environmental changes, market opportunities and internal capability evolution [22]. In this article, the term service innovation refers to both the creation of a fundamental new service and the incremental change of existing ones. However, it excludes the customization of service processes during an ongoing service encounter.

The actual process of planning and implementing improved or new services is typically described as a deliberate affair in which organizations follow a formal, methodological procedure with well-defined steps [15], [22]. In this regard, numerous normative procedure models have been suggested to guide service organizations in defining their approaches to service innovation [11], [13], [30]. Such models comprise those activities, tasks, and information flows required of a service organization to conceptualize, develop, evaluate, and prepare services for the market [6], [30]. Many of these models outline a rather sequential process (e.g., [11], [30]) whereas other approaches emphasize the iterative nature of service innovations that involves multiple circles of process design and marketing program testing (e.g., [13], [31]). Generally, it is expected that there is a performance advantage for those service firms that have a formalized innovation process in place [6]. The actual take-up of normative NSD approaches in practice, however, is often considered to be limited [22]. Reports from practice show that “[service] innovation processes often gained a life of their own which broke all planned organisational patterns” [32, p. 445]. In the majority of service organizations, a distinct research and development (R&D) department does not even exist [15]. In essence, the service innovation process tends to

be “interwoven with the capabilities embedded in the processes and routines throughout an organization” [14, p. 491].

Recently, some alternative frameworks have been suggested that aim at addressing the shortcomings of existing service innovation models and the plethora of normative, sequential NSD models in particular. Stevens and Dimitriadis [15], for instance, proposed a NSD model that focuses on organizational learning. Den Hertog et al. [14] draw from dynamic capability theory to identify six dynamic service innovation capabilities. Kindström et al. [18] and Fischer et al. [17] also refer to dynamic capability theory in order to explain how manufacturing companies can extend their solution portfolio through service innovations.

### *B. Service Innovation as a Dynamic Capability*

The Resource-Based View (RBV) of the firm argues that organizations can be seen as collections of distinct resources [33–35]. Following this perception, resources are most commonly framed as “anything which could be thought of as a strength or weakness of a given firm” [33, p. 172], [33]. Moreover, we understand resources as an umbrella term covering both assets and capabilities. In this notion, assets are anything tangible or intangible that can be used by an organization [34]. In contrast, capabilities refer to the ability of an organization to perform a coordinated set of tasks for the purpose of achieving a particular end result: a process [36]. An example could be an organization having access to gold (asset), the machinery needed to mine gold (asset), and the ability to use this machinery in an efficient and effective way (capability). Hence, we understand capabilities as repeatable patterns of action that utilize assets as input [34], [36], [37]. The RBV argues that organizations that have certain assets and capabilities can achieve a competitive advantage as long as these resources fulfill the VRIN conditions, i.e., they must be valuable, rare, imperfectly imitable, and non-substitutable [38].

However, scholars argue that a mere focus on the VRIN attributes is not sufficient for sustained competitive advantage, as this view might under-emphasize market dynamics. A position of competitive advantage that an organizational resource generates today cannot be sustained as changes in the environment may lead to erosion of the resource or replacement by a different resource [39]. A stable resource configuration cannot guarantee long-term competitive advantage as organizations have to adapt this configuration to the market environment [40]. This argument is even stronger in dynamic market environments where there is “rapid change in technology and market forces, and, feedback effects on firms” [16, p. 512]. Hence, organizations need capabilities that enable them to adapt their resource configuration. These capabilities are called dynamic capabilities [16], [40]–[42].

TABLE I.  
SYNOPSIS OF SERVICE INNOVATION FRAMEWORKS

Source	Sensing Activities	Seizing Activities	Transformation Activities
[2]	<ul style="list-style-type: none"> <li>Develop objectives for the service process</li> </ul>	<ul style="list-style-type: none"> <li>Define process to be designed</li> <li>Select design factors (i.e., process type, layout, environment, capacity, quality, IT)</li> </ul>	<ul style="list-style-type: none"> <li>Build and test a prototype of the process</li> <li>Implement the process</li> </ul>
[11]	<ul style="list-style-type: none"> <li>Formulation of new service objectives and strategy</li> </ul>	<ul style="list-style-type: none"> <li>Idea generation</li> <li>Idea screening</li> <li>Concept development</li> <li>Concept testing</li> <li>Business analysis</li> <li>Project authorization</li> </ul>	<ul style="list-style-type: none"> <li>Service design and testing</li> <li>Marketing program design and testing</li> <li>Personnel training</li> <li>Service testing and pilot run</li> <li>Test marketing</li> <li>Full-scale launch</li> <li>Post-launch review</li> </ul>
[13]	<ul style="list-style-type: none"> <li>Feedback and learning</li> <li>Strategic assessment</li> </ul>	<ul style="list-style-type: none"> <li>Concept development</li> <li>System design</li> <li>Component design</li> </ul>	<ul style="list-style-type: none"> <li>Implementation</li> </ul>
[14]	<ul style="list-style-type: none"> <li>Signalling user needs and technological options</li> <li>(Un-)bundling</li> <li>Co-producing and orchestrating</li> </ul>	<ul style="list-style-type: none"> <li>Conceptualising</li> </ul>	<ul style="list-style-type: none"> <li>Co-producing and orchestrating</li> <li>Scaling and stretching</li> <li>Learning and adapting</li> </ul>
[15]	<ul style="list-style-type: none"> <li>Dissonance</li> </ul>	<ul style="list-style-type: none"> <li>Interpretation</li> <li>Test</li> </ul>	<ul style="list-style-type: none"> <li>Implementation/Adoption</li> <li>Routinization/Adaptation</li> </ul>
[30]	<ul style="list-style-type: none"> <li>Develop a business strategy</li> <li>Develop a service strategy</li> </ul>	<ul style="list-style-type: none"> <li>Idea generation</li> <li>Concept development and evaluation</li> <li>Business analysis</li> </ul>	<ul style="list-style-type: none"> <li>Service development and evaluation</li> <li>Market testing</li> </ul>
[31]	<ul style="list-style-type: none"> <li>Audit the existing service system</li> </ul>	<ul style="list-style-type: none"> <li>Assess the new service concept</li> <li>Define the new service system “processes” and extent of change</li> <li>Define the new service system “participants” and extent of change</li> <li>Define the new service system “physical facilities” and extent of change</li> </ul>	<ul style="list-style-type: none"> <li>Assess the impact of integrating service systems</li> <li>Assess the internal capability to handle change</li> </ul>
[43]	<ul style="list-style-type: none"> <li>Problem definition</li> </ul>	<ul style="list-style-type: none"> <li>Problem resolution</li> <li>Solution evaluation</li> </ul>	

Hence, scholars differentiate two types of capabilities from one another: First, the basic functional activities of organizations are called *operational capabilities*. Such capabilities are, e.g., plant layout, distribution logistics, or marketing campaigns [39]. Operational capabilities are needed for the operational functioning of the organizations and relate closely to the original conceptualization of capabilities from the RBV [41]. With relation to the understanding of operational capabilities as the ability to perform a coordinated set of tasks for the purpose of the operational functioning of the organization [36], [41], [44] we understand the provision of services as an operational capability. Second, Teece et al. [16] introduced *dynamic capabilities* as the abilities of an organization to integrate, build, and reconfigure operational capabilities as well as external competences to address rapidly changing environments. Other scholars build on this conceptualization and argue that dynamic capabilities are “a learned and stable pattern of collective activity through which the organization systematically generates and modifies its operat-

ing routines in pursuit of improved effectiveness” [41, p. 340]. Based on these arguments, we will understand dynamic capabilities as the firm’s ability to integrate, build, and reconfigure operational capabilities for the purpose achieving a fit with the market environment. Building upon the understanding of providing services as an operational capability we can thus understand service innovation as a dynamic capability enabling the adaptation of service processes to changing environments.

Each dynamic capability contains sensing, seizing, and transformation activities [16]. In the context of service innovation, sensing refers to the identification of the need to change service operations or opportunities for service innovation, seizing refers to exploring and selecting feasible opportunities for change, and transformation is concerned with the implementation of changed (or new) services in the organization. In line with this perception, we argue that scholarly models for new service development, service engineering, service innovation, or service design can be seen as specific

descriptions of the dynamic capability service innovation. Eventually, all phases of such models can be mapped in one of the three classes of activities (Table 1).

### III. SERVICE INNOVATION FRAMEWORK

We structure service innovation capability into three classes of activities: sensing, seizing, and transformation. Similar to recent research [45], we set out to identify different activities within each of these classes. For this purpose, we consult existing literature on NSD, (service) innovation, and organizational change.

Service innovation literature frequently suggests a differentiation between ‘ideas’ emerging within the early phases of an innovation process (sensing) and ‘concepts’ which are relevant to a later stage of the process (seizing/transformation) [11], [14], [30], [45]–[47]. In contrast to this perception, we see idea generation and concept development as being relevant for both sensing and seizing and thus propose a differentiation based on knowledge types. We refer to Berardi-Coletta et al. [48], who, in their paper on metacognition and problem solving, differentiate problem knowledge from solution knowledge. From a dynamic capability perspective, sensing addresses mostly problem knowledge due to its focus on identifying *that* a service innovation needs to be achieved. In seizing, on the other hand, primarily solution knowledge is of need because the activities in this class focus on identifying *how* this change is put forward within the organization. For the transformation phase we adopt the concept of transformation knowledge as presented by Pohl and Hadorn [49, p. 36] and we thus refer to “technical, social, legal, cultural and other possible means of acting that aim to transform existing practices and introduce desired ones”.

In contrast to many of the available normative models for NSD, we deliberately restrain from prescribing a sequence in which the capability areas and activities should be linked to each other. They are ordered in a way that is intuitively comprehensible and may seem like the common waterfall model [5]. However, we consider the capability areas and activities of our framework to be relevant to all approaches to service innovation, including, for instance, iterative prototyping, as well as parallel or concurrent design [5], [22].

#### A. Sensing

Sensing refers to the management of different sources of information and knowledge that need to be translated into leading problems and unmet service needs before a more focused conceptualization of new service solutions follows in the seizing phase [14]. Literature suggests that service organizations should actively engage in sensing and establish formal processes for this [30]. A general differentiation can be made between sensing external and internal impulses for service innovation. Service innovation is traditionally considered to be triggered by a perceived gap between market requirements and service delivery [22], [24] or the option to translate technology developments into new service propositions [14]. Moreover, competitors may serve as an important source of ideas for new services [24]. The externally stimulated identification of impulses for service innovation focuses on market opportunities [50] and is in

line with the original understanding of sensing capability as put by Teece [16]. In addition to this external perspective, the internally stimulated recognition of needs for change is also important [50]. The internal perspective implies that inefficiencies in current service operations might exhibit the need for change. Usually, such process weaknesses are identified by the service personnel within the organizations thanks to their direct involvement and comprehensive process knowledge. Another internal impulse can be the development of operational capabilities, sometimes even accidentally, for which there is currently no estimable market potential but which could be exploited by introducing new marketing concepts [22]. The inward sensing of service innovation impulses accordingly focuses on the avoidance of internal operational losses (e.g., opportunity costs that would occur if there are no returns from the operational capabilities) [22]. Hence, sensing is not limited to the outward look on customer needs, competitive service offerings, and technological options but also covers the recognition of internal deficiencies in service provision or the exploitation of available operational capabilities. Both the internal and external perspective of sensing mainly include three key activities: 1) scanning, 2) evaluation, and 3) detailing.

*Scanning* has been described as a major driver for innovation [14], [51], [52]. While most publications speak of environmental scanning and thus take a rather external perspective [14], we generalize this capability to comprise the observation of both internal and external impulses. Following Basadur et al. [53, p. 60] we define scanning as a process of “[...] continuously and deliberately discovering and surfacing new and useful problems to be solved”. Den Hertog et al. [14] refer to this activity as an “intelligence function”, which typically resides in marketing, new business development, innovation management or an IT department. Scanning may require the constant dialogue with customers, personnel, and technology providers [14].

*Evaluating* refers to the ability of an organization to quickly screen a particular opportunity or need for service innovation with regard to, for instance, the problem situation as a whole [43], business objectives [13], [54] or general feasibility [55]. This activity involves an initial decision making whether a sensed impulse is worth further detailing which is then followed by the development of possible solutions [15]. Such a decision is typically made long before an official development project is established [15].

The *detailing* activity refers to precisely defining the problem and elaborating side conditions (e.g., technology, legislation and cultural aspects) that need to be taken into account within the development of possible solutions, e.g., by means of new service processes [52] or service concepts [25]. Chai [43] refers to this step as “problem modeling and formation” which includes the definition of an exhaustive set of problem statements and the identification of functions that the new service should fulfill for the customer.

All three sensing activities are not to be seen in strict sequence as they may just as well be executed in an alternate or iterative manner.

#### B. Seizing

As for seizing, we also identify three main activities from literature: 1) solution development, 2) solution evaluation

and selection, and 3) solution detailing. In service organizations, it is typically cross-functional teams that seize the opportunity for service innovation and jointly develop new services through cooperation [12], [15].

The activity *solution development* refers to the service organization’s ability to generate different potential solutions and thus to identify possible paths it could take in redesigning their service offerings according to the previously formulated problem. In service innovation literature this is referred to by notions such as service process design [2], concept development [30], [46], [47], problem resolution [43], building alternative solutions [15], or idea refinement [15]. The development of new solutions does not necessarily mean the creation of completely novel service concepts but may also consist in creatively rearranging existing services into innovative packages [30]. Basadur and Gelade [56] state that the innovation process involves both convergent and divergent thinking. Accordingly, we distinguish the rather divergent task of coming up with options for new service development (or packaging) from the more convergent activities of selecting a particular alternative. From this perspective, solution development can be considered a more divergent activity.

On the other hand, the activity of *solution evaluation and selection* is rather convergent in nature. Here, a company needs established procedures that allow for informed decision making and thus for selecting the most adequate solution for a specific problem. Research found that, e.g., team sizes and participation are important factors that influence this ability [57]. Possible solutions, for instance, can be selected based on a comparison with an “implementation-free description of the [ideal] situation after a problem has been solved” [43, p. 54]. Estimates of the new service concept’s profitability usually also influence the selection to a large degree [46].

Similar to the third sensing activity, a *detailing* ability is also required in seizing. Rough-cut service concepts that are defined on an idea-level for new services are specified in detail. This involves the final determination of the to-be processes, procedures, facilities, information systems, participants, and behaviors that need to be put in place for the new service offering [3], [15], [24]. Here, the development of a comprehensive project plan for the implementation of the se-

lected solution needs to be put forward and implementation project teams as well as control mechanisms have to be set up [15], [54].

As with seizing, the activities of sensing will evolve in an iterative process, stepwise refining and specifying the solution in actionable artifacts.

### C. Transformation

Following Lewin [58], we divide transformation into the three activities of unfreezing, changing, and (re-) freezing. These activities are crucial as they path the way from ideas and concepts to lived new service operations.

*Unfreezing* refers to breaking up existing work structures which is an important aspect when implementing new service processes and interfaces to the customer. Preparations have to be made so that the acceptance of the new work practices can be achieved, e.g., by actively communicating the changes and benefits that result from them [59]. Furthermore, different types of resistance have to be addressed [60]. The *changing* activity refers to the actual implementation of the new service offering. The key question here is how fast, in which steps, at which locations and by what means new work practices are to be adapted within the service organization and the distribution network [15]. Often, prototypes of the new services are developed and tested before a full-scale introduction to the public happens [30].

Finally, the *freezing* activity relates to all tasks necessary to foster internalization of the newly implemented service processes. Here, for instance, continuous motivation [61] and trainings [11] have been identified as possible drivers. For the latter, Bashein et al. [62] state that once a new process is established “the people who will perform the processes need training – not only in the redesigned jobs but in new ways of working together.” Furthermore, the use of information systems (because software works in a defined way) and the use of communication and promotion tools (through which customers adopt standardized expectations) can contribute to freezing a new service [15]. The goal of this activity is to achieve a routinization; this means that the personnel adopts the new service offering and transforms the explicit knowledge about what the new service is like and how to deliver it into tacit knowledge [15].

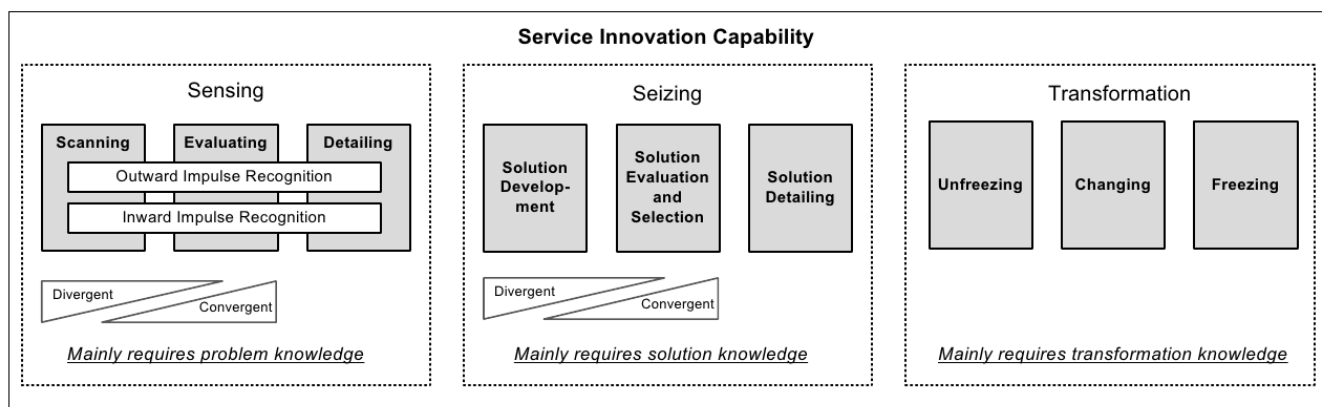


Fig. 1 Service Innovation Capability



#### IV. CONCLUSION

Within this research, we proposed a comprehensive model for understanding service innovation as a dynamic capability of an organization. Based on a literature analysis focusing on service innovation frameworks and procedure models for NSD we were able to show that the majority of existing models comprise activities that can be mapped to the capability areas of sensing, seizing, and transformation. Thus, dynamic capability theory was confirmed as a valid perspective on service innovation. We expect this new framework to offer several benefits for both theory and practice.

From a practical point of view, the conceptualization of service innovation as a dynamic capability helps to better understand the internal antecedents for service innovation within an organization. The presented framework could enable managers to adequately assess and evaluate their service innovation efforts with respect to their individual resource endowments and the market environment. Furthermore, the IT support for service innovation initiatives, which is considered to be lacking [5], could be adapted to fit the needs of particular activities within the framework, or to provide more general support for one of the capability areas of sensing, seizing, or transformation.

As for theory, our research contributes to the field of services science in providing a solid framework for the analysis of service innovation capability. A solid theoretical underpinning is oftentimes missing in related studies. Thus, understanding service innovation as dynamic capability is a valuable perspective, also for a wider array of research in this area, e.g., on how to foster service infusion and growth, create and maintain a service culture, enhance service design, and optimize service networks and value chains [7]. It opens up several possibilities of applying proven models from strategic management literature to the emerging and constantly growing research area of services science [1].

However, these contributions are beset with certain limitations. On the one hand, the presented research has to be classified as purely conceptual and, thus, lacks empirical evidence at this point in time. While the developed framework is grounded in theory, we generally describe possible capability areas of service innovation and explicitly do not give normative recommendations. As a theoretical model, the framework raises the following questions which have to be addressed in future empirical studies: What is the impact of every single capability area on service innovation capability as a whole? How can the success of service innovation as a dynamic capability be adequately measured? What is the impact of the dynamic capability on the business success of service organizations? Moreover, service innovation capability is reflected in collective activities. Hence, the aspect of collaboration shall be subject to further investigation. In this context, concepts from boundary spanning theory could provide a differentiated perspective on collaboration [63].

Hence, future research could (and should) focus on evaluating the specific implementations of the described activities in practice. In this regard, the support through IT and sys-

tematic methodologies that are possibly utilized for service innovation are of particular interest. Furthermore, by comparing new service development efforts and service improvement endeavors within an organization, research could investigate possible differences as regards the relevance of certain capability areas and activities.

#### V. ACKNOWLEDGEMENTS

This paper was written in the context of the research project KollaPro (promotional reference 01FL10004) which is funded by the German Federal Ministry of Education and Research (BMBF).

#### REFERENCES

- [1] J. C. Spohrer and P. P. Maglio, "The Emergence of Service Science: Toward Systematic Service Innovations to Accelerate Co-Creation of Value," *Production and Operations Management*, vol. 17, no. 3, pp. 238-246, May. 2008.
- [2] S. R. Das and C. Canel, "Designing service processes: a design factor based process model," *International Journal of Services Technology and Management*, vol. 7, no. 1, pp. 85-107, 2006.
- [3] F. I. Stuart, "The influence of organizational culture and internal politics on new service design and introduction," *International Journal of Service Industry Management*, vol. 9, no. 5, pp. 469-485, 1998.
- [4] P. R. Magnusson, J. Matthing, and P. Kristensson, "Managing User Involvement in Service Innovation: Experiments with Innovating End Users," *Journal of Service Research*, vol. 6, no. 2, pp. 111-124, Nov. 2003.
- [5] H.-J. Bullinger, K.-P. Fähnrich, and T. Meiren, "Service engineering: methodical development of new service products," *International Journal of Production Economics*, vol. 85, no. 3, pp. 275-287, 2003.
- [6] L. Menor and A. Roth, "New service development competence in retail banking: Construct development and measurement validation," *Journal of Operations Management*, vol. 25, no. 4, pp. 825-846, Jun. 2007.
- [7] A. L. Ostrom et al., "Moving Forward and Making a Difference: Research Priorities for the Science of Service," *Journal of Service Research*, vol. 13, no. 1, pp. 4-36, Jan. 2010.
- [8] G. L. Shostack, "How to design a service," *European Journal of Marketing*, vol. 16, no. 1, pp. 49-63, 1982.
- [9] M. J. Bitner, A. L. Ostrom, and F. N. Morgan, "Service Blueprinting" *California Management Review*, vol. 50, no. 3, pp. 66-95, 2008.
- [10] J. Antony, "Six Sigma for service processes," *Business Process Management Journal*, 2006.
- [11] E. E. Scheuing and E. M. Johnson, "A proposed model for new service development," *Journal of Product Innovation Management*, vol. 6, no. 4, pp. 303-304, Dec. 1989.
- [12] I. Alam and C. Perry, "A customer-oriented new service development process," *Journal of Services Marketing*, vol. 16, no. 6, pp. 515-534, 2002.
- [13] G. Bitran and L. Pedrosa, "A structured product development perspective for service operations," *European Management Journal*, vol. 16, no. 2, pp. 169-189, 1998.
- [14] P. Den Hertog, W. Van Der Aa, and M. W. De Jong, "Capabilities for managing service innovation: towards a conceptual framework," *Journal of Service Management*, vol. 21, no. 4, pp. 490-514, 2010.
- [15] E. Stevens and S. Dimitriadis, "Managing the new service development process: towards a systemic model," *European Journal of Marketing*, vol. 39, no. 1/2, pp. 175-198, 2005.
- [16] D. J. Teece, G. Pisano, and A. Shuen, "Dynamic capabilities and strategic management," *Strategic Management Journal*, vol. 18, no. 7, pp. 509-533, Aug. 1997.
- [17] T. Fischer, H. Gebauer, M. Gregory, G. Ren, and E. Fleisch, "Exploitation or exploration in service business development?: Insights from a dynamic capabilities perspective," *Journal of Service Management*, vol. 21, no. 5, pp. 591-624, 2010.
- [18] D. Kindström, C. Kowalkowski, and E. Sandberg, "A dynamic capabilities approach to service infusion in manufacturing" in

- Proceedings of the QUIS 11 (11th Quality in Services Symposium): Moving Forward with Service Quality*, 2009, pp. 331-340.
- [19] S. L. Vargo and R. F. Lusch, "Evolving to a New Dominant Logic for Marketing," *The Journal of Marketing*, vol. 68, no. 1, pp. 1-17, 2004.
- [20] J. C. Spohrer, P. P. Maglio, J. Bailey, and D. Gruhl, "Steps toward a science of service systems," *Computer*, vol. 40, no. 1, pp. 71-77, 2007.
- [21] H. Katzan, *Service Science: Concepts, Technology, Management*. New York, Bloomington: iUniverse, 2008.
- [22] M. Shulver, "Operational loss and new service design," *International Journal of Service Industry Management*, vol. 16, no. 5, pp. 455-479, 2005.
- [23] L. Menor, M. V. Tatikonda, and S. E. Sampson, "New service development: areas for exploitation and exploration," *Journal of Operations Management*, vol. 20, no. 2, pp. 135-157, Apr. 2002.
- [24] A. Johnes and C. Storey, "New service development: a review of the literature and annotated bibliography," *European Journal of Marketing*, vol. 32, no. 3/4, pp. 184-251, 1998.
- [25] S. Goldstein, "The service concept: the missing link in service design research?," *Journal of Operations Management*, vol. 20, no. 2, pp. 121-134, Apr. 2002.
- [26] H. Droege, D. Hildebrand, and M. A. Heras Forcada, "Innovation in services: present findings, and future pathways," *Journal of Service Management*, vol. 20, no. 2, pp. 131-155, 2009.
- [27] C. Armistead and S. Machin, "Implications of business process management for operations management," *International Journal of Operations & Production Management*, vol. 17, no. 9, pp. 886-898, 1997.
- [28] U. de Brentani, "Innovative versus incremental new business services: different keys for achieving success," *Journal of Product Innovation Management*, 2001.
- [29] A. Oke, "Innovation types and innovation management practices in service companies," *International Journal of Operations & Production Management*, vol. 27, no. 6, pp. 564-587, 2007.
- [30] M. R. Bowers, "Developing New Services: Improving the Process Makes it Better," *Journal of Services Marketing*, vol. 3, no. 1, pp. 15-20, 1989.
- [31] S. Tax, "Designing and implementing new services: The challenges of integrating service systems," *Journal of Retailing*, vol. 73, no. 1, pp. 105-134, 1997.
- [32] J. Sundbo, "Management of Innovation in Services," *The Service Industries Journal*, vol. 17, no. 3, pp. 432-455, Jul. 1997.
- [33] B. Wernerfelt, "A resource-based view of the firm," *Strategic Management Journal*, vol. 5, no. 2, pp. 171-180, 1984.
- [34] M. Wade and J. Hulland, "Review: The Resource-Based View and Information Systems Research: Review, Extension and Suggestions for Future Research," *MIS Quarterly*, vol. 28, no. 1, pp. 107-142, 2004.
- [35] E. P. Penrose, "The Theory of the Growth of the Firm." John Wiley & Sons, 1959.
- [36] C. E. Helfat and M. A. Peteraf, "The dynamic resource-based view: Capability lifecycles," *Strategic Management Journal*, vol. 24, no. 10, pp. 997-1010, 2003.
- [37] R. Amit and P. Schoemaker, "Strategic assets and organizational rent," *Strategic Management Journal*, vol. 14, no. 1, pp. 33-46, 1993.
- [38] J. B. Barney, "Firm Resources and Sustained Competitive Advantage," *Journal of Management*, vol. 17, no. 1, pp. 99-120, 1991.
- [39] D. J. Collis, "Research Note: How Valuable are Organizational Capabilities?," *Strategic Management Journal*, vol. 15, no. 1, pp. 143-152, 1994.
- [40] K. M. Eisenhardt and J. A. Martin, "Dynamic capabilities: what are they?," *Strategic Management Journal*, vol. 21, no. 10-11, pp. 1105-1121, 2000.
- [41] M. Zollo and S. G. Winter, "Deliberate learning and the evolution of dynamic capabilities," *Organization Science*, vol. 13, no. 3, pp. 339-351, 2002.
- [42] H. Koch, "Developing dynamic capabilities in electronic marketplaces: A cross-case study," *The Journal of Strategic Information Systems*, vol. 19, no. 1, pp. 28-38, 2010.
- [43] K.-H. Chai, "A TRIZ-Based Method for New Service Design," *Journal of Service Research*, vol. 8, no. 1, pp. 48-66, Aug. 2005.
- [44] S. G. Winter, "Understanding dynamic capabilities," *Strategic Management Journal*, vol. 24, no. 10, pp. 991-995, 2003.
- [45] S. Balaji and C. Ranganathan, "Exploring the key capabilities for offshore IS sourcing," in *Proceedings of the International Conference on Information Systems (ICIS 2006)*, 2006, pp. 543-552.
- [46] R. E. Reidenbach and D. L. Moak, "Exploring Retail Bank Performance and New Product Development: A Profile of Industry Practices," *Journal of Product Innovation Management*, vol. 3, no. 3, pp. 187-194, Sep. 1986.
- [47] D. Cowell, "New service development," *Journal of Marketing Management*, vol. 3, no. 3, pp. 296-312, 1988.
- [48] B. Berardi-Coletta, L. S. Buyer, R. L. Dominowski, and E. R. Rellinger, "Metacognition and problem solving: A process-oriented approach," *Journal of Experimental Psychology: Learning, Memory, and Cognition*, vol. 21, no. 1, pp. 205-223, 1995.
- [49] C. Pohl and G. H. Hadorn, *Principles for designing transdisciplinary research*. München: oekom, 2007.
- [50] M. Bhave, "A process model of entrepreneurial venture creation," *Journal of Business Venturing*, vol. 9, no. 3, pp. 223-242, May. 1994.
- [51] I. DeToro and T. McCabe, "How to stay flexible and elude fads," *Quality Progress*, vol. 30, no. 3, pp. 55-60, 1997.
- [52] M. Zairi and D. Sinclair, "Business process re-engineering and process management: A survey of current practice and future trends in integrated management," *Business Process Management Journal*, vol. 1, no. 1, pp. 8-30, 1995.
- [53] M. Basadur, P. Pringle, G. Speranzini, and M. Bacot, "Collaborative Problem Solving Through Creativity in Problem Definition: Expanding the Pie," *Creativity and Innovation Management*, no. March, pp. 54-76, 2000.
- [54] B. Bernstein and P. Singh, "An integrated innovation process model based on practices of Australian biotechnology firms," *Technovation*, vol. 26, no. 5-6, pp. 561-572, May. 2006.
- [55] S. Majaro, *The creative gap: managing ideas for profit*. Longman Trade/Caroline House, 1988.
- [56] M. Basadur and G. Gelade, "The Role of Knowledge Management in the Innovation Process," *Creativity and Innovation Management*, vol. 15, no. 1, pp. 45-62, Mar. 2006.
- [57] C. K. W. De Dreu and M. A. West, "Minority dissent and team innovation: The importance of participation in decision making," *Journal of Applied Psychology*, vol. 86, no. 6, p. 1191, 2001.
- [58] K. Lewin and D. Cartwright, *Field theory in social science*. New York: Harper & Brothers, 1951.
- [59] J. P. Kotter, "Leading Change why Transformation Efforts Fail," *Harvard Business Review*, no. Jan, pp. 92-107, 2007.
- [60] J. O'Toole, *Leading Change*. San Francisco: Jossey-Bass, 1996.
- [61] A. Mento, R. Jones, and W. Dirndorfer, "A change management process: Grounded in both theory and practice," *Journal of Change Management*, vol. 3, no. 1, pp. 45-59, Mar. 2002.
- [62] B. J. Bashein, M. L. Markus, and P. Riley, "Preconditions for BPR success and how to prevent failures," *Information Systems Management*, vol. 11, no. 2, pp. 7-13, 1994.
- [63] N. Levina and E. Vaast, "The Emergence of Boundary Spanning Competence in Practice: Implications for Implementation and Use of Information Systems," *MIS Quarterly*, vol. 29, no. 2, pp. 335-363, 2005.



## A Framework for Comparing Cloud-Environments

Rainer Schmidt  
 Aalen University  
 Anton-Huber-Str. 25  
 73430 Aalen, Germany  
 Email: Rainer.Schmidt@htw-aalen.de

**Abstract**—Cloud-services are more and more part of so-called cloud-environments. Cloud-environments provide a set of cloud-services and resources. Management interactions allow configuring services and resources to individual requirements. Enterprises selecting a cloud environment have not only to consider the functionality of the cloud-services, but also the management interactions offered by the cloud-environment. Therefore, a framework for the comparison of cloud-environments is introduced. It uses meta-services to specify both the functional and non-functional properties of management interactions.

### I. INTRODUCTION

CLOUD-computing [1], [2] integrates concepts and ideas such as service-oriented computing [3] and information systems outsourcing [4] [5] in order to realize utility computing [6]. Cloud-Services are services provided by services providers using the cloud-computing approach [1] [2]. Cloud-Services may be software-, platform or infrastructure services (SaaS/PaaS/IaaS) [1] [2].

In the beginning, cloud-services have been offered in isolation. Nowadays cloud-services are more and more offered as part of so-called cloud-environments. Prominent examples are Office365 [7] and Google Apps [8]. They provide a set of cloud-services such as text-processing, email, spreadsheet calculation and provide storage for text and spreadsheets. Cloud-environments provide management interactions in order to adapt cloud-environments to individual requirements. There are three basic types of management interactions: integrate/disintegrate, configure, and import/export as shown in Fig. 1. The first kind of interaction is the integration of services (1) and resources (3) into the cloud-environment. To finish the usage of a source or resource, there is also a disintegrate interaction. E.g. both Office 365 and Google Apps allow to integrate the active directory services [9]. Integrated services and resources can be used for service provisioning but remain outside the sphere of control of the cloud-environment.

Resources, however, can also be moved into the cloud-environment. To do so, an import interaction is provided (4). To avoid a vendor-lock-in, it must be possible to export resources. Both Office 365 and Google Apps allow to import

files up into the cloud-environment or to export them. The configure interactions allow to adapt services (2) and resources (5) to individual requirements. E.g. it is possible to configure the email services by changing the reply-address, defining an out-of-office message etc.

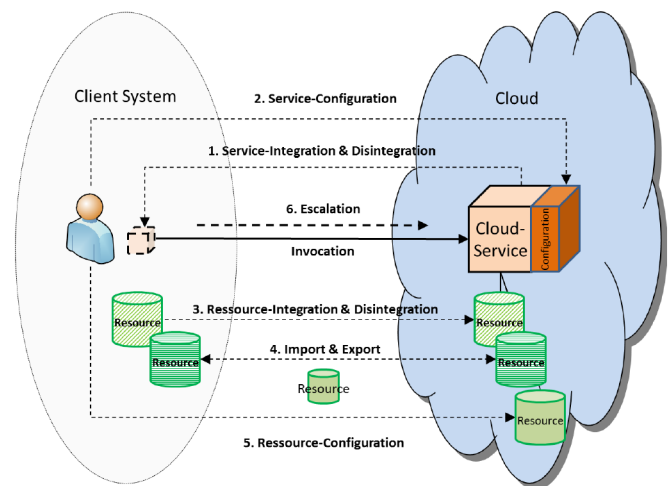


Fig. 1 Management interactions with Cloud-environment

There are a number of approaches to describe cloud-services in the same way as other types of services, e.g. the Unified Service Description Language [10]. However, these approaches do not consider management interactions. Thus, the description of cloud-environments remains incomplete. Therefore, in this paper a framework for the description and comparison of cloud-environments is introduced. The framework uses meta-services [11] [12] to formalize the management interactions.

The paper proceeds as follows. First a formalized description of cloud-environments it is created. Then meta-services are used to represent management interactions. Using them a comparison framework for cloud-environments is created. It is then used to compare two popular cloud-environments, Office365 [7] and Google Apps [8]. Finally, a conclusion and outlook is given.

## II. FORMALIZING CLOUD-ENVIRONMENTS

To create a solid basis for the framework, cloud-environments have to be formalized. Cloud-environments contain services and resources which can be configured to individual requirements. Thus, a cloud-environment can be regarded as a configuration of services and resources, as shown in Fig. 2. A configuration consists of a set of configuration items representing individual services or resources. Configuration items can be associated in different ways. By introducing the entities configuration and configuration item, it is possible to separate abstract specifications from the real services and resources. Thus, a layer of indirection is created that allows assigning real services and resources.

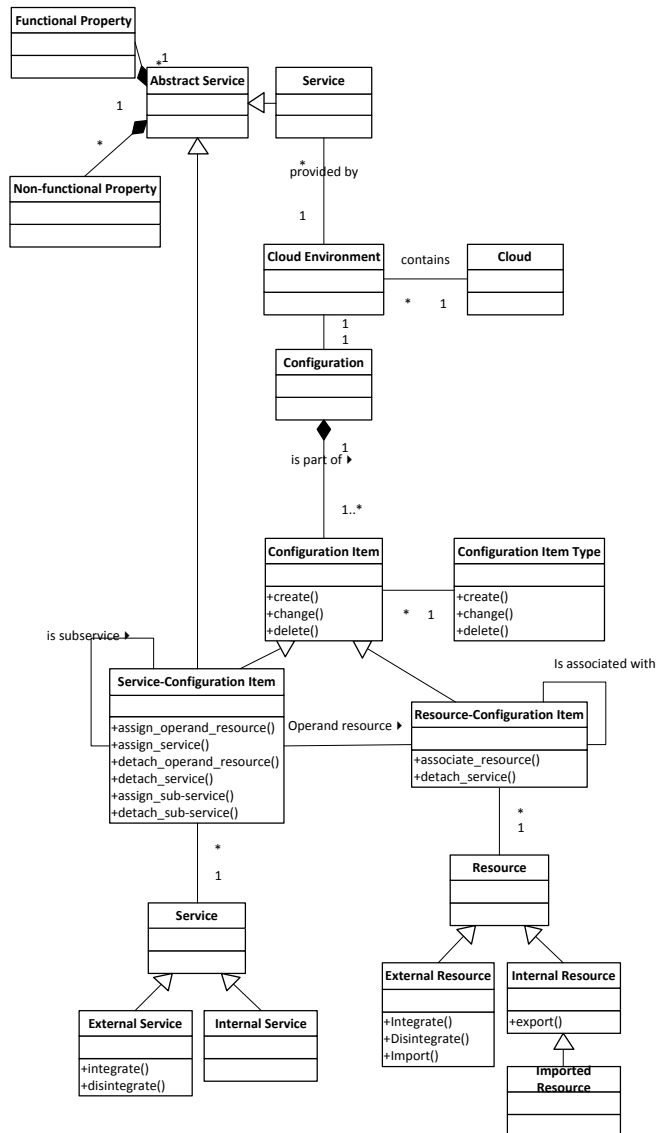


Fig. 2 Formalized Cloud-environment

There are two types of configuration items, service and resource configuration items. Service configuration items specify services to be used for service provisioning. Resource configuration items specify resources to be used

for service provisioning. Both service and resource configuration items may be nested. The configuration items can be connected in different ways to represent the relationships of services and resources. Services may use sub-ordinated services in order to provide a service. Service may act upon resources. Resources may be associated with one another.

Both services and resources are differentiated into internal and external ones. External services and resources may be used by the cloud environment for service provisioning. They are outside the sphere of control of the cloud environment. That means the service or resource cannot be influenced in its lifecycle status by the cloud-environment. On the contrary, internal services and resources can be influenced by the cloud-environment in their lifecycle status.

It may be necessary to move a resource into the sphere of control of the cloud-environment in order to manage it more efficiently, e.g. change it in the context of a transaction [13]. To do so, an external resource is imported and becomes an imported resource. Both imported resources and internal resources in general may be outside the sphere of control of the cloud-environment. Therefore not only an import but also an export operation is provided.

The entities of the formalized cloud-environment and the operations defined on them can now be used to represent the management interactions of cloud-environments. Especially differences in the power of management interactions can be expressed exactly. Integration interactions may offer three levels of variability. First, it may be possible to integrate every type of service or resource. This can be expressed by the creation of new entities of configuration item types. Then, concrete services or resources may be assigned. Second, only the integration of services and resources of predefined types may be possible. This is expressed by the creation of configuration items. Third, also the cardinality may be restricted. In this case, it is only possible to assign services and resources to already existing configuration items. Also, the possibilities to change the assignment between configuration items and services and resources can express important properties of the cloud-environment. The capability to change the assignment of services and resources expresses the capability to replace cloud-services. If this capability is missing, the set of cloud-services may be extended, but the assignment of the already existing services may not be changed. The disintegration of a service or resource can be described as deletion of a configuration item. The configuration of services and resources is denoted by the modification of configurations items and their relationships.

## III. COMPARING OFFICE 365 AND GOOGLE APPS USING META-SERVICES

At first sight, it seems to be sufficient to define the functionality of the interactions only. However, to an enterprise it may be a very decisive point, how fast and

reliable management interactions are performed. Therefore, to represent both the functionality and the non-functional properties management interactions, meta-services [11] [12] are used. Meta-Services are services acting upon services [11]. Thus, the interactions are defined as services acting upon the cloud-services and their resources.

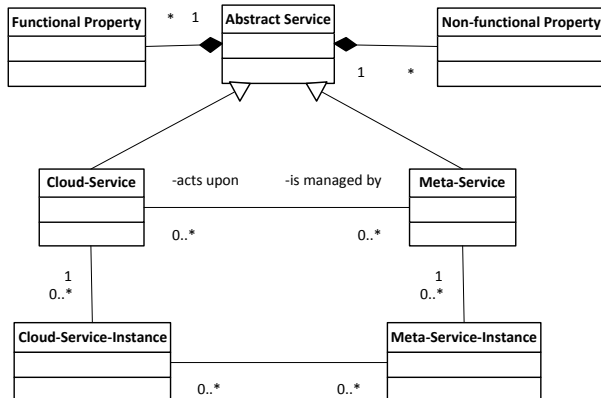


Fig. 3 Meta-Services

Using meta-services also has the advantage of creating a homogeneous approach for describing both services and management interactions. The meta-services identified create a framework for the comparison of cloud-environments. To evaluate it, two popular cloud-environments shall be compared: Office365 [7] and Google Apps [8]. The results are summarized in tables containing the functionality of the meta-services and their non-functional properties, as far as available. Non-functional properties are written in italics.

#### A. Integration-oriented Meta-Services

Office 365 does not allow enlarging the set of services and resources. This incapability can be described as the lack of a meta-service to create service and resource configuration item types.

However, Office365 allows the integration of predefined services such as the Active Directory Service [9]. A configuration item represents the external active directory service. Also, the integration of resources is restricted: it allows integrating Azure-based resources, only. All integration meta-services are immediately effective.

TABLE I.  
COMPARING THE INTEGRATION ORIENTED META-SERVICES

Cloud-environment	Integration		Disintegration	
	Service	Resource	Service	Resource
Office 365	Active Directory	Azure resources	Yes	Yes
	<i>Immediately</i>	<i>Immediately</i>	<i>Immediately</i>	<i>Immediately</i>
Google Apps	Gadgets	File resources using an adapter	Yes	Yes
	<i>Immediately</i>	<i>Several hours</i>	<i>Immediately</i>	<i>Several hours</i>

Google Sites allow integrating external services into web pages using gadgets. Again, only an existing configure item can be used to represent a predefined type of service, but no

integration of arbitrary types of services can take place. The integration takes place immediately. File resources can be integrated using an adapter. They allow to access external services. As same as Office 365, Google Apps allows integrating an external Active Directory Service [9]. Integration may take several ours due to the configuration effort. Office 365 supports the disintegration of integrated services and resources; the same applies to Google Apps. The non-functional-properties are the same as for the integration meta-services.

#### B. Import/Export Meta-Services

Office 365 supports the import and export of documents either manually or in batch mode. User accounts can be imported. Google Apps supports the import and export of documents both manually and automatically. There is also the possibility to import user accounts.

#### C. Configuration Meta-Services

Both Office 365 and Google Apps allow changing the configuration of the services and resources already available. Thus, both provide meta-services for changing configuration items of services and resources. Changes become effective immediately. Office 365 does not allow deleting services or resources from the cloud-environment. Google Apps allows deleting services that have been added using the Google apps Marketplace. This may need one day.

## IV. RELATED WORK

Up to now, there are only some ad-hoc approaches for identifying management interactions in the context of services. In [14], a first approach for capturing interactions in the context of cloud-services is developed. However, this approach only considers interactions for defining services and does not support other kinds of interactions. The approach in [15] provides means to model conversations within complex web-services. In particular it allows specifying valid operations in every status of the web service. However, it handles the operating phase only. The WSDM-standard created by OASIS (Web Services Distributed Management) [16] is based on the OASIS Web Services Resource Framework (WSRF) [17]. The WSDM-standard manages the operational status of web services, but not the interactions outside the operation phase of the web service. One of the earliest ones is the Web Service Description Language [18] which has been augmented by the following approaches: The Web Service Modelling Ontology [19] provides the semantic description of web services in order to facilitate the discovery, combination and invocation of web services. It is limited to the static properties of web services and does not take into account their life-cycle. OWL-S [20] is an ontology describing web services by a profile, grounding and a process. However, it does only cover the operational phase of the service. The grounding defines how to interact with the service. To do so, a mapping

between the process of the process model and concrete operations in WSDL is defined.

Cloud-environments relate to the term service system introduced by Maglio et al. [21]. They define a service system “as an open system capable of improving the state of another system through sharing or applying its resources and capable of improving its own state by acquiring external resources”. Thus, service systems parallel with cloud-environments by the ability to acquire resources. However, there is no formalization of service systems so far.

Nevertheless, the literature on service systems confirms the separation between abstract configurations and concrete services and resources. In [22] a service system is defined as “a value co-production configuration of people, technology, other internal and external service systems, and shared resources (such as language, processes, metrics, prices, policies, and laws)”.

## V. CONCLUSION

Cloud-services are more and more embedded into cloud-environments. Cloud-environments contain a set of cloud-services such as text processing, email. Cloud-environments also provide interactions to configure services, integrate resources etc. Therefore, the existing means for describing services had to be extended by means for capturing the interactions provided by the cloud-environments. To accomplish this, a description of cloud-environments have been analysed and formalized. The conceptualisation of cloud-environments has been used to define meta-services representing the interactions provided by cloud-environments. These meta-services create a framework, that allows to compare different cloud-environments not only by the functionality of the cloud-services provided, but also by the capabilities for configuration, integration, import and export.

The meta-services identified are of particular importance because their availability within a cloud-environment strongly influences the value provided by a cloud-environment offered. To a customer not only the functionality provided by the cloud-environment is important, but also possibilities to tailor the cloud-environment to individual needs or to adapt it to changed requirements. Furthermore, by describing the interactions as meta-services, not only the possibility to configure, integrate resources etc. is described, but also the time necessary to perform such interactions is defined, as same as a number of further quality attributes. By this means the definition of the meta-services of a cloud-environment becomes a metric for the agility of the cloud-environment.

Further work will have to give more details on the framework. In particular, not only the meta-services itself but also their results should be described. E.g. the integration of resources may lead to different levels of integration. There may be a read-only access, a write access and there may be an transaction-protected integration.

## ACKNOWLEDGMENTS

I would like to thank Anatol Dück, Christoph Fritz, Swen Fuchs, Albert Hauler, Tim Krause, Max Wassiljew for fruitful discussions.

## REFERENCES

- [1] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” 10-Jul-2009. [Online]. Available: <http://csrc.nist.gov/groups/SNS/cloud-computing/>. [Accessed: 06-Jan-2011].
- [2] M. Armbrust et al., “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, pp. 50-58, 2010.
- [3] M. P. Papazoglou, “Service-oriented computing: Concepts, characteristics and directions,” 2003.
- [4] M. C. Lacity and R. Hirschheim, *Information systems outsourcing*. Wiley New York, 1993.
- [5] J. Dibbern, T. Goles, R. Hirschheim, and B. Jayatilaka, “Information systems outsourcing: a survey and analysis of the literature,” *SIGMIS Database*, vol. 35, no. 4, pp. 6-102, 2004.
- [6] M. Armbrust et al., “Above the clouds: A Berkeley view of cloud computing,” *University of California, Berkeley, Tech. Rep.*, 2009.
- [7] “Office 365 for Small Business - Hosted Productivity Software.” [Online]. Available: <http://www.microsoft.com/en-us/office365/online-software.aspx>. [Accessed: 30-May-2011].
- [8] “Welcome to Google Apps.” [Online]. Available: <http://www.google.com/apps/>. [Accessed: 30-May-2011].
- [9] B. Desmond, J. Richards, R. Allen, and A. G. Lowe-Norris, *Active Directory*. O’Reilly Media, Inc., 2008.
- [10] S. Kona, A. Bansal, L. Simon, A. Mallya, G. Gupta, and T. D. Hite, “USDL: A Service-Semantics Description Language for Automatic Service Discovery and Composition 1,” *International Journal of Web Services Research*, vol. 6, no. 1, pp. 20-48, 2009.
- [11] R. Schmidt, “Meta-Services as Third Dimension of Service-Oriented Enterprise Architecture,” in *2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops*, 2010, pp. 157-164.
- [12] R. Schmidt, “Perspectives for Moving Business Processes into the Cloud,” in *Enterprise, Business-Process and Information Systems Modeling*, 2010, pp. 49-61.
- [13] P. A. Bernstein and E. Newcomer, *Principles of transaction processing*. Morgan Kaufmann, 2009.
- [14] L. Wang, L. F. Pires, A. Wombacher, M. J. van Sinderen, and C. Chi, “Stakeholder Interactions to Support Service Creation in Cloud Computing,” in *2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops*, Vit ria, Brazil, 2010, pp. 173-176.
- [15] B. Benatallah, F. Casati, F. Toumani, and R. Hamadi, “Conceptual modeling of web service conversations,” in *Advanced Information Systems Engineering*, 2010, pp. 1031-1031.
- [16] V. Bullard, B. Murray, and K. Wilson, “An Introduction to WSDM.” OASIS, 24-Feb-2006.
- [17] OASIS, *Web Services Resource Framework(WSRF) – Primer v1.2*. pp. Committee Draft 02 - 23 May 2006.
- [18] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, *Web services description language (WSDL) 1.1*. Citeseer, 2001.
- [19] D. Roman et al., “Web service modeling ontology,” *Applied Ontology*, vol. 1, no. 1, pp. 77-106, 2005.
- [20] D. Martin et al., “Bringing semantics to web services with owl-s,” *World Wide Web*, vol. 10, no. 3, pp. 243-277, 2007.
- [21] P. Maglio, S. Vargo, N. Caswell, and J. Spohrer, “The service system is the basic abstraction of service science,” *Information Systems and E-Business Management*, vol. 7, no. 4, pp. 395-406, 2009.
- [22] J. Spohrer, P. P. Maglio, J. Bailey, and D. Gruhl, “Steps Toward a Science of Service Systems,” *COMPUTER*, no. January, pp. 71-77, 2007.



# Joint Agent-oriented Workshops in Synergy

**J**OINT Agent-oriented Workshops in Synergy is a coalition of agent-oriented workshops that come together to build upon synergies of interests and aim at bringing together researchers from the agent community for lively discussions and exchange of ideas. Workshops that constitute JAWS are:

- Workshop on Agent Based Computing: from Model to Implementation VIII (ABC:MI)
- 5th International Workshop on Multi-Agent Systems and Simulation (MAS&S)
- Service-Oriented Computing: Agents, Semantics, and Engineering (SOCASE)

JAWS is endorsed by the EU COST Action IC0801 Agreement Technologies.

## STEERING COMMITTEE

**Giancarlo Fortino**, Università della Calabria (Italy),  
g.fortino@unical.it

**Maria Ganzha**, University of Gdansk and IBS PAN,  
Poland

**Marcin Paprzycki**, WSM and IBS PAN, Poland

**Rainer Unland**, University of Duisburg-Essen, Germany  
rainer.unland@icb.uni-due.de



# Workshop on Agent Based Computing: from Model to Implementation—VIII

**T**HE FIELD of agent technology is rapidly maturing. One of key factors that influence this process is the gathered body of knowledge that allows in-depth reflection on the very nature of designing and implementing agent systems. As a result, we know better how to design and implement them. We also understand the most important issues to be addressed in the process. Therefore, on the top-most level we see progress in development of methodologies for design of agent-based systems. Furthermore, these methodologies are usually supported by tools that allow not only top level conceptualization but guide the process towards implementation (e.g. by generating at least some code). Next, we can see that new languages for agent based systems are created, e.g. AML or API Calculus. Separately, tools/platforms/environments that can be used for design and implementation of agent systems have been through a number of releases, eliminating problems and adding new, important features. Resulting products are becoming truly robust and flexible. Furthermore, open source products (e.g. JADE) are surrounded by user communities, which often generate powerful add-on components, further increasing value of existing solutions.

## TOPICS

During the Workshop we are primarily interested in all aspects of the process that leads from the model of the problem domain to the actual agent-based solution. These aspects will cover both principled approaches and established practices of software engineering aimed at producing high quality software. In this context, research into the application of agent-based solutions to key challenges faced by software engineering (e.g. reduction of costs and delivery times, coping with a larger diversity of problems) will be of primary importance.

ABC:MI Workshop welcomes submissions of original papers concerning all aspects of software agent engineering.

Topics include but are not limited to:

- Methodologies for design of agent systems
- Multi-agent systems product lines
- Modeling agent systems
- Agent architectures
- Agent-based simulations
- Simulating and verifying agent systems
- Agent benchmarking and performance measurement
- Agent communication, coordination and cooperation
- Agent languages
- Agent learning and planning
- Agent mobility
- Agent modeling, calculi, and logic
- Agent security
- Agents and Service Oriented Computing
- Agents in the Semantic Web
- Applications and Experiences

## PROGRAM COMMITTEE

- Thomas Agotnes**, University of Bergen, Norway  
**Sattar Al-Maliky**, University of Babilon, Iraq  
**Makoto Amamiya**, Osaka Institute of Technology and Kyushu University, Japan  
**Lars Braubach**, University of Hamburg, Germany  
**Frances Brazier**, TU Delft, Netherlands  
**Paolo Bresciani**, European Commission – DG Information Society and Media, Belgium  
**Zoran Budimac**, University of Novi Sad, Serbia  
**Giacomo Cabri**, University of Modena and Reggio Emilia, Italy  
**Bengt Carlsson**, Blekinge Institute of Technology, Sweden  
**Radovan Cervenka**, Whitestein Technologies, Slovakia  
**Krzysztof Cetnarowicz**, AGH University of Science and Technology, Poland  
**Ireneusz Czarnowski**, Gdynia Maritime University, Poland  
**Hoa Khanh Dam**, University of Wollongong, Australia  
**Beniamino Di Martino**, Seconda Università di Napoli, Italy  
**Barbara Dunin-Kępicz**, Warsaw University, Poland  
**George Eleftherakis**, CITY International Faculty of the University of Sheffield, Greece  
**Amal El Fallah Seghrouchni**, LIP6, France  
**Mohammad Essaaidi**, Abdelmalek Essaadi University, Morocco  
**Adina Magda Florea**, University “Politehnica” of Bucharest, Romania  
**Giancarlo Fortino**, University of Calabria, Italy  
**Ana Garcia-Fornes**, Universidad Politècnica de Valencia, Spain  
**Vladimir Gorodetsky**, St.Petersburg Institute for Informatics and Automation of RAS, Russia  
**Jean Gourd**, Louisiana Tech University, USA  
**Dominic Greenwood**, Whitestein Technologies, Switzerland  
**Maurice Grinberg**, New Bulgarian University, Bulgaria  
**Mike Hinchey**, Lero-the Irish Software Engineering Research Centre, Ireland  
**Mirjana Ivanovic**, University of Novi Sad, Serbia  
**Wojtek Jamroga**, University of Luxembourg, Luxembourg  
**Piotr Jędrzejowicz**, Gdynia Maritime University, Poland  
**Gordan Jezic**, University of Zagreb, Croatia  
**Matthias Klusch**, German Research Center for Artificial Intelligence, Germany  
**Igor Kotenko**, SPIRAS, Russian Federation  
**Zofia Kruczkiewicz**, Technical University of Wrocław, Poland  
**Michal Laclavik**, Institute of Informatics, Slovak Academy of Sciences, Slovakia

**Jiming Liu**, Hong Kong Baptist University, China  
**Vincenzo Loia**, University of Salerno, Italy  
**Michele Loreti**, University of Florence, Italy  
**Giuseppe Mangioni**, University of Catania, Italy  
**Felipe Meneguzzi**, Carnegie Mellon University, USA  
**Viorel Negru**, West University of Timisoara, Romania  
**Mariusz Nowostawski**, University of Otago, New Zealand  
**Ngoc Thanh Nguyen**, Technical University of Wroclaw, Poland  
**Andrea Omicini**, Alma Mater Studiorum, Università di Bologna, Italy  
**Nir Oren**, University of Aberdeen, United Kingdom  
**Sascha Ossowski**, University Rey Juan Carlos, Spain  
**Joaquin Peña**, University of Seville, Spain  
**Laurent Perrussel**, University of Toulouse 1 Capitole, France  
**Agostino Poggi**, DII – University of Parma, Italy  
**Alexander Pokahr**, University of Hamburg, Germany  
**Florin Pop**, Technical University of Bucharest, Romania  
**Thomas Potok**, Oak Ridge National Lab, USA  
**Bhanu Prasad**, Florida A&M University, USA  
**Jarogniew Rykowski**, Poznan University of Economics, Poland  
**Sabrina Senatore**, University of Salerno, Italy  
**Marija Slavkovic**, University of Luxembourg, Luxembourg

**Stanislaw Stanek**, University of Economics in Katowice, Poland  
**Niranjan Suri**, IHMC, USA  
**Kuldar Taveter**, Department of Informatics, Tallinn University of Technology, Estonia  
**Adriaan ter Mors**, TU Delft, Netherlands  
**Paolo Torrioni**, University of Bologna, Italy  
**Nicolas Troquard**, University of Essex, UK  
**Walter Truszkowski**, NASA, USA  
**Rainer Unland**, University of Duisburg-Essen, ICB, Germany  
**Andrzej Uszok**, IHMC, USA  
**Laurentiu Vasiliu**, DERI, Ireland  
**Salvatore Venticinque**, Second University of Naples, Italy  
**Paulus Wahjudi**, Marshall University, USA  
**Antoine Zimmermann**, Digital Enterprise Research Institute, Ireland

#### ORGANIZING COMMITTEE

**Costin Badica**, University of Craiova, Romania  
**Maria Ganzha**, University of Gdansk and IBS PAN, Poland  
**Marcin Paprzycki** (Chair), IBS PAN and WSM, Poland  
**Shahram Rahimi**, Southern Illinois University, USA

# A methodology for developing component-based agent systems focusing on component quality

George Eleftherakis  
Petros Kefalas

Computer Science Department  
CITY College, Thessaloniki, Greece  
International Faculty of the University of Sheffield  
Email: {eleftherakis, kefalas}@city.academic.gr

Evangelos Kehris

Department of Business Administration,  
Technological Education Institute of Serres,  
Greece  
Email: kehris@teiser.gr

**Abstract**—Formal development of agent systems with inherent high complexity is not a trivial task, especially if a formal method used is not accompanied by an appropriate methodology. X-machines is a formal method that resembles Finite State Machines but has two important extensions, namely internal memory structure and functions. In this paper, we present a disciplined methodology for developing agent systems using communicating X-machine agents and we demonstrate its applicability through an example. In practice, the development of a communicating system model can be based on a number of well-defined distinct steps, i.e. development of types of X-machine models, agents as instances of those types, communication between agents, and testing as well as model checking each of these agents individually. To each of the steps a set of appropriate tools is employed. Therefore the proposed methodology utilises a priori techniques to avoid any flaws in the early stages of the development together with a posteriori techniques to discover any undiscovered flaws in later stages. This way it makes the best use of the development effort to achieve highest confidence in the quality of the developed agents. We use this methodology for modelling naturally distributed systems, such as multi-agent systems. We use a generalized example in order to demonstrate the methodology and explain in detail how each activity is carried out. We briefly present the theory behind communicating X-machine agents and then we describe in detail the practical issues related using the same example throughout.

## I. INTRODUCTION

AGENT oriented software engineering aims to manage the inherent complexity of distributed systems [1]. The developing process should be accompanied by methodologies and tools that can lead towards the implementation of “correct” systems: system models that match the requirements and satisfy any necessary properties in order to meet the design objectives, and system implementation that passes all tests constructed using a complete functional test generation method. All the above criteria are closely related to three stages of system development, namely modelling, verification and testing. It is argued that the use of formal methods can achieve this goal to some extent [2].

Formal modelling has centred on the use of models of data types, either functional or relational. Although these have led to some considerable advances in software design, they lack the ability to express the dynamics of the system. Some other methods, such as Finite State Machines (FSM)

or Petri Nets capture the dynamics, but fail to describe the system completely, since there is little or no reference at all to the internal data and how this data is affected by system operations. Finally, methods like Statecharts, capture the requirements of dynamic behaviour and modelling of data but are rather informal with respect to clarity and semantics. So far, little attention has been paid in formal methods that could facilitate all crucial stages of “correct” system development, modelling, verification and testing.

In this paper we use a formal method, namely X-machines and its extension Communicating X-machines, which closely suits the needs of agent-based development [3], while at the same time being practical. We present a disciplined methodology for the incremental development of simple reactive agent-based systems and we present in a formal way all the required extensions of the model which will optimize towards agent systems. The proposed methodology utilises a priori techniques (formal modelling and verification) to avoid any flaws in the early stages of the development together with a posteriori techniques (a black box formal testing strategy) to discover any undiscovered flaws in later stages. This way it makes the best use of the development effort to achieve highest confidence in the quality of the developed agents, allowing safer composition of trusted, reusable agents. The methodology is achieving all these using communicating X-machine agents as building blocks. X-machines is a formal method that enhances the class of FSM with two important characteristics, namely memory and functions. X-machine model types are defined by an input stream, an output stream, a set of values that describe their memory structure, a set of states, a state transition set and a set of functions. Labels in transitions are functions which are triggered through an input symbol and a memory instance and produce an output symbol and a new memory instance (Figure 1).

X-machines can be thought to apply in similar cases where Statecharts and other similar notations, such as SDL, do. However, X-machines have other significant advantages. Firstly, they provide a mathematical modelling formalism for a system. Consequently, a model checking method for X-machines is devised [4] that facilitates the verification of safety properties of a model. Finally, they offer a strategy to test the

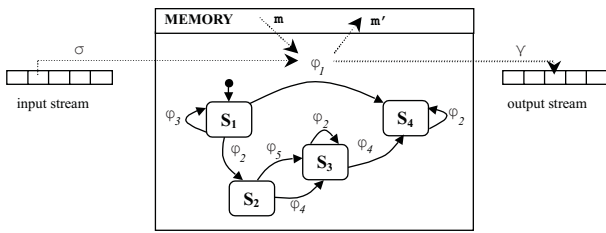


Fig. 1. An abstract X-machine model

implementation against the model [5], [6], which is a generalization of W-method for FSM testing. It is proved that this testing method is guaranteed to determine correctness if certain assumptions in the implementation hold [7]. In principle, X-machines are considered a generalization of models written in similar formalisms since concepts devised and findings proven for X-machines form a solid theoretical framework, which can be adapted to other, more tool-oriented methods, such as Statecharts or SDL.

In addition, communicating X-machines provide notation to create agents as instances of X-machine types and define their interaction and communication [8]. Functions can send messages to input streams of other X-machine agents which are consumed by local functions. In practice, it is found that the development of a communicating system model can be based on a number of well-defined distinct steps, i.e. development of types of X-machine models, creation of agents as instances of those types, construction of communicating agents, and then testing as well as model checking each of these agents individually. To each of the steps a set of appropriate tools, such as an interchange description language, parser, animator, test set generator etc., is employed in order to make the methodology applicable in real cases [9]. Such cases emerge during modelling naturally distributed systems, such as multi-agent systems. Here, we use a generalized example of a reactive agent in order to demonstrate the methodology and explain in detail how each activity, namely modelling, testing and verification is carried out. In the following, we present the methodology and in each of the sections of the paper we briefly present the theory behind each step. We then demonstrate in detail each proposed activity of the approach using a generalized example as a vehicle of study. Finally, we comment on the methodology and discuss further work to be carried out in order to deal with dynamically configurable systems as well as testing and verification of these systems as a whole.

## II. METHODOLOGY

Communicating X-machines is viewed as a modelling method, where a complex system can be decomposed in small agents (elements) modelled as simple X-machine models. The communication of all these agents is specified separately in order to form the complete system as a communicating X-machine model. This implies a modular bottom-up approach and supports an iterative gradual development. It also fa-

cilitates the reusability of existing X-machine type models, making the management of the whole project more flexible and efficient, achieving its completion with lower cost and less development time.

### A. Steps

The communicating X-machine method supports a disciplined modular development, allowing the developers to decompose the system under development into communicating agents and thus model interacting agent-based systems. We suggest that the development of a system model can be mapped into the following well-defined distinct actions:

- Develop X-machine type models (X-machine agent types) independently of the target system, or use existing type models as they are.
- Code the X-machine type model into a language that facilitates the subsequent steps. Use the animator that accompanies the language and get early feedback from the domain experts (informal verification).
- Express the desired properties in a suitable formalism (temporal logic) and use the formal verification technique (model checking) for X-machine type models in order to increase the confidence that the proposed model has the desired characteristics.
- Use testing strategies in order to test the implementation (Unit testing, where the unit is considered to be the agent type) against the model.
- Create X-machine agents and determine the way in which they communicate and interact.
- Extend the communicating system in order to achieve the desired overall functionality.

A set of appropriate tools has been developed and can be employed to each of the steps of the above methodology in order to make it applicable in real cases. Thus, apart from the mathematical notation used in step (a), all others are supported as follows:

- Step (b): coding of X-machine type model is carried out using the XMDL notation which acts as an interchange language for describing X-machine type models and its corresponding tools (syntax and type checker, compiler, animator)[9]. Through the animation, it is possible for the developers to informally verify that the model corresponds to the actual agent under development, and then also to demonstrate the model to the domain experts prompting them to identify any misconceptions regarding the user requirements between them and the development team.
- Step (c): formal verification of X-machine models is achieved with the use of an automated tool, a model checker. Model checking of X-machine models is supported by  $\mathcal{X}mCTL$ . This technique enables the designer to verify the developed model against temporal logic  $\mathcal{X}mCTL$  formulas which express the properties that the system should have.
- Step (d): test-cases for testing the implementation are automatically derived using the X-machine test case

generator. It is possible to use the formal testing strategy to test the implementation and prove its correctness with respect to the X-machine model.

- Step (e): the creation, communication and interaction of the agents are established through the XMDL-c notation and its corresponding tools. Achieve informal validation by demonstrating an analysis of the results from the animator for XMDL-c (simulation study) to domain experts.
- Step (f): all the above mentioned tools may be used to refine the resulting model.

There are many cases of naturally distributed multi-agent systems in which we have applied the above methodology [10], [11]. Here, we devised a generalized example of a reactive agent in order to demonstrate the methodology and explain in detail how each activity is carried out.

### B. Reactive Agent Case

Reactive agents are simple agents that their responses are very closely tied to perception and they do not possess any (or they have limited) knowledge about the environment. Their behaviour can be modelled with state machines and that is why the X-machine is a perfect candidate since it can provide very elegant models of such agents [10]. For reasons of demonstration our example is an abstract and generalized one. Assume a simple reactive agent (e.g. a reactive robot, software agent) (figure 2) which consumes items (e.g. objects of a physical or artificial environment, inputs, perceptions) of the environment, processes them, and produces new items (new physical objects, output, actions/effects). Each item is uniquely identified by an identification number and its description. The simplified agent system contains two buffers which are storage spaces of limited capacity and a single agent which carries out the processing. In order to control the two buffers, two control agents are handling the communication with the processing reactive agent and the buffers. The whole agent system now can be viewed as a simple reactive agent that could communicate with other similar agent systems, forming a more complex system, providing a simple way to scale up.

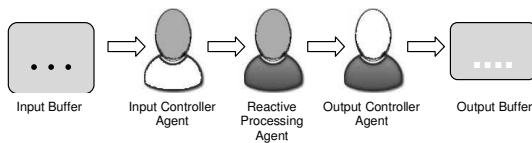


Fig. 2. Physical layout of the agent system

The items that are required to be processed by the processing agent are placed in the input buffer, while the items that have been processed by the reactive agent are stored in the output buffer. Items may be added in any buffer only if there is available space in it, while items are removed from a buffer in a specified order (e.g FIFO, LIFO or other discipline, handled by the input, output controlling agents). When the reactive agent is idle and there are items stored in the input buffer, the reactive agent may start the processing

of an item: The first item  $p$  placed in the input buffer is removed from the input buffer and the reactive agent starts processing it. The processing of the item lasts for  $t$  time units. If at the completion of the item processing, the output buffer is not full, then item  $p$  is placed in the output buffer and the machines either becomes idle or starts processing another item depending on whether the input buffer is empty or not. If, however, the output buffer is full when the reactive agent completes the processing of an item, the item  $p$  may not be removed from the reactive agent and thus the reactive agent is blocked. The reactive agent is unblocked when space becomes available in the output buffer.

## III. FORMAL MODELLING X-MACHINE TYPE MODELS

### A. Theory of X-machines

A deterministic stream X-machine [6] is an 8-tuple

$$X = (\Sigma, \Gamma, Q, M, \Phi, F, q_0, m_0)$$

where:

- $\Sigma$  and  $\Gamma$  are the input and output alphabets respectively.
- $Q$  is the finite set of states.
- $M$  is the (possibly) infinite set called memory.
- $\Phi$ , the *type* of the machine  $X$ , is a set of partial functions  $\varphi$  that map an input and a memory state to an output and a possibly different memory state,  $\varphi : \Sigma \times M \rightarrow \Gamma \times M$ .
- $F$  is the next state partial function,  $F : Q \times \Phi \rightarrow Q$ , which given a state and a function from the type  $\Phi$  determines the next state.  $F$  is often described as a state transition diagram.
- $q_0$  and  $m_0$  are the initial state and initial memory respectively.

The state diagram of an abstract X-machine model is shown in figure 1. An X-machine type is defined as a deterministic X-Machine without the initial state and the initial memory. Types will be used in order to create X-machine agents as shown later.

### B. Mathematical Modelling

Two X-machine types are naturally identified in the manufacturing facility example, i.e. the buffer and the processing reactive agent. For example, the state transition diagrams of these two X-machine types are depicted in figure 3.

Using mathematical notation, the definition of the buffer type is as follows:

- The set of inputs is  $\Sigma = \text{ITEMS}$  where  $\text{ITEMS} = \text{ITEM\_TYPE} \times \text{ID}$ ,  $\text{ITEM\_TYPE} = \{ \text{TypeA}, \text{TypeB}, \dots \}$  and the set of outputs  $\Gamma = \text{ITEMS} \times \text{MESSAGES}$ , where  $\text{MESSAGES} = \{ \text{item\_removed\_empty}, \text{item\_removed}, \text{item\_ignored}, \dots \}$ .
- The set of states is  $Q = \{ \text{empty}, \text{non\_empty}, \text{full} \}$ .
- The memory is  $M = \mathcal{P}\text{ITEMS} \times N$ , with  $N$  denoting the capacity of the buffer.
- The type of the X-machine is  $\Phi = \{ \text{add\_item}, \text{remove\_item}, \text{become\_empty}, \text{become\_full}, \text{ignore\_add} \}$ .



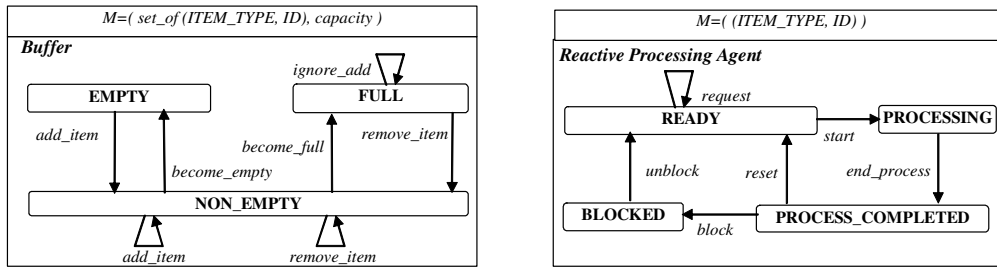


Fig. 3. X-machine model types of a processing reactive agent and a buffer

Finally, the functions  $\varphi \in \Phi$  of the X-machine need to be defined. For example the function `add_item` is defined as:

$$\text{add\_item}((type, id), (items, c)) =$$

$$(item\_added, (items \cup \{(type, id)\}, c))$$

$$\text{if } (type, id) \notin items \wedge card(items) + 1 \leq capacity$$

#### IV. CODING THE TYPE MODELS

##### A. Modelling with XMDL

X-machine modelling is based on a mathematical notation, which, however, implies a certain degree of freedom, especially as far as the definitions of functions are concerned. In order to make the approach practical and suitable for the development of tools around X-machines, a standard notation is devised and its semantics fully defined [9]. The aim is to use this notation, namely X-machine Definition Language (XMDL), as an interchange language between developers who could share models written in XMDL for different purposes. To avoid complex mathematical notation, the language symbols are completely defined in ASCII. Briefly, an XMDL model is a list of definitions corresponding to the construct tuple of the X-machine definition. The language also provides syntax for (a) use of built-in types such as integers, sets, sequences, bags, etc., (b) use of operations on these types, such as arithmetic, logic, set operations etc., (c) definition of new types, and (d) definition of functions and the conditions under which they are applicable. In XMDL, the functions take two parameter tuples, i.e. an input symbol and a memory value, and return two new parameter tuples, i.e. an output and a new memory value. A function may be applicable under conditions (*if-then*) or unconditionally. Variables are denoted by a preceding `?`. The informative *where* in combination with the operator `<-` is used to describe operations on memory values. A function has the following general syntax:

```
#fun <function name> (<input tuple>, <memory tuple>)=
if <condition expression> then
  (<output tuple>, <memory tuple>)
where <informative expression>.
```

The following is a part of the buffer model as coded in XMDL:

```
#model buffer.
#basic_types = [ITEM_TYPE].
#type ID = Natural.
#type capacity = Natural.
#type ITEM = (ITEM_TYPE, ID).
#type buffer = set_of ITEM.
```

```
#memory (buffer, capacity).
#input (ITEM).
#output (messages, ITEM).
#type messages = {item_added, item_removed, item_ignored,
  item_added_full, item_removed_empty}.
#states = {empty, non_empty, full}.
#transition (empty, add_item)=non_empty.
#transition (non_empty, add_item)=non_empty.
...
#fun add_item ((?ITEM_TYPE, ?ID), (?items, ?c)) =
if (?ITEM_TYPE, ?ID) not_belongs ?items and ?new_length < ?c
then ( item_added, (?ITEM_TYPE, ?ID), (?new_items, ?c) )
where
?new_items <- (?ITEM_TYPE, ?ID) addsetelement ?items and
?temp <- cardinality ?items and
?new_length <- ?temp + 1.
...
```

##### B. Animation of Models

X-System is a tool created to support modelling with X-machines [9]. Using XMDL as the modelling language, X-System allows the animation of X-Machine models. The parser of XMDL was built using the DCG (Definite Clause Grammars) notation, which is integrated in the Prolog language and is responsible for the syntax check of the models as well as type and logical errors. After the parser has ensured the correctness and completeness of a model, X-System allows its compilation into Prolog executable code. The Prolog code may then be used by X-System's animator, a program which implements an algorithm that simulates the computation of an X-machine. Through this simulation it is possible first of all for the developers to informally verify that the model simulates the actual system under development, and then also to demonstrate the model to the domain experts aiding them to identify any misconceptions regarding the user requirements between them and the development team.

#### V. $\mathcal{X}mCTL$ MODEL CHECKING

An automatic and formal verification technique for X-machines based on model checking is provided. This formal verification technique for X-machine models enables the designer to verify the developed model against temporal logic formulas that express the properties that the system should have. For this purpose an extended version of temporal logic was devised that is appropriate for X-machine models, named  $\mathcal{X}mCTL$  [4].

The version of temporal logic that is usually used to express the properties in model checking is the Computation Tree Logic (CTL). In CTL [12] each of the five temporal operators (**X**, **F**, **G**, **U**, **R**) must be preceded by either **A** (for all paths) or **E** (there exists path) path quantifiers. The temporal operators used in  $\mathcal{X}mCTL$  are the operators of CTL with the addition of two new memory quantifiers, namely  $\mathbf{M}_x$  and  $\mathbf{m}_x$ :

- $\mathbf{M}_x$  (for all memory instances) requires that a property holds at all possible memory instances of an X-machine state.
- $\mathbf{m}_x$  (there exists a memory instance) requires that a property holds at some memory instances of an X-machine state.

Having developed an X-machine model type it is possible to verify it for desired properties. The properties are expressed as  $\mathcal{X}mCTL$  formulas, which together with the X-machine model is given as input to the model checker. This tool outputs true if the model satisfies the property or false together with a counterexample. If the latter is the outcome the model is altered accordingly using the debugging information (counterexample) until the model will satisfy the property. When all formulas have been verified the X-machine model is proved to have all the desired properties, i.e. the model is “correct” with respect to the requirements.

For example in the case of the buffer model the property the number of elements in the sequence will never exceed buffer’s capacity can be expressed with the following  $\mathcal{X}mCTL$  formula:  $\mathbf{AG} \mathbf{M}_x(\text{card}(M(1)) \leq M(2))$ . The notation  $M(i)$  is used in  $\mathcal{X}mCTL$  to denote the  $i$ -th variable in the memory. The formula can be interpreted as: for all computational paths of the X-machine model and for all states in these paths the cardinality of the sequence holding the items will be always less or equal to the capacity of the buffer for all memory instances of each state.

## VI. TESTING

One important advantage that modelling with X-machines has to offer is the fact that it allows for complete testing of the models. The devised testing strategy for X-machine models was proved to find all faults in an implementation [13] and it is a generalisation of Chow’s  $W$ -method for the testing of FSMs[14]. The method works based on certain assumptions, and design-for-test conditions, i.e. output distinguishability and completeness, and can produce a complete test set of input sequences. In order to check whether the design-for-test conditions are met, the executable model described above is used by providing a transition cover set ( $S$ ) and a characterisation set ( $W$ ). Informally, a characterisation set  $W \subseteq \Phi^*$  is a set of processing functions for which any two distinct states of the machine are distinguishable. The state cover  $S \subseteq \Phi^*$  is a set of processing functions such that all states are reachable by the initial state.

In the provided example, the modeller can derive the transition cover set and a characterisation set of the processing reactive agent model:  $W = \{\text{start, reset, unblock, end\_process}\}$ ,  $S = \{\text{request,}$

$\text{start, start\_end\_process, start\_end\_process\_block, start\_end\_process\_block\_unblock}\}$

Consequently, the complete test case set is produced by applying the test case function [6] and indicatively a test case is:

```
test case 1
input sequence: {typeA, 1} (finish, process)
                (out_buffer, full) (out_buffer, not_full)
output sequence: process_started processing_finished
                 agent_block agent_unblock
```

## VII. COMMUNICATION OF AGENTS

### A. Theory of Communicating X-machines

A Communicating X-machine System  $Z$  as defined in [8] is a 2-tuple:

$$Z = ((C_i)_{i=1,\dots,n}, CR)$$

where:

- $C_i$  is the  $i$ -th Communicating X-machine agent, and
- $CR$  is a relation defining the communication among the agents,  $CR \subseteq C \times C$  and  $C = \{C_1, \dots, C_n\}$ . A tuple  $(C_i, C_k) \in CR$  denotes that the X-machine agent  $C_i$  can output a message to a corresponding input stream of X-machine agent  $C_k$  for any  $i, k \in \{1, \dots, n\}$ ,  $i \neq k$ .

Communicating X-machine model consists of several X-machine agents that are able to interact by exchanging messages. The structure  $CR$  defines a directed graph which statically determines the direction of messages between agents. An X-machine agent is defined as an X-machine, i.e. X-machine type with initial memory and initial state, in which the functions do not only read and write from/to their input and output streams respectively but also read and write from/to streams that are used to communicate with other X-machine agents. More analytically, functions are of the form:  $\varphi_i((\sigma)_j, m) = ((\gamma)_k, m')$  where  $(\sigma)_j$  means that input is provided by machine  $C_j$  whereas  $(\gamma)_k$  denotes an outgoing message to machine  $C_k$ . If  $i = j$  and/or  $i = k$ , that means that machine  $C_i$  reads from its standard input stream and/or writes to its standard output stream.

Graphically on the state transition diagram we denote the acceptance of input by a stream other than the standard by a solid circle along with the name  $C_j$  of the communicating X-machine agent that sends it. Similarly, a solid diamond with the name  $C_k$  denotes that output is sent to the  $C_k$  communicating X-machine agent.

### B. Creation of Communicating Agents with XMDL-c

XMDL has also been extended (XMDL-c) in order to code communicating agents. XMDL-c is used to define instances of models by providing a new initial state and a new initial memory instance:

```
#model <model_instance> instance_of <model_type>
with:
#init_state <initial_state>;
#init_memory <initial_memory>.
```

In addition, XMDL-c provides syntax that facilitates the definition of the communicating functions. The general syntax is the following:

```
#communication of function <function_name>:
#reads from <model instance>;
#writes <message tuple> to <model_instance>
  using <variable> from output <output tuple> and
  using <variable> from input <input tuple> and
  using <variable> from memory <memory tuple>
where <informative expression>.
```

A function can either read or write or both from other agents (model instances). It is not necessary to specify the incoming message because it is of the same type as the input defined in the original agent. However, it is necessary to specify the outgoing message as a tuple which may contain values that exist in either output or input tuples of the function or even in the memory tuple of the agent. The informative expression is used to perform various operations on these values before they become part of the outgoing message tuple.

### C. Compiling X-machine agents

CommX-System is a tool created to support modelling with Communicating X-machines [8]. To start with, CommX-System initially uses an XMDL-c description of the communication interface of a system's agent. The parser ensures the syntactic and logical correctness of the description, the compiler performs the semantic analysis and transforms the description into executable code. The compiler is then responsible for combining the communication code with that of the actual model code. One unique executable file is produced corresponding to the communicating agent of the overall system. After all the above have been performed for each of the agents of the system, all produced files are combined to create one that corresponds to the entire system and which will be used by the tool's animator.

Indicatively, we present a part of the communicating system. According to the description of the problem, the input buffer and the processing reactive agent should communicate in the sense that an item should be sent from the buffer to the reactive agent when the reactive agent is ready to process it. The following XMDL-c code creates an agent for the (input) buffer and its communication with the reactive agent agent.

```
#model buf_in instance_of buffer
with:
#init_state {empty};
#init_memory (nil, 5).

#communication of function remove_item:
#writes ((?item,?id)) to (mach)
using ?item from output (?m, (?item,?id)) and
using ?id from output (?m, (?item,?id)).

#communication of function become_empty:
#writes ((?item,?id)) to (mach)
using ?item from output (?m, (?item,?id)) and
using ?id from output (?m, (?item,?id)).
```

Similarly, the following XMDL-c code defines the reactive agent model and its communication with the input buffer. The communication between these two agent X-machine models is

depicted in figure 4. The reactive agent and the output buffer interact in a similar way.

```
#model robot1 instance_of reactive agent
with:
#init_state {ready};
#init_memory ((none,0)).

#communication of function start:
#reads from buf_in.
#end.
```

## VIII. THE OVERALL SYSTEM

So far, we have followed steps (a) to (e) of the methodology and we assume that all agents (the input buffer, the processing reactive agent and the output buffer) have been developed, animated, verified and tested as well as communication between them has been established.

The system modelled at this stage is a simple agent system with one reactive agent that has an input and an output buffer as depicted in figure 2 without the controller agents. The input buffer stores the items added in it and communicates to the reactive agent an item stored in the buffer, whenever the reactive agent needs one; i.e. the input buffer models a heap. In a similar fashion, the output buffer, accepts an item processed by the reactive agent and stores it. In the case that it is required to employ an input buffer with another discipline (e.g. FIFO), then it is necessary to create another buffer agent that has the same transition diagram as the one shown in figure 4, but different implementation of the transition function `remove_item`. It is therefore evident that this approach is not coping well with changes mainly because it restricts reusability.

The last step (f) demonstrates the flexibility of the proposed methodology addressing this issue. Changes in the system behaviour can be easily handled by the proposed methodology with the addition of (probably off-the-self) controller agents that encapsulate the desired behaviour. In the manufacturing facility case the input controller (ctrl-in) has been added to control the feeding of the items from the input buffer to the reactive agent in a FIFO manner. The complete model is depicted in figure 5 providing a flexible and modular solution. Any change of the requirements e.g. in the manner the items are fed to the reactive agent can be dealt with the use of a different specialised input controller, replacing the old one in the model.

It has been demonstrated that the proposed methodology offers an intuitive way to model agent-based systems by providing the flexibility that each agent identified in the real system is mapped directly to an X-machine agent model in the design of the system. By applying this methodology to agent-based systems it is possible to incrementally model the complete system and assure that all desired properties of the agents of the system hold in the final product.

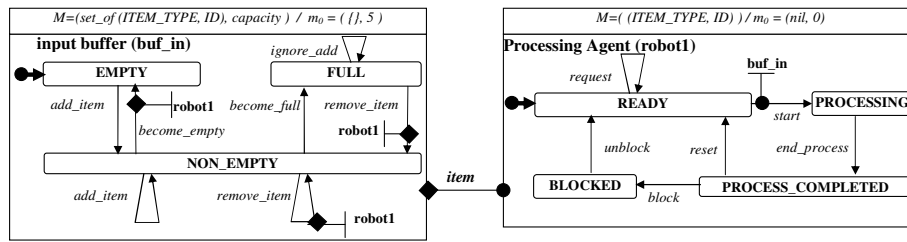


Fig. 4. Communication between an instance of a Processing Agent and an Input Buffer

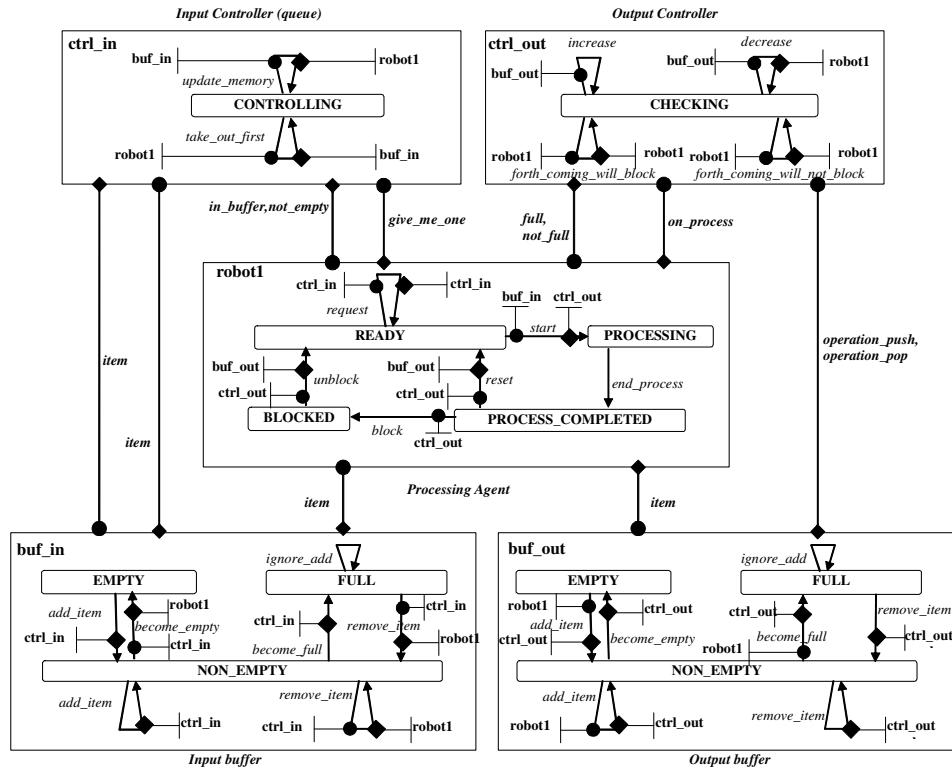


Fig. 5. The model of the complete system

IX. CONCLUSIONS AND FURTHER WORK

We have presented a methodology for developing simple reactive agent-based systems using communicating X-machines formal method. X-machines attracted many researchers interest over the last twenty years [15] mainly because of the intuitiveness in modelling reactive systems and the additional features they provide in terms of testing and verification. The main contribution of this paper is the detailed description of the methodology that allows to scale up to larger and more complex systems with a focus on developing correct components, the formal introduction of the X-machine type models and their instances as parts of a more complex system and the simple but general and complete example to demonstrate the applicability of the proposed method. The methodology and its accompanying tools impose an incremental bottom-up practical development. It is useful in cases where complex systems can be viewed as an aggregation of simple agents that

can communicate in order to achieve the overall behaviour of a distributed system. A particular example is the multi-agent systems [3], in which similar methodologies might be employed, such as Gaia, AAIL, Cassiopeia etc. [1].

With the continuous verification and testing of agents from the early stages risks are reduced and the developer is confident of the correctness of the system under development throughout the whole process. It is worth noticing that the proposed methodology utilises a priori techniques to avoid any flaws in the early stages of the development together with a posteriori techniques to discover any undiscovered flaws in later stages. This way it makes the best use of the development effort to achieve highest confidence in the quality of the developed agents that have been verified and tested therefore they can be reused as trusted agents. The proposed communicating X-machine concept is based on the idea of reusability, thus minimizing the development time without risking the quality of

the product. Further works also include modelling of dynamic systems, models of which the configuration of communicating agents changes over time. A set of appropriate rules has been devised [16], [11] and ideas are borrowed from biological computational paradigms, such as membrane computing [17], in order to facilitate definition of the appropriate hybrid formal method [18].

There is a need for the extension of the model checking technique in order to facilitate the formal verification of communicating X-machine models. Research is conducted towards this direction but also towards the establishment of a successful testing strategy for the communicating X-machine models [19]. Finally there is a continuous need to evaluate the proposed methodology with real case studies and with industrial development teams to prove and not demonstrate its applicability and to compare with other industrial strength methodologies.

#### REFERENCES

- [1] M. Wooldridge and P. Ciancarini, "Agent-Oriented Software Engineering: The State of the Art," in *Agent-Oriented Software Engineering*, ser. Lecture Notes in AI, P. Ciancarini and M. Wooldridge, Eds. Springer-Verlag, 2001, vol. 1957, pp. 337–350.
- [2] B. Meyer, "The Grand Challenge of Trusted Components," in *25th International Conference on Software Engineering*, Portland, Oregon, May 2003, pp. 660–667.
- [3] P. Kefalas, M. Holcombe, G. Eleftherakis, and M. Gheorghe, "A formal method for the development of agent-based systems," in *Intelligent Agent Software Engineering*, V. Plekhanova, Ed. Idea Group Publishing, 2003, ch. 4, pp. 68–98.
- [4] G. Eleftherakis and P. Kefalas, "Formal Verification of Generalised State Machines," in *12th Panhellenic Conference on Informatics (PCI 2008)*, S. Gritzalis, D. Plexousakis, and D. Pnevmatikatos, Eds. Samos, Greece: IEEE Computer Society, Aug. 2008, pp. 227–231.
- [5] F. Ipate, "Complete deterministic stream X-machine testing," *Formal Aspects of Computing*, vol. 16, no. 4, pp. 374–386, Nov. 2004.
- [6] M. Holcombe and F. Ipate, *Correct Systems: Building a Business Process Solution*. London: Springer Verlag, 1998.
- [7] F. Ipate and M. Holcombe, "An integration testing method that is proved to find all faults," *International Journal of Computer Mathematics*, vol. 63, no. 3, pp. 159–178, 1997.
- [8] P. Kefalas, G. Eleftherakis, and E. Kehris, "Communicating X-machines: a practical approach for formal and modular specification of large systems," *Information and Software Technology*, vol. 45, no. 5, pp. 269–280, Apr. 2003, woS Impact Factor (1.821 - 1.426/5y).
- [9] P. Kefalas, G. Eleftherakis, and A. Sotiriadou, "Developing Tools for Formal Methods," in *9th Panhellenic Conference on Informatics*, Thessaloniki, Nov. 2003, pp. 625–639.
- [10] G. Eleftherakis, P. Kefalas, A. Sotiriadou, and E. Kehris, "Modeling Biology Inspired Reactive Agents Using X-machines," *Proceedings of World Academy of Science, Engineering and Technology*, vol. 1, pp. 93–96, Jan. 2005, also published in: International Conference on Computational Intelligence (ICCI04) [c12].
- [11] P. Kefalas, I. Stamatopoulou, I. Sakellariou, and G. Eleftherakis, "Transforming Communicating X-machines into P Systems," *Natural Computing*, vol. 8, no. 4, pp. 817–832, Dec. 2009.
- [12] E. Clarke, O. Grumberg, and D. Peled, *Model Checking*. Cambridge, Massachusetts: MIT Press, 1999.
- [13] F. Ipate and M. Holcombe, "Specification and testing using generalised machines: a presentation and a case study," *Software Testing, Verification and Reliability*, vol. 8, pp. 61–81, 1998.
- [14] T. Chow, "Testing Software Design Modeled by Finite-State Machines," *IEEE Transactions on Software Engineering*, vol. 4, no. 3, pp. 178–187, 1978.
- [15] "Special Issue on X-machines," *Formal Aspects of Computing*, vol. 12, no. 6, 2000.
- [16] P. Kefalas, G. Eleftherakis, M. Holcombe, and I. Stamatopoulou, "Formal modelling of the dynamic behaviour of biology-inspired agent-based systems," in *Molecular Computational Models: Unconventional Approaches*, M. Gheorghe, Ed. Idea Group Publishing, 2005, ch. 9, pp. 243–276.
- [17] P. Kefalas, I. Stamatopoulou, M. Gheorghe, and G. Eleftherakis, "Membrane Computing and X-machines," in *The Oxford Handbook of Membrane Computing*, G. Paun, Ed. Oxford University Press, 2009, ch. 23.4, pp. 612–620.
- [18] I. Stamatopoulou, P. Kefalas, and M. Gheorghe, "Operas: A framework for the formal modelling of multi-agent systems and its application to swarm-based systems," in *ESAW*, ser. Lecture Notes in Computer Science, A. Artikis, G. M. P. O'Hare, K. Stathis, and G. A. Vouros, Eds., vol. 4995. Springer, 2007, pp. 158–174.
- [19] F. Ipate, T. Bălănescu, and G. Eleftherakis, "Testing Communicating Stream X-machines," in *1st Balkan Conference on Informatics*, Thessaloniki, Nov. 2003, pp. 161–173.

# Monitoring Building Indoors through Clustered Embedded Agents

Giancarlo Fortino, Antonio Guerrieri

DEIS – University of Calabria,

Via P. Bucci, cubo 41c,

Rende (CS), 87036, Italy

Email: g.fortino@unical.it, aguerrieri@deis.unical.it

**Abstract**—Future buildings will be smart to support personalized people comfort and building energy efficiency as well as safety, emergency, and context-aware information exchange scenarios. In this work we propose a decentralized and embedded architecture based on agents and wireless sensor and actuator networks (WSANs) for enabling efficient and effective management of buildings. The main purpose of the agent-based architecture is to efficiently support distributed and coordinated sensing and actuation operations. The building management architecture is implemented in MAPS (Mobile Agent Platform for Sun SPOTs), an agent-based framework for programming WSN applications based on the Sun SPOT sensor platform. The proposed architecture is demonstrated in a simple yet effective operating scenario related to monitoring workstation usage in computer laboratories. The high modularity of the proposed architecture allows for easy adaptation of higher-level application-specific agents that can therefore exploit the architecture to implement intelligent building management policies.

## I. INTRODUCTION

NOWADAYS, due to advances in communication and computing technologies, the need to have high comfort levels together with an optimization of the energy consumption is becoming important for inhabitants of buildings. Moreover, buildings should also support their inhabitants with automatic emergency and safety procedures as well as context aware information services. To meet all these requirements, future buildings have to incorporate diversified forms of intelligence [1].

We believe that agent-based computing [2] can be exploited to implement the concept of intelligent buildings due to the agent features of autonomy, proactiveness, reactivity, learnability, mobility and social ability. Specifically agents can continuously monitor building indoors and their living inhabitants to gather useful data from people and environment and can cooperatively achieve even conflicting specific goals such as personalized people comfort and building energy efficiency.

A few research efforts based on agents have been to date proposed to design and implement intelligent building systems [3][4][5]. However, none of them provide agents embedded in the sensor and actuator devices that would introduce intelligence decentralization and improve system efficiency. This is due to the exploitation of conventional sensing and actuation systems that do not offer distributed computing devices for sensing and actuation. To overcome this

limitation, wireless sensor and actuator networks (WSAN) [6] can be adopted. WSANs represent a viable and more flexible solution to traditional building monitoring and actuating systems (BMAS), which require retrofitting the whole building and therefore are difficult to implement in existing structures. In contrast, WSAN-based solutions for monitoring buildings and controlling equipment, such as electrical devices, heating, ventilation and cooling (HVAC), can be installed in existing structures with minimal effort. This should enable monitoring of structure conditions, and space and energy (electricity, gas, water) usage while facilitating the design of techniques for intelligent device actuation.

In this paper we propose a decentralized and embedded management architecture for intelligent buildings that is based on WSANs and overcomes the limitations of the aforementioned solutions [3][4][5]. In particular, the aim of our architecture is to optimize and fully decentralize the sensing and actuation operations through distributed cooperative agents both embedded in sensor/actuator devices and running on more capable coordinators (PC, plug computers, PDA/smartphones). The proposed architecture can be easily programmed to support a wide range of building management applications integrating comfort, energy efficiency, emergency, safety, and context-aware information exchange aspects.

The rest of this paper is organized as follows. Section II describes approaches related to our work. In Section III the proposed agent-based architecture for building management is defined. Section IV presents the MAPS-based implementation of the architecture, specifically the sensor/actuator agents. Section V shows the system GUI and a system deployment for monitoring the workstation usage in computer laboratories. Finally, conclusions are drawn and directions of future work elucidated.

## II. RELATED WORK

In [3] the authors present the MASBO (Multi-Agent System for Building cOntrol) architecture that aims to provide a set of software agents to support both on-line and off-line applications for intelligent work environments. MASBO is used to develop a multi-agent system (MAS) able to tradeoff energy saving and inhabitants' preferences where preferences can be learnt and predicted through an unsupervised online real-time learning algorithm (analyzing inhabitants' behavior).

MASBO agents reside on a server and constantly monitor data from sensors and eventually actuate some commands. MASBO works as an enhancement to an existing building automation system by adding learning, reasoning and autonomous capabilities. The responsibility of controlling sensors and actuators, and keeping a requested environmental value constant is not addressed by MASBO.

In [4] the authors propose a working solution to the problem of thermal resource distribution in a building using a market-based MAS. Computational agents representing individual temperature controllers bid to buy or sell cool or warm air. The agents, running in a monolithic process on a workstation, are able to distribute the thermal resources so that all the building offices have an equitable temperature distribution. Temperature sensors and air flow actuators are all accessible directly through distributed hardware modules via a network connection.

In [5] the authors describe a MAS that monitors and controls an office building in order to provide added values like energy saving together with the delivery of energy. The developed system is distributed in the sense that some agents are located on PDAs and others run on the Bluetooth access points (workstations) that communicate with the PDAs. The system makes use of the existing power lines for communication between the agents and the sensing and actuation system controlling lights, heating, ventilation, etc.

Differently from the described approaches, our agent-based architecture embeds agents into the wireless sensor and actuator network used as infrastructure for building monitoring and control. This important feature would provide decentralized intelligence and improve system efficiency.

### III. AGENT-BASED ARCHITECTURE

The agent-based architecture (see Fig. 1) for decentralized and embedded building management is composed of a building manager agent (BMA), which is installed in the control workstation, coordinator agents (CAs), which run in the basestations, and sensor agents (SAs), which are executed in the sensor/actuator nodes. Specifically, the architecture relies on a multi-basestation approach to allow for large buildings composed of multiple floors and diversified environments. Thus, the architecture is purposely hybrid: hierarchical and peer-to-peer. Interaction between CAs is peer-to-peer whereas interactions between CAs and their related SAs (or SA cluster) and between BMA and CAs are usually master/slave. Moreover, SAs of the same cluster coordinate to dynamically form up a multi-hop ad-hoc network rooted at the master CA.

In Fig. 2 the main functionalities of BMA, CA and SA are shown according to a layered organization that is partially derived from the Building Management Framework (BMF) [7].

The BMA makes it available the monitoring and control GUI through which the building manager can issue requests to configure/program the agent-based building network and visualize its status and the monitored data. Moreover, the BMA can be purposely extended to incorporate goal-directed

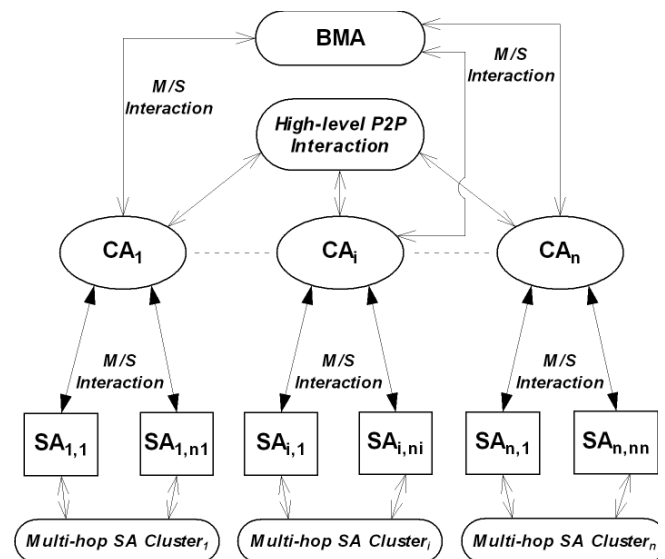


Fig. 1 Agent-based architecture for decentralized and embedded management of buildings based on wireless sensor and actuator networks.

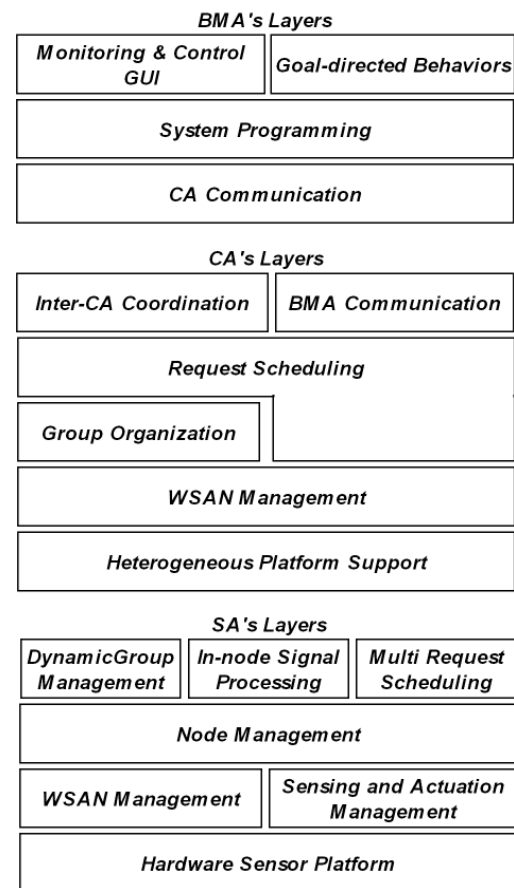


Fig. 2 The layered organization of BMA, CA and SA.

ed behaviors for implementing specific building monitoring and control strategies.

The CA includes the following layers:



– *Heterogeneous Platform Support* incorporates a set of adapters that allow interfacing the system with different type of sensor/actuator platforms. An adapter is linked to a specific hardware device able to communicate with a specific sensor platform in the network.

– *WSAN Management* allows to fully manage a WSAN cluster. This layer supports packet coding/decoding according to the BMF application-level protocol and packet transmission/reception to/from the WSAN cluster. Moreover, this layer supports device discovery within the cluster.

– *Group Organization* provides group-based programming of sensors and actuators, tracking of nodes and groups in the system, and management of node configurations and group compositions. Node organization in groups is specifically defined to capture the morphology of buildings. Nodes belong to groups depending on their physical (location) or logical (operation type) characteristics.

– *Request Scheduling* allows the support for higher-level application-specific requests. Through this layer, a CA can ask for the execution of specific tasks to single or multiple SAs or groups of SAs. Moreover, this layer keeps track of the requests submitted to the system, waits for data from the nodes and passes them to the requesting applications. A request is formalized through the following tuple:  $R = \langle \text{Obj}, \text{Act}, R, \text{LT} \rangle$ , where Obj is a specific sensor or actuator belonging to a node, Act is the action to be executed on Obj, R is the frequency of each executed Act, LT is the length of time over which these actions are to be reiterated. Moreover, a request can target a single node or a group of nodes having Obj.

– *Inter-CA Coordination* offers efficient mechanisms for coordination between CAs. Specifically, CAs cooperate for submitting queries and retrieving data spanning multiple SA clusters.

The SA is designed around the following layers:

– *Hardware Sensor Platform* allows to access the hardware sensor/actuator platform. In particular, the layer facilitates the configuration of the platform specific drivers and the use of the radio.

– *WSAN Management* manages the node communication with the reference CA according to the BMF application protocol and among the cluster nodes through the network protocol provided by the node sensor platform.

– *Sensing and Actuation Management* allows to acquire data from sensors and execute actions on actuators. In particular, this layer allows to address different types of sensors/actuators in a platform independent way.

– *Node Management* is the core of the SA and allows to coordinate all the layers for task execution. In particular, it handles events from the lower layers every time that a network packet arrives or data from sensor/actuator are available, and from the upper layers every time that data are processed or a stored request has to be executed.

– *Dynamic Group Management* provides group management functionalities to the SA. A node can belong to several groups at the same time and its membership can be dynamically updated on the basis of requests from CAs.

– *In-node Signal Processing* allows the SA to execute signal processing functions on data acquired from sen-

sors [8]. It can compute simple aggregation functions (e.g. mean, min, max, variance, R.M.S.) and more complex user-defined functions on buffers of acquired data.

– *Multi Request Scheduling* allows the scheduling of sensing and actuation requests. In particular, it stores the requests from CAs and schedules them according to their execution rate.

#### IV. MAPS-BASED IMPLEMENTATION

The agent-based building management architecture is currently implemented through MAPS [9], our agent-based framework for developing WSN applications on the Sun SPOT sensor platform. In this section we first provide a brief overview of MAPS (details can be found in [9, 10]) and, then, present the MAPS-based implementation of the proposed building management architecture at sensor-node side, specifically behavior and event-based interactions of the SA.

##### A. MAPS: a brief overview

MAPS [9, 10] is an innovative Java-based framework specifically developed on Sun SPOT technology for enabling agent-oriented programming of WSN applications. It has been defined according to the following requirements:

– *Component-based lightweight agent server architecture* to avoid heavy concurrency and agents cooperation models.

– *Lightweight agent architecture* to efficiently execute and migrate agents.

– *Minimal core services* involving agent migration, agent naming, agent communication, timing and sensor node resources access (sensors, actuators, flash memory, and radio).

– *Plug-in-based architecture* extensions through which any other service can be defined in terms of one or more dynamically installable components implemented as single or cooperating (mobile) agents.

– *Use of Java language* for defining the mobile agent behavior.

The architecture of MAPS (see Fig. 3) is based on several components interacting through events and offering a set of services to mobile agents, including message transmission, agent creation, agent cloning, agent migration, timer handling, and an easy access to the sensor node resources. In particular, the main components are the following:

– *Mobile Agent (MA)*. MAs are the basic high-level component defined by user for constituting the agent-based applications.

– *Mobile Agent Execution Engine (MAEE)*. It manages the execution of MAs by means of an event-based scheduler enabling lightweight concurrency. MAEE also interacts with the other services-provider components to fulfill service requests (message transmission, sensor reading, timer setting, etc) issued by MAs.

– *Mobile Agent Migration Manager (MAMM)*. This component supports agents migration through the Isolate (de)hibernation feature provided by the Sun SPOT environment. The MAs hibernation and serialization involve data and execution state whereas the code must already reside at the destination node (this is a current limitation of the Sun SPOTs

which do not support dynamic class loading and code migration).

– *Mobile Agent Communication Channel (MACC)*. It enables inter-agent communications based on asynchronous messages (unicast or broadcast) supported by the Radiogram protocol.

– *Mobile Agent Naming (MAN)*. MAN provides agent naming based on proxies for supporting MAMM and MACC in their operations. It also manages the (dynamic) list of the neighbor sensor nodes which is updated through a beaconing mechanism based on broadcast messages.

– *Timer Manager (TM)*. It manages the timer service for supporting timing of MA operations.

– *Resource Manager (RM)*. RM allows access to the resources of the Sun SPOT node: sensors (3-axial accelerometer, temperature, light), switches, leds, battery, and flash memory.

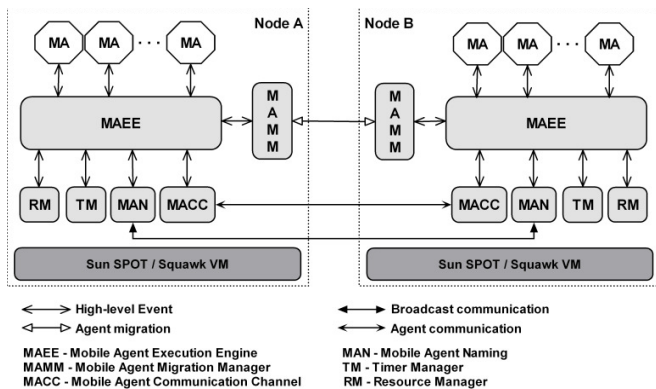


Fig. 3 The architecture of MAPS.

The dynamic behavior of a mobile agent (MA) is modeled through a multi-plane state machine (MPSM). Each plane may represent the behavior of the MA in a specific role so enabling role-based programming. In particular, a plane is composed of local variables, local functions, and an automaton whose transitions are labeled by Event-Condition-Action (ECA) rules  $E[C]/A$ , where  $E$  is the event name,  $[C]$  is a boolean expression evaluated on global and local variables, and  $A$  is the atomic action. Thus, agents interact through events, which are asynchronously delivered and managed by the MAEE component.

It is worth noting that the MPSM-based agent behavior programming allows exploiting the benefits deriving from three main paradigms for WSN programming: event-driven programming, state-based programming and mobile agent-based programming.

MAPS is also interoperable with the JADE framework [11]. Specifically, a JADE-MAPS gateway [12] has been developed for allowing JADE agents to interact with MAPS agents and vice versa. While both MAPS and JADE are Java-based, they use a different communication method. JADE sends messages according to the FIPA standards (using the ACL specifications), while MAPS creates its own messages based on events. Therefore, the JADE-MAPS Gateway facilitates message exchange between MAPS and JADE agents. This inter-platform communication infrastruc-

ture allows rapid prototyping of WSN-based distributed applications/systems that use JADE at the basestation/coordinator/host sides and MAPS at the sensor node side.

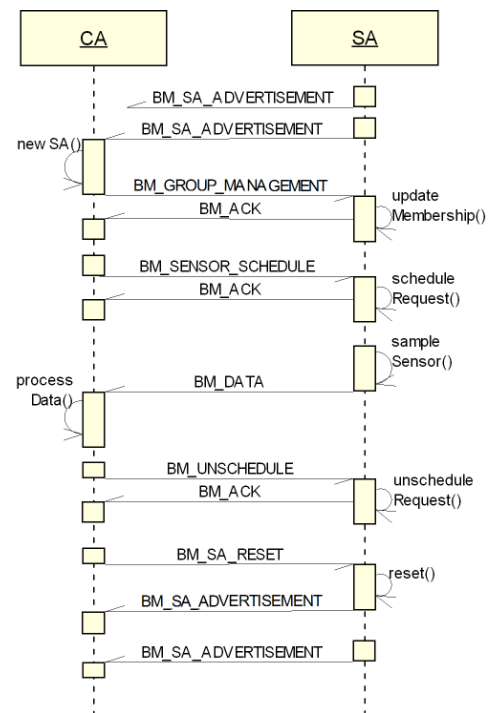


Fig. 4 Sequence Diagram of the interactions between CA and SA

### B. MAPS-based sensor agents

The MAPS-based SA (hereafter simply named SA) interacts with its cluster CA through events as sketched in the sequence diagram of Fig. 4. Once the SA is created, it periodically emits the `BM_SA_ADVERTISEMENT` event until the CA sends a configuring event (group management or request scheduling). Through the `BM_GROUP_MANAGEMENT` event, the CA manages the membership of target SAs (see section III). After the SA processes the received event, it sends the `BM_ACK` event to the CA.

The `BM_SENSOR_SCHEDULE` (or `BM_ACTUATOR_SCHEDULE`) event allows to request a specific sensing (or actuation) operation to target SAs. The SA transmits sensed (processed) data to the CA through the `BM_DATA` event. The CA can unschedule previously scheduled requests through the `BM_UNRESCHEDULE` event. Finally the CA sends out the `BM_SA_RESET` event to reset target SAs.

Tables I and II reports the defined MAPS-based building management events and the predefined values of their parameters. In particular, an event is defined by its *standard parameters*: EventSender ID, EventTarget ID, Event Type, Event Occurrence. The defined events are of two possible super types: `MSG` (sent by CA to SA) and `MSG_TO_BASESTATION` (sent by SA to CA). Both types are further specialized in the defined BM events as reported in the pairs  $\langle \text{MSG\_TYPE}, \text{BM\_event} \rangle$  of the 3<sup>rd</sup> column of Table I. Moreover, each event type has its own *additional parameters*, which are described in Table II. It is worth noting that the `ADDRESSEE` value can be set through the fol-

TABLE I.  
DEFINED BUILDING MANAGEMENT EVENTS

Event Name	Standard Parameters	Additional Parameters <KEY, VALUE>
BM_SA_ADVERTISEMENT	ID_SA; ID_CA; Event.MSG_TO_BASESTATION; Event.NOW	<MSG_TYPE, BM_SA_ADVERTISEMENT> <SENSOR_TYPE, VALUE>* <ACTUATOR_TYPE, VALUE>* if exists(<SENSOR_TYPE, VALUE>*) <FUNCTION, VALUE>*
BM_SENSOR_SCHEDULE	ID_CA; ID_SA; Event.MSG; Event.NOW	<MSG_TYPE, BM_SENSOR_SCHEDULE> <ADDRESSEE_TYPE, VALUE> <ADDRESSEE, VALUE> <REQUEST_ID, VALUE> <PERIOD_TIMESCALE, VALUE> <PERIOD_VALUE, VALUE> <LIFETIME_TIMESCALE, VALUE> <LIFETIME_VALUE, VALUE> <SENSOR_TYPE, VALUE> <DATA_TYPE, VALUE> <SYNTHETIC_DATA_TYPE, VALUE> if DATA_TYPE.VALUE == THRESHOLD_NOTIFICATION <THRESHOLD_TYPE, VALUE> <THRESHOLD_VALUE, VALUE> endif
BM_ACTUATOR_SCHEDULE	ID_CA; ID_SA; Event.MSG; Event.NOW	<MSG_TYPE, BM_ACTUATOR_SCHEDULE> <ADDRESSEE_TYPE, VALUE> <ADDRESSEE, VALUE> <REQUEST_ID, VALUE> <PERIOD_TIMESCALE, VALUE> <PERIOD_VALUE, VALUE> <LIFETIME_TIMESCALE, VALUE> <LIFETIME_VALUE, VALUE> <ACTUATOR_TYPE, VALUE> <ACTUATOR_PARAM, VALUE>*
BM_UNCHEDULE	ID_CA; ID_SA; Event.MSG; Event.NOW	<MSG_TYPE, BM_UNCHEDULE> <ADDRESSEE_TYPE, VALUE> <ADDRESSEE, VALUE > <REQUEST_ID, VALUE>
BM_GROUP_MANAGEMENT	ID_CA; ID_SA; Event.MSG; Event.NOW	<MSG_TYPE, BM_GROUP_MANAGEMENT> <ADDRESSEE_TYPE, VALUE > <ADDRESSEE, VALUE> <MEMBERSHIP_TYPE, VALUE> <MEMBERSHIP_COUNT, VALUE> if MEMBERSHIP_TYPE.VALUE != RESET <MEMBERSHIP_GROUPS, VALUE>
BM_SA_RESET	ID_CA; ID_SA; Event.MSG; Event.NOW	<MSG_TYPE, BM_SA_RESET> <ADDRESSEE_TYPE, VALUE > <ADDRESSEE, VALUE >
BM_DATA	ID_SA; ID_CA; Event.MSG_TO_BASESTATION; Event.NOW	<MSG_TYPE, BM_DATA> <TIMESTAMP, VALUE> <REQUEST_ID, VALUE> <RESULT, VALUE>
BM_ACK	ID_SA; ID_CA; Event.MSG_TO_BASESTATION; Event.NOW	<MSG_TYPE, BM_ACK> <MSG_TYPE_TO_ACK, VALUE> <ACK_PARAM, VALUE>

lowing regular expression:  $SA+ | ([NOT] G [TSO [NOT] G]*)$ , where SA is a sensor agent of the building management architecture, G is an element from the set of defined groups, STO is a set theory operator (e.g. union, intersection, difference) and NOT is the negation. Thus, the addressee of an event can be either one or more SAs, or SAs belonging to groups or complex compositions of groups.

The SA agent behavior consists of two types of planes: Manager plane and Request plane. While the Manager plane is created at the SA creation time and handles all node targeting events, a Request plane is created by the Manager plane every time that a new request schedule is received. This type of plane is removed when it completes its task or

TABLE II.  
ADDITIONAL PARAMETERS OF THE BUILDING MANAGEMENT EVENTS

Additional Parameter	Description	PREDEFINED VALUES
ADDRESSEE_TYPE	The type of event target	SA, List of SAs, GROUP, GROUP COMPOSITION
ADDRESSEE	The event target	SA+   ([NOT] G [STO [NOT] G]*)
REQUEST_ID	The unique identifier of a request	no predefined int value
PERIOD_VALUE	The period of the request execution	no predefined int value
PERIOD_TIMESCALE	The timescale of the period	MSEC, SEC, MIN, HOUR, DAY
LIFETIME_TIMESCALE	The lifetime of the request	MSEC, SEC, MIN, HOUR, DAY
LIFETIME_VALUE	The timescale of the request	no predefined int value
SENSOR_TYPE	The specific sensor type	ACC_X, ACC_Y, ACC_Z, HUMIDITY, IR, LIGHT, SOUND, ELECTRICITY, MAGNETIC_X, MAGNETIC_Y, , TEMPERATURE, INTERNAL_VOLTAGE
ACTUATOR_TYPE	The specific actuator type	LED
ACTUATOR_PARAM	An actuator parameter	if ACTUATOR_TYPE == LED LED_0_TOGGLE, LED_1_TOGGLE, LED_2_TOGGLE
DATA_TYPE	The data type of sensor readings	SENSED_DATA, THRESHOLD_NOTIFICATION
SYNTHETIC_DATA_TYPE	The synthetic data type of sensor readings. Data aggregation can be set.	NO_SYNTHETIC (RAW DATA), AVERAGE, MIN, MAX
THRESHOLD_TYPE	The threshold type applied on sensor reading	LOWER, BIGGER, TRANSITION
MEMBERSHIP_TYPE	The type of membership operation	UPDATE, ADD, DELETE, RESET
MEMBERSHIP_COUNT	The counter of the membership configuration	no predefined int value
FUNCTION	The type of in-node function computed on the sampled data	ELABORATION_AND_THRESHOLD_STANDARD, ELABORATION_STANDARD, THRESHOLD_STANDARD, AVERAGE, MIN, MAX, THRESHOLD_TYPE_LOWER, THRESHOLD_TYPE_BIGGER, THRESHOLD_TYPE_TRANSITION
TIMESTAMP	Timestamp of the transmitted data	no predefined int value
RESULT	Transmitted data	no predefined int value
MSG_TYPE_TO_ACK	The message type to ack	BM_SENSOR_SCHEDULE, BM_ACTUATOR_SCHEDULE, BM_UNCHEDULE, BM_GROUP_MANAGEMENT
ACK_PARAM	Type of ack	if MSG_TYPE_TO_ACK == BM_SENSOR_SCHEDULE    BM_ACTUATOR_SCHEDULE    BM_UNCHEDULE REQUEST_ID.VALUE if MSG_TYPE_TO_ACK == BM_GROUP_MANAGEMENT MEMBERSHIP_COUNT.VALUE

due to the reception of an unchedule event. Agent planes receive events from the MAPS dispatcher component that is programmed to deliver the events fetched from the agent

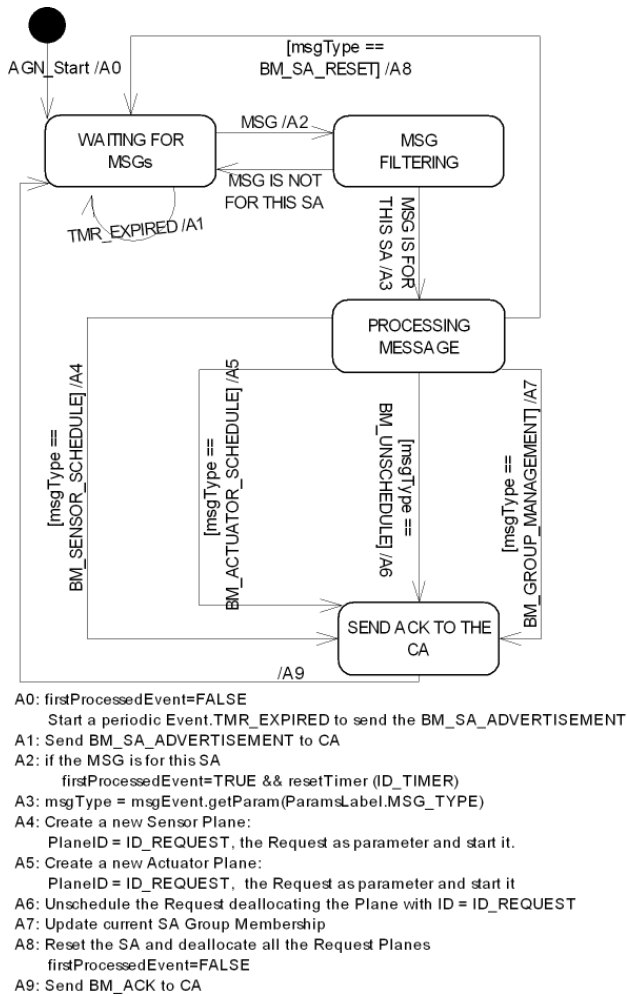


Fig. 5 The SA's Manager plane.

queue to the plane in charge to process them. The dispatcher rules are reported in Table III.

TABLE III.  
DISPATCHER RULES

Event	Plane
BM_SENSOR_SCHEDULE	MANAGER
BM_ACTUATOR_SCHEDULE	MANAGER
BM_UNRESCHEDULE	MANAGER
BM_GROUP_MANAGEMENT	MANAGER
BM_SA_RESET	MANAGER
Event.TMR_EXPIRED <ID, ID_MANAGER_PLANE>	MANAGER
Event.TMR_EXPIRED, <ID, REQUEST_PLANE_ID>	REQUEST
Event.SENSOR_CURRENT_READING, <ID, REQUEST_PLANE_ID>	REQUEST

The Manager plane is reported in Fig. 5. In particular, after agent creation, the Manager plane starts a periodic timer to advertise the agent presence along with its sensor/actuator available functions and waits for an incoming event from the CA. When it receives the first event, the timer is reset. Each received event is filtered against the current SA's group membership. If the filtered event is for the current SA, it is processed according to its type. A more detailed description

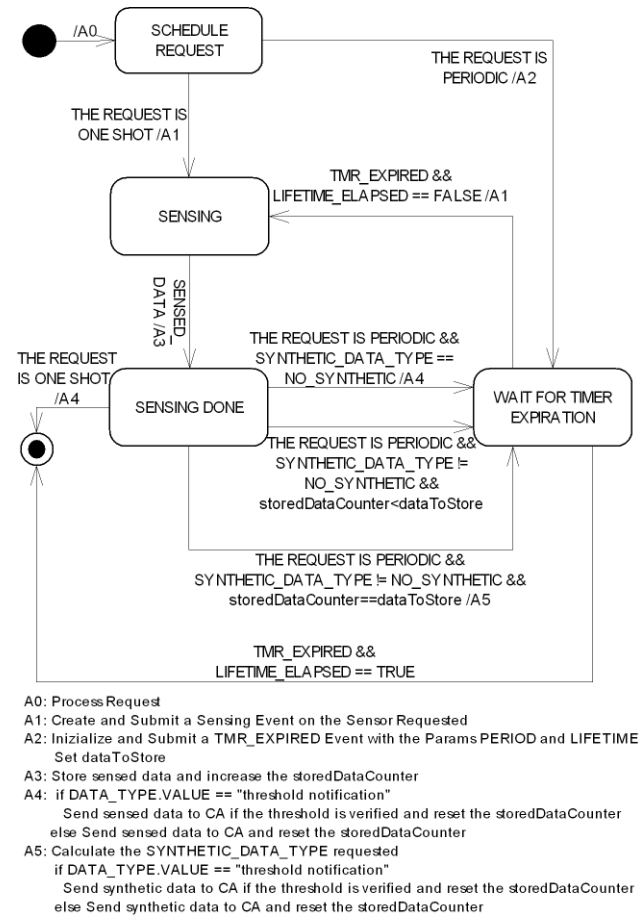


Fig. 6 The SA's Sensing Request plane.

of each action of the Manager plane is provided using a self-explanatory pseudocode (see Fig. 5).

In Fig. 6 the Sensing Request plane is portrayed. This plane is created every time that the agent receives a BM\_SENSOR\_SCHEDULE event. In particular, after the Sensing Request plane creation, the plane creates and submits the MAPS sensing event formalizing the sensing request. A sensing request can be either one-shot or periodic with a given lifetime. The request is scheduled until LIFETIME\_ELAPSED==true after the expiration of the periodic timer driving the submission of the sensing event.

A more detailed description of each action of the Sensing Request plane is provided using a self-explanatory pseudocode (see Fig. 6).

## V. A SYSTEM DEPLOYMENT: MONITORING WORKSTATION USAGE IN COMPUTER LABORATORIES

To show the functionality and effectiveness of the proposed architecture for the management of building indoors, we present an example of system deployment for the monitoring of workstation usage in a computer laboratory or in offices. The wireless sensor network consists of heterogeneous sensor nodes based on Sun SPOTs that are used to collect information about the ambient light (through the standard Sun SPOT light sensor), the user presence (through a

Wieye IR sensorboard [13]) and the electricity consumed by the workstation (through a customization of the ACme electricity sensorboard [14]).

In Fig. 7, the main window of the Building Management GUI is shown. It is organized in five main sections supporting all the functionalities provided by the system:

- *Nodes and Groups Management* sections allows to visualize the nodes of the WSA and configure groups, respectively. By right clicking on the sensors/groups the user can configure sensor/actuator requests to schedule on the nodes;

- *Request* section allows to list details of scheduled requests, display data charts related to the scheduled requests, un-schedule and re-schedule requests;

- *Maps and Graphs* section allows visualizing WSA deployment maps and displaying charts of the data coming from the sensors (examples of charts are shown in Fig. 9);

- *Console* section displays the real-time log of the activity of the system;

- *File and Saving* menu section enables to save data from the system in structured files and load stored files to display them in the GUI.

In Fig. 8, the graphical window for sensor/actuator request scheduling is shown. The window allows setting the parameter of a new request: name, destination (specific nodes or group composition), execution period, lifetime, one shot request or unlimited lifetime flags, action type and related device, possible actuator parameters, requested sensed data possibly filtered by thresholds and/or synthetic data is requested and its type (average/max/min) and eventual threshold parameters can be set.

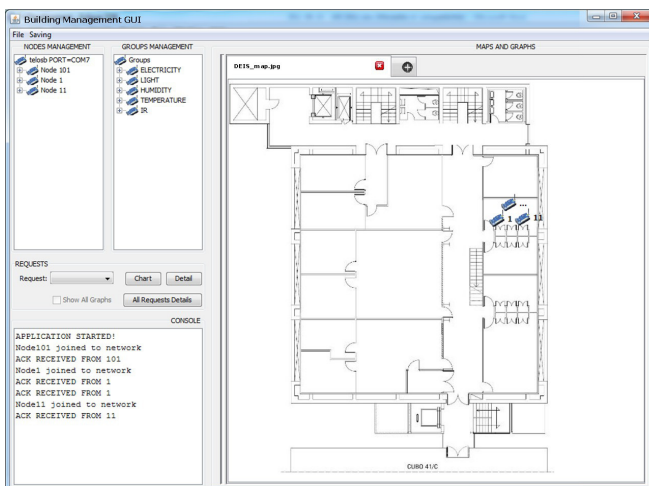


Fig. 7 The Building Management GUI

In the experimental system deployment the following requests were set:

- the raw electricity data (in watt) are gathered every second;
- the average of the ambient light value (in lux) is collected every 10 seconds;
- the max IR sensor value is sensed every minute.

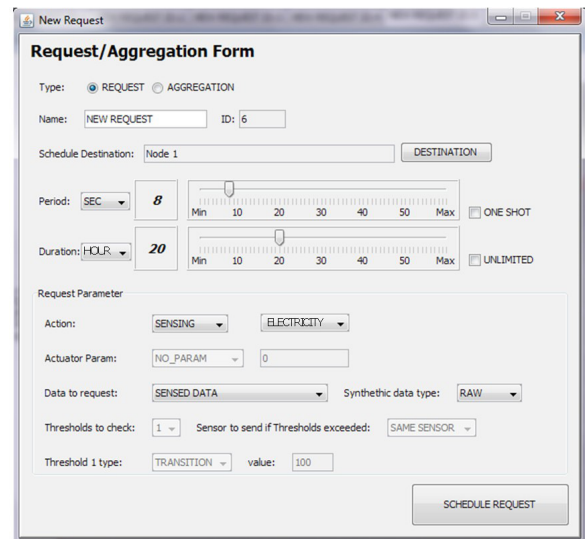


Fig. 8 The graphical window for sensor/actuator request scheduling.

The aim of the experiment was the monitoring of a workstation in a computer laboratory of the Dept. of Electronics, Informatics and Systems to understand its user's behavior. A snapshot of a significant monitoring activity of the duration of 45 min is shown in Fig. 9. In particular, in Fig. 9 two important time instants ( $t1$  and  $t2$ ) are marked. Before  $t1$  the user was working at his workstation by using a word processor application and the ambient light is low as artificial light is off and window curtains were partially closed. Between  $t1$  and  $t2$  the user was out of the office, his workstation automatically switched the monitor off after a period of inactivity and the light was decreasing as late evening was approaching. At  $t2$ , the user came back, started a video streaming application, turned the ceiling lamp on, and after five minutes came out again.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper we have proposed an agent-based architecture for flexible, efficient and embedded sensing and actuation in buildings. Specifically, the distributed software architecture is embedded into both WSA and more capable computing devices (e.g. PCs, smartphones, plug computers). The proposed architecture can be seen as basic middleware for developing intelligent building management systems to achieve the Smart Building concept. Currently the proposed architecture is exploited to monitor the space occupation and energy expenditure in computer laboratories for students to analyze energy consumption patterns with respect to users' behavior so as to semi-automatically implement behavior policies. In the current implementation, BMA and CA are merged into a component-based application implemented through OSGi [15]. Moreover, only one cluster can be deployed. On-going work is aimed at completing the JADE-based implementation of the multi-cluster architecture founded on the BMA and on multiple coordinated CAs. Future work will be devoted to the design of a higher-level agent-based architecture for Smart Buildings atop the proposed ar-



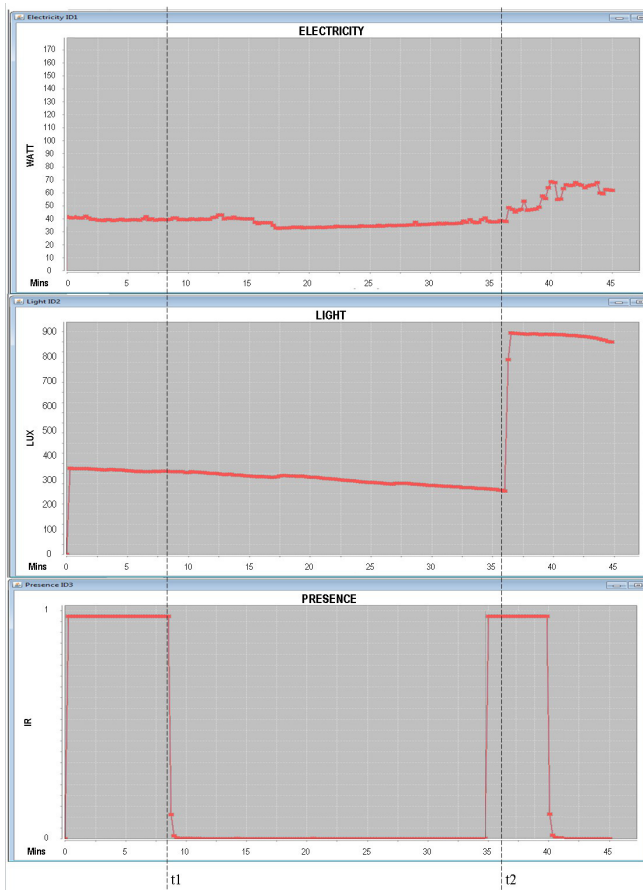


Fig. 9 Real-time data of the workstation usage (workstation consumed power, ambient light and user presence)

chitecture to trade off inhabitants' personal comfort and building energy expenditure.

#### ACKNOWLEDGMENT

This work has been partially supported by CONET, the Cooperating Objects Network of Excellence, funded by the European Commission under FP7 with contract number FP7-2007-2-224053, and by TETRIS – TETRA Innovative Open Source Services, funded by the Italian Government (PON 01-00451).

#### REFERENCES

- [1] Davidsson, P., Boman, M.: A multi-agent system for controlling intelligent buildings. In the Fourth International Conference on MultiAgent Systems, pp. 377-378, Boston (2000)
- [2] Luck, M., McBurney, P., Preist, C.: A manifesto for agent technology: towards next generation computing. *Journal of Autonomous Agents and Multi-Agent Systems*, vol. 9, n. 3, pp. 203-252 (2004)
- [3] Qiao, B., Liu, K., Guy, C.: A Multi-Agent System for Building Control. In the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT '06), pp.653-659, Hong Kong (2006).
- [4] Huberman, B. A., Clearwater, S. H.: A Multi-Agent System for Controlling Building Environments. In the International Conference on Multiagent Systems (ICMAS-95), pp. 171-176, San Francisco (1995)
- [5] Davidsson, P., Boman, M.: Distributed monitoring and control of office buildings by embedded agents. In *Information Sciences—Informatics and Computer Science: An International Journal - Special issue: Intelligent embedded agents*, vol. 171, issue 4, pp. 293-307 (2005)
- [6] Stankovic J.: When sensor and actuator cover the world. *ETRI Journal*; vol. 30, n. 5, pp. 627–633 (2008)
- [7] Guerrieri, A., Ruzzelli, A., Fortino, G., O'Hare, G.: A WSN-based Building Management Framework to Support Energy-Saving Applications in Buildings. In *Advancements in Distributed Computing and Internet Technologies: Trends and Issues*, Al-Sakib Khan Pathan, Mukaddim Pathan, Hae Young Lee, eds, chapter 12, pp. 161-174, IGI Global (2011)
- [8] Bellifemine, F., Fortino, G., Giannantonio, R., Gravina, R., Guerrieri, A., Sgroi, M.: SPINE: A domain-specific framework for rapid prototyping of WBSN applications. *Software Practice and Experience*, vol. 41, issue 3, pp. 237-265 (2011)
- [9] Aiello, F., Fortino, G., Gravina, R., Guerrieri, A.: A Java-based Agent Platform for Programming Wireless Sensor Networks. *The Computer Journal*, vol. 54, issue 3, pp. 439-454 (2011)
- [10] Mobile Agent Platform for Sun SPOT (MAPS), documentation and software at: <http://maps.deis.unical.it/>.
- [11] Bellifemine, F., Poggi, A., Rimassa, G.: Developing multi-agent systems with a FIPA-compliant agent framework. *Softw., Pract. Exper.* vol. 31, issue 2: pp. 103-128 (2001)
- [12] Domanski, J.J., Dziadkiewicz, R., Ganzha, M., Gab, A., Mesjasz M.M.: Implementing GliderAgent – an agent-based decision support system for glider pilots. In *NATO ASI Book*, IOS press, 2011, to appear.
- [13] <http://www.easysen.com/WiEye.htm>
- [14] Jiang, X., Dawson-Haggerty, S., Dutta, P., and Culler, D. Design and Implementation of a High-Fidelity AC Metering Network. In *Proc. of the 8th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN09) Track on Sensor Platforms, Tools, and Design Methods (SPOTS 09)*. 2009.
- [15] Open System Gateway Initiative (OSGi), documents and software at: <http://www.osgi.org>

## Multiagent Distributed Grid Scheduler

Victor Korneev, Dmitry Semenov, Andrey  
Kiselev  
Nii “Kvant”, Moscow, Russia  
Email: {korv@rdi-kvant.ru, sdvbox@gmail.com,  
a\_v\_k@rambler.ru}

Boris Shabanov, Pavel Telegin  
Joint Supercomputer Center RAS, Leninsky pr. 32a,  
Moscow, 119991, Russia  
Email: {shabanov, telegin}@jscc.ru}

**Abstract**—An approach for resource scheduling based on multiagent model with distributed queue is discussed. Algorithms of functioning agents for distributed Grid scheduling are presented

### I. INTRODUCTION

GRID-technologies [1] provide program solutions for creating Grids as certain class of networks. The main requirements for Grid environment in Joint Supercomputer Center RAS (JSCC RAS) are: raising the capacity of aggregate resources by eliminating the situation when some resources are idle while other resources are overloaded, and providing computational resources exceeding capacity of individual systems for execution of large scale parallel programs which can be efficiently implemented on several computational systems (CS).

Each CS contains control computer (CC). Control computers of all CS are integrated by network, they can transfer programs and data and execute remote programs.

Basic modules (BM or nodes) of several CS can be integrated with one or more high-speed networks like Infiniband for data transfers while program execution [2]. For each system in Grid a local batch system operates on CC. Usage of batch system in every CC allows using a single pool of computational modules. Batch system allocates jobs on nodes, terminates, delivers results to users and provides access control.

To include CSs under different administration into Grid without changing software and functioning the middleware software is used which implements functions of management system (MS) of Grid environment. MS of Grid enables submitting user’s jobs to single queue, running jobs on one or several CS using middleware, monitoring Grid environment, providing fault tolerance and access control.

Management systems based on centralized resource sharing model are most studied and implemented. In this model user jobs are submitted to a single queue, which is shared by all processors of parallel system. When the processor is free, it takes a job from the queue, or it is made by a system process, tracking the processors status. This model is used in metadispatcher in GridWay project [5, 6]. However in Grid with multiple CS and significantly different bandwidth between and inside CS it is impossible to achieve in reasonable time the complete and accurate description of the current state of resources and jobs. So, in large GRIDs it is neces-

sary to use distributed metaschedulers based on distributed queuing system model. This paper describes an approach to release Content Addressable Network [4] as multiagent system to resource scheduling based on distributed single queue model and implementation its algorithms of functioning distributed Grid scheduler in JSCC RAS [7, 8]

The paper has the following structure. In second section architecture of Grid management system is presented. The third section discusses a several heuristic algorithms of decentralized job scheduling. The fourth section contains results of experiments on efficiency of proposed decentralized management system of Grid.

### II. DISTRIBUTED MULTIAGENT METASCHEDULER

Submitted user’s jobs must be registered in one of queues of distributed queuing system. Each queue is served by its own agent –local scheduler which accepts one of three decisions for each job: job can be scheduled for execution on resources of one or several CS, can be left in queue for further scheduling or transferred to another queue. Developing a distributed multiagent metascheduler, it is necessary on one hand to enable independent simultaneous scheduling of jobs in different queues by local schedulers, on the other hand usage of Grid resources must be coordinated.

Approach in this article suggests a resolution of this contradiction by allocation of Grid resource domain for each queue for independent scheduling by agent – local domain scheduler. If you allocate resources in separate domains of each CS in Grid and create one extra domain which includes resources of all CSs, then a hierarchy of queues is formed. In this hierarchy it is possible to coordinate the allocation of resources between jobs within the following algorithm of the agent with queues: the transfer of jobs between the queues of the lower level of the hierarchy is possible only through the upper level queue, which is used only for scheduling between lower level queues.

Each dedicated domain is managed by Grid CS component that is CS manager. Manager contains data structure necessary for local scheduler:

- information system (IS) containing resources table of managed domain and description of general Grid environment state;

- queue of jobs to be scheduled.

The basic functional processes of the manager are:



- its own local scheduler, which makes decisions on allocation of jobs on resources or transfer jobs to another queue based on IS data and queue state;

- process supporting current state of IS data to be coherent with IS of other schedulers;

- service processes which provide fault tolerance hierarchy of the managers and information security (protection against unauthorized access to resources).

CS manager, local scheduler of which makes decisions on allocation of jobs to CS resources we will call the manager M1 or 1-st level manager. Manager, scheduler of which allocates jobs between local schedulers, will be called M2 or the 2-nd level manager. M1 managers transfer jobs to batch system queue or to M2 manager. CC of each CS always executes a M1 manager. Number of M2 managers can be one or more depending on required reliability and throughput of management system of Grid. MS managers can run on CS control computers or on additional dedicated computers. Information links between managers form an acyclic graph. Managers are interacting using IP-addresses and port numbers.

Different algorithms can be used in local schedulers and MS managers: from solving optimization problems to heuristic algorithms, that allows taking into account specific heterogeneity of the Grid components.

A protocol for parallel resource allocation by hierarchy structure managers is suggested in [9]. For this protocol it is proved that there are no deadlocks caused by interlocking because of partial simultaneous allocation resources by different managers, and inability to continue jobs due to lack of resources for a job without releasing of resources by another job.

Hierarchy organization of CS managers in Grid allows:

- Ensure absence of deadlocks during distributed execution of scheduling algorithms and resources allocation;

- Take into account the specifics of managing heterogeneous objects, combining similar objects (CSs, domains) under control of single manager. Combining CSs to domain can be done using different similarity criteria: architecture, hardware and software platform, administration policy, geographical location, ownership of organization, etc.;

- Control MS managers in the same domain by single organization providing their support, and use scheduling algorithms common for given domain;

- Reduce number and variety of control object types for each manager, this simplifies formulation and implementation of management decision and reduces the uncertainty of complex multiprogramming case, determining and fixing the number of parameters for the higher level.

### III. ALGORITHMS FOR DISTRIBUTED SCHEDULING.

Let us consider that job can be executed on any CS from Grid and development of control solutions in the local manager uses two job parameters: required number of computational nodes and required time. In practice more parameters are used [8], but these two are sufficient for understanding the idea.

The following characteristics of jobs and batch systems are used for description of computational resources:

- Area of user job, it equals to product of requested number of nodes and requested time;

- Summary area of jobs on certain CS, it equals to sum of areas of jobs, which are queued or executed on this CS;

- Load, it equals to the ratio of the summary area of jobs on CS to the total number of computational nodes, which can execute user's programs. This characteristic describes mean time that nodes will be busy executing jobs;

- Upper load bound of CS, it limits CS load.

Current difference between bound and load is total area of jobs, which can be submitted to batch system on given CS.

M1 managers submit user jobs to batch system queue without exceeding upper load bound. It should be noted that in some cases, the batch system imposes a restriction on the maximum time user jobs. So, area of jobs which can be submitted is limited either by difference between bound and current load or by value equal to product of requested number of nodes to maximum allowed execution time for given batch system. Local scheduler takes into account this feature and will not schedule job with area exceeding this limit. Changing upper load bounds one can redistribute jobs between batch systems queues and CS managers queues.

As it is shown in [9], with appropriate objective function like deviation of load values of computational node from mean load value in local neighborhood, it is possible to minimize of the objective function through the development of local management decisions for CS load balancing.

Scheduling strategy is based on principle of making sub-optimal decision in a coherent interaction between MS managers. Managers, which are distributed over a network, make local decisions, forming parts of global decision. Decision on resource allocation for user job is made only by MS manager controlling the given resource as it has most accurate information about allocated resource, that allows to make decision on the basis of actual data.

Let us explain using Fig. 1 how system of Grid managers functions. One can see Grid consisting of 7 CSs, each of CC executes manager M1i,  $i = 1, \dots, 7$ . Each M1i manager stores in its resource table a row contains value received from the batch system of corresponding CSi is recorded.

M2 managers have a set of enumerated logical channels ports, which link them with M1 and M2 managers. For each port M2 manager stores a row in resource table with values of areas of jobs which can be submitted to batch systems, and their M1 managers can be reached by acyclic graph of logical channels from the given M2 manager. E.g., for M24 manager resource table row for port 1 contains information about CS3 and CS4, in row for port 2 information about CS1 and CS2, in row for port 3 information about CS5, CS6 and CS7.

Termination of jobs, failure of CS and communication channels, recovery of CS and communication channels, connection of new CS to Grid result corresponding changes in resource tables. In general, each M2 manager contains complete information about Grid resources, but different managers have their own tables.

Jobs queue of MS manager uses FIFO strategy. Each time when resource table is changed or after a specified time interval manager attempts to schedule jobs from its queue. First of all list of CS in resource table is analyzed. In accordance with the applicable scheduling algorithm a CS with sufficient number of nodes for job execution is selected. If there are no appropriate CS found in resource table of MS manager, search is made by records corresponding to adjacent MS managers.

When requested resources found in system job is transferred to corresponding adjacent MS manager. When there are no resources available job goes to the end of MS manager queue, and manager allocates the next job. The set of all MS managers' queues forms a single global queue for aggregated resources of the Grid.

It is obvious that the transfer of jobs between managers may result infinite residence in queues of managers. To prevent this, a label is assigned to job, which allows or prohibits job rescheduling. If rescheduling is prohibited then local scheduler transfers job to manager which resources are allocated for job execution. Of course, this transfer is possible only if there is place in that M1 manager, otherwise job waits for possibility of transfer. Setting label prohibiting rescheduling may be result of excess of limit of reschedulings, or expiration of time interval of job stay in Grid.

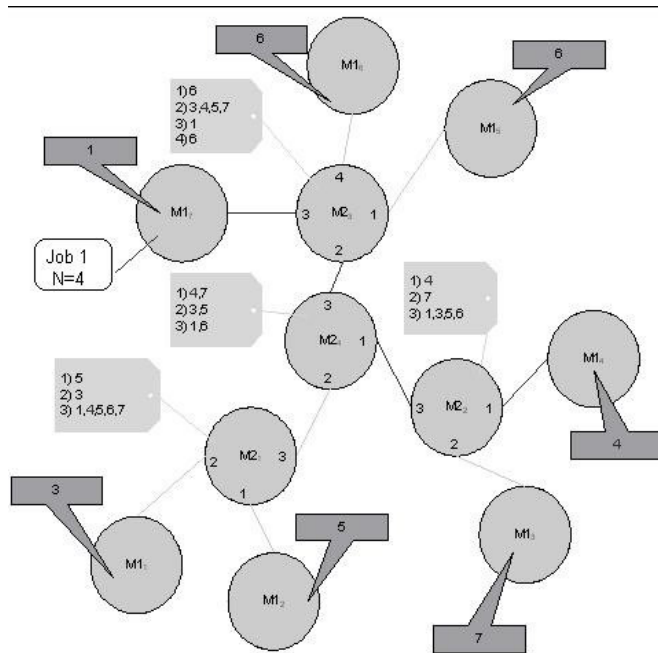


Fig. 1. Grid managers

The same label is used for scheduling parallel job on nodes of different CS. Parts of one job allocated on different CS are represented as separate labeled jobs, this guarantees reception of these jobs in the assigned CSs.

In some studies, particularly [10], of scheduling jobs in the tree-shaped queue algorithms leveling the number of jobs in queues are used. However, it seems that algorithms, taking into account the load of Grid resources, would be more appropriate.

The article investigates heuristic algorithms based on principles of minimal load and minimal sufficiency [3].

According to the minimal load principle a job is sent to the CS with least difference of upper load bound and load of CS, i.e. job is allocated to less busy system or in case of equal load to the system with minimum required number of units of a free resource. This algorithm provides dynamic leveling of computational load for CSs, so we call it balancing algorithm.

Alternative is usage of minimal sufficiency principle: job is scheduled to CS with sufficient for an immediate start job number of units of a free resource. Two modifications of minimal sufficiency algorithm were investigated. In the first modification jobs were assigned to Grid resources in turn, in the second modification jobs were assigned when possible without regard to priority: as soon as required number of computational nodes became free in cluster CS, the first job able to be executed there was job was extracted from managers' queue. This allows to better load of CS, but order of jobs is violated.

Let us explain functioning of MS managers using example of scheduling on Grid in Fig. 1. In these examples allocation strategy with minimal sufficiency is shown.

Example 1. Job running on CS7 requires 4 resource units. Job will be transferred from M17 to M23 and later according to resource table to M24, then to M22 and finally to M14. M14 manager allocates job at CS4 which it controls. Job allocation on CS4 will change resource tables of managers M17, M23, M24 and M14 (number 4 will be excluded from all resource tables). Due to the fact that level 2 managers have aggregate information, decision on resource allocation can be taken only by level 1 manager or adjacent level 2 manager.

Example 2. Job requiring 8 resource units, running on any of the M1 managers will be suspended by the adjacent M2 until there is sufficient number of free nodes.

In managers' resource tables predicted values of free Grid resources for a given scale of time in the future can be formed. Managers can make decisions basing on these predicted values.

#### IV. INVESTIGATION OF EFFICIENCY OF DISTRIBUTED SCHEDULING ALGORITHMS.

A set of experiments was performed: once formed test jobs flow was fed to the Grid containing of two CS called next and neo, and operating under developed Grid environment [8].

Test flow consists of 2 parts, 25 jobs each and represents typical jobs flow for MVS-1000 system. In a separate series of experiments test the flow of jobs ran independently on each CS in the Grid.

In further experiments these flows were fed to the Grid management system simultaneously. In Table results of experiments are presented. In experiments A1-A5 balancing algorithm with different values of bound parameter.

In experiments B1 and B2 scheduling algorithm based on minimal sufficiency is used. In experiment B1 data jobs were

allocated in turn, in B2 jobs were allocated whenever possible, without regard to priority.

TABLE I.  
EXPERIMENTAL RESULTS

	CS	Number of jobs	Wall time
independent	next	25	02:43:54
	neo	25	05:3:58
A1, no limit	next	42	03:47:35
	neo	8	03:36:18
A2, bound = 80	next	41	03:21:11
	neo	9	03: 18:29
A3, bound = 70	next	26	03:31:46
	neo	24	03: 04:30
A3, bound = 60	next	32	04:58:01
	neo	18	02:32:39
A3, bound = 50	next	29	03:14:21
	neo	21	04:23:49
B1 in turn	next	33	03:38:32
	neo	17	03:54:27
B2, no priority	next	28	02:56:17
	neo	22	03:06:31

In experiments A1 with infinite upper load bound jobs were allocated to resources without staying in managers' queues. It is needed to note that if jobs flow is distributed to two identical systems the way that there is the same load level at initial moment then times of execution of parts of jobs flow will vary because of differences of real execution times. On the "next" CS part of jobs flow was executed faster and CS was idle while the other CS was executing the rest of its jobs. This can be seen in results of experiments with infinite bound. However, it should be noted that due to more rational distribution of jobs (CS with more resources received greater part of flow) it was possible to reduce maximum of jobs processing times compared to execution of the flows on independent CSs.

In Table 1 one can see trend deterioration in quality of scheduling jobs flow with bound less than 80. With big bound values results of experiments tend to results with infinite bound. When bounds are lower it happens that jobs with small number of requested nodes and high requested time contribute significantly to the CS workload/ thus increasing load bound, while there are free nodes in CS. In cases like this idle time of nodes is high resulting summary time of executions.

It should also be noted that for all experiments MS [8] demonstrated stable operation both in normal mode and in high load mode: all jobs were scheduled and executed.

#### REFERENCES

- [1] Foster I., Kesselman C., Tsudik G., Tuecke S. A security architecture for computational grids // Proc. 5th ACM Conference on Computer and Communications Security. San Francisco: ACM Press, 1998. 83–92.
- [2] Корнеев В. В. Вычислительные системы. М.: Гелиос АРВ, 2004.
- [3] Корнеев В. В., Киселев А. В., Семенов Д. В., Сахаров И. Е. Управление метакомпьютерными системами // Открытые системы. 2005. № 2. 11–16.
- [4] Lee J., Keleher P., and Sussman A. "Decentralized resource management for multi-core desktop grids," in 24th *IEEE International Parallel & Distributed Processing Symposium*, Atlanta, Georgia, USA, 2010.
- [5] Богданов С. А., Коваленко В. Н., Хухлаев Е. В., Шорин О. Н. Метадиспетчер: реализация средствами метакомпьютерной системы Globus. Препринт ИПИМ № 30. Москва, 2001.
- [6] GridWay Metascheduler: Metascheduling Technologies for the Grid. URL: <http://gridway.org>.
- [7] Савин Г. И., Корнеев В. В., Шабанов Б. М., Телегин П. Н., Семенов Д. В., Киселев А. В., Кузнецов А. В., Вдовикин О. И., Аладышев О. С., Овсянников А. П. Создание распределенной инфраструктуры для суперкомпьютерных приложений// Программные продукты и системы. 2008. № 2. 2–7.
- [8] Руководство программиста грид (<http://www.jscc.ru/informat/grid1.zip>).
- [9] Корнеев В. В. Архитектура вычислительных систем с программируемой структурой. Новосибирск: Наука, 1985. (<http://andrei.klimov.net/reading/1985.Korneev.-Arkhitektura.vychislitel'nykh.sistem.s.programmiruemoi.strukturoi.zip>)
- [10] Houle M., Symvonis A., Wood D. Dimension-exchange algorithms for token distribution on tree-connected architectures // J. of Parallel and Distributed Computing. 2004. № 64. 591–605.

# Tuning Computer Gaming Agents using Q-Learning

Purvag G. Patel

Department of Computer Science  
Southern Illinois University Carbondale  
Carbondale, IL 62901  
Email: purvag@siu.edu

Norman Carver

Department of Computer Science  
Southern Illinois University Carbondale  
Carbondale, IL 62901  
Email: carver@cs.siu.edu

Shahram Rahimi

Department of Computer Science  
Southern Illinois University Carbondale  
Carbondale, IL 62901  
Email: rahimi@cs.siu.edu

**Abstract**—The aim of intelligent techniques, termed *game AI*, used in computer video games is to provide an interesting and challenging game play to a game player. Being highly sophisticated, these games present game developers with similar kind of requirements and challenges as faced by academic AI community. The game companies claim to use sophisticated game AI to model artificial characters such as computer game bots, intelligent realistic AI agents. However, these bots work via simple routines pre-programmed to suit the game map, game rules, game type, and other parameters unique to each game. Mostly, illusive intelligent behaviors are programmed using simple conditional statements and are hard-coded in the bots' logic. Moreover, a game programmer has to spend considerable time configuring crisp inputs for these conditional statements. Therefore, we realize a need for machine learning techniques to dynamically improve bots' behavior and save precious computer programmers' man-hours. We selected *Q-learning*, a reinforcement learning technique, to evolve dynamic intelligent bots, as it is a simple, efficient, and online learning algorithm. Machine learning techniques such as reinforcement learning are known to be intractable if they use a detailed model of the world, and also require tuning of various parameters to give satisfactory performance. Therefore, this paper examine Q-learning for evolving a few basic behaviors viz. learning to fight, and planting the bomb for computer game bots. Furthermore, we experimented on how bots would use knowledge learned from abstract models to evolve its behavior in more detailed model of the world.

## I. INTRODUCTION

SINCE the advent of game development, game developers have always used game AI for developing the game characters that could appear intelligent. All the games incorporate some form of *game AI*. It can be in the form of ghosts in the classic game of PAC man or sophisticated bots in first-person shooter(FPS) games such as Counter-Strike and Half-life[1]. Human players while playing against or with computer players, which are used to replace humans, have a few expectations such as predictability and unpredictability, support, surprise, winning, losing and losing well[2].

The goal of agents in *game AI* is similar to the machine used in the Turing test, which humans cannot identify whom they are answering to. [3] organized a game bot programming competition, the BotPrize, in order to find answers to the simple questions such as, can artificial intelligence techniques design bots to credibly simulate a human player?, or simple tweaks and tricks are effective? Competitors submit a bot in order to pass a "Turing Test for Bots". It is relatively easy to

identify bots in the system. Some general characteristics used to identify bots were [3]:

- Lack of planning
- Lack of consistency - 'forgets' opponents behavior
- Getting 'stuck'
- Static movement
- Extremely accurate shooting
- Stubbornness

Primary reason for exhibiting such behavior is that these bots are usually modeled using finite-state machines(FSM) and programmed using simple conditional statements, resulting in a very predictable bot to an experienced game player[4]. These bots play fixed strategies, rather than improving as a result of the game play. Moreover, designing such bots is time consuming because game developers need to configure many crisp values in their hard-coded logic. Resultant, bots lose their credibility as a human being.

Believability plays a major role for the characters in books and movies, even if it is fiction. Similarly, believability and credibility also plays a major role in video games especially with the artificial characters. However, it is more challenging to design an artificial character for a video game compared to the characters in the books and movies. Characters in video game need to constantly interact with the humans, and adapt their game play without any guidance. There are wide varieties of situation to cope with, and present variety of challenges such as real time, incomplete knowledge, limited resources, and planning[5]. Therefore, the ability to learn will have advantage in increasing the believability of the characters, and should be considered as an important feature[6].

Methods of machine learning could be used effectively in games to address the limitations of current approaches to building bots. The advantages of using a machine learning technique to improve computer games bots' behaviors are:

- it would eliminate/reduce the efforts of game developers in configuring each crisp parameter and hence save costly man-hours in game development, and
- it would allow bots to dynamically evolve their game play as a result of interacting with human players, making games more interesting and unpredictable to human game players.

This paper investigates the use of Q-learning, a type of reinforcement learning technique(RL), to improve the behavior

of game bots. Q-learning is relatively simple and efficient algorithm, and it can be applied to dynamic online learning. We developed our own game platform for experimentation, a highly simplified simulation of FPS games like Counter-Strike. Bots, which uses learning algorithm, in all our experiments are modeled as terrorist agents. Goals of these agents include killing counter-terrorists, planting the bomb on critical locations, or surviving till the end of the game play.

While machine learning techniques can be easy to apply, they can become intractable if they use detailed models of the world, but simplified, abstract models may not result in acceptable learned performance. Furthermore, like most machine learning techniques, RL has a number of parameters that can greatly affect how well the technique works, but there is only limited guidance available for setting these parameters. This paper set out to answer some basic questions about how well reinforcement learning might be able to work for FPS game bots. We focused on the following three sets of experiments:

- *learning to fight*: testing if and how well bots could use RL to learn to fight, and how the resulting performance would compare to human programmed opponent bots,
- *learning to plant the bomb*: instead of rewarding bots for fighting, what would happen to bots' behavior if they were rewarded for accomplishing the goal of planting bombs, and
- *learning for deployment*: if bots initially learn using abstract states models (as might be done by the game designers), how does initializing their knowledge from the abstract models help in learning with more detailed models.

The ultimate goal of these experiments is to evolve sophisticated and unpredictable bots which can be used by game developers and provide unprecedented fun to game players.

The rest of the paper is organized as following. Section II provides a background on the FPS game of Counter-Strike and the model of the bots use in such games. Related work in presented in section III. Details of the simulation and methodology are described in section IV. In Section V results of experiments are presented. Conclusion and future work are discussed in section VI.

## II. COUNTER-STRIKE AND BOTS

### A. Counter-strike

Counter-Strike is a team-based FPS which runs on the Half-life game engine. Counter-Strike is one of the most popular open source computer games available in the market, and is played by thousands of players simultaneously on the Internet. Our initial attempt was to conduct the experiments on a modification of the actual game, but due to improper documentation and complexity of the available source code we developed a scale down simulation of the game. Nevertheless, most of the discussions and the experiments conducted in the paper are inspired from this game.

One of the typical game playing scenarios in Counter-Strike is bomb-planting scenario. There are two teams in the game,



Fig. 1. GAME MAP

namely terrorist and counter-terrorist. The terrorist aims to plant the bomb while counter-terrorist aims to stop them from planting the bomb. In the processes, sub-goal of each team is to eliminate all the opponents by killing them. Figure 1 shows a standard map of Counter-strike called DE\_DUST. On the map, two sites labeled A, and B are bomb sites where a terrorist plant the bomb. On the contrary, a counter-terrorist defend these bomb sites and if a bomb gets planted by the terrorists, then counter-terrorists attempt to defuse the bomb before it explodes. In the beginning of each round, both the teams are located at designated locations on map. For example, the position labeled CC in figure 1 is **counter-terrorist camp**, which is the location of counter-terrorists at the beginning of each round. Similarly, the position marked by label TC in figure 1 is the **terrorist camp** for terrorists. Once the round begins, they start advancing to different location in map, simultaneously, fighting with each other on encounters and thereby trying to achieve their respective goals.

We simulated very similar game environment, as presented in this section, with an exception of game map; our game map is relatively simple

### B. Bots in computer games

Counter-Strike uses the *bots*, also called Non-player Characters(NPCs), to simulate human players in the teams to give the 'illusion' of playing against actual game players. Bots play as a part of the team and achieve goals similar to humans. Currently, bots used in Counter-Strike are programmed to find path, attack opponent players, or run away from the site if they have heavy retaliation, providing an illusion that they are intelligent. Similar species of bots are used in many other FPS games, such as Half-Life, Quake and Unreal-Tournament, with similar methods of programming. Usually, bots in computer games are modeled using a FSM, as shown in figure 2, where rectangles represent possible states and leading edges show transitions between states. This is just a simplified representation of actual bots, where many more such states exist with more complicated transitions. A FSM for bots is quite self explanatory. First the bot starts by making initial decisions viz. game strategies, buying weapons, etc. and then starts searching for enemies/opponent. After the opponent is spotted, it makes a transition to attack state in which he fires

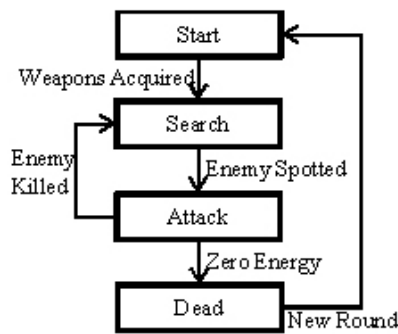


Fig. 2. A PROTOTYPICAL FSM FOR A BOTS

the bullets at the opponent. A Bot may kill an opponent. In that case, it will again start searching for other opponents. Also, a bot could be in any of the above mentioned states and might get killed by the opponent.

There are inherent flaws in using classic FSM model for bots. All transitions/rules needs to be hardcoded in bots logic, and programmers have to spend time configuring these parameters. For example, a rule for agent's attacking behavior based on agent's speed and energy is shown in algorithm 1. Parameters such as energy, distance of enemy, etc., needs to be configured, for which programmers run a large number of time-consuming simulations. Moreover, the use crisp value

---

#### Algorithm 1 Hardcoded rules

---

```

1: if agent.speed ≥ 4% & agent.range ≥ 4% then
2:   attack()
3: else
4:   flee()
5: end if
  
```

---

in decision making makes these bots predictable. As a result, it makes a game less interesting and less believable to an experienced game player and eventually may lose interest in the game.

### III. RELATED WORK

There are several recent attempts for using machine learning techniques, especially reinforcement learning, for developing a learning bot. Reinforcement Learning (RL) is a machine learning technique where an agent learns to solve problems while interacting with the environment[7].

[8] suggested a learning algorithm to investigate the extent to which RL could be used to learn basic FPS bots behaviors. Their team discovered that using RL over rule-based systems rendered a number of advantages such as: (a) game programmers used minimal code for this particular algorithm, and (b) there was a significant decrease in the time spent for tuning up the parameters. Also, the applied algorithm was used to successfully learn the bots behaviors of navigation and combat, and the results showed that by changing its planning sets of parameters, different bots personality types could be produced.

Thus, the paper suggested how an agent can learn to be a bot with the help of RL in shooter games[8].

[9] demonstrates several interesting results using RL algorithm, Sarsa, for training the bots, yet again signifying the effectiveness of such learning techniques. They designed a testbed 2D environment with walls creating partitions on the map. Preliminary experiments were conducted for the bots to learn the Navigation task. Based on the rewards, the bots learn to minimize collisions, maximize distance travel, and maximize number of items collected. After certain number of iterations the bots started receiving greater number of rewards signifying that the bots have learned a positive desired behavior. On manipulating the values of the rewards, the bots learn different behavior. For example, increasing the penalty for collision the bots would learn to remain away from wall, simultaneously ignoring collectible items near the wall. Although, the bots did not meet the industry standard their experiment demonstrated that with right parameters the behavior of the bots can be controlled. They also demonstrated the bots learning Combat behavior. Results were similar to navigation task, whereby the rewards for the bots increased after certain number of iterations proving that bots are learning fruitful behavior. Nevertheless, these experiments demonstrate successful use of reinforcement learning to a simple FPS game [9].

Primary reason for using testbest instead of actual game in this this paper and in [8][9] was to reduce c.p.u. cycle and difficulty in dealing with complexities involving in coding for actual game. Nevertheless, several efforts include the use of reinforcement learning in actual video game [10][11]. [10] designed a bot, RL-DOT, for a Unreal Tournament domination game. In RL-DOT, the commander NPC makes team decision, and sends orders to other NPC soldiers. RL-DOT uses Q-learning for making policy decision[10]. There are efforts to develop a NPC that would learn to overtake in racing game like The Open Racing Car Simulator (TORCS). It is suggested that, using Q-learning sophisticated behaviors, such as overtaking on straight stretch or tight bend, can be learned in a dynamically changing game situation[11].

N. Cole et. al. argues that to save computation and programmer's time, the game AI uses many hard-coded parameters for bot's logic, which results in usage of enormous amount of time for setting these parameters [4]. Therefore, N. Cole et. al. proposed the use of genetic algorithm for the task of tuning these parameters and showed that these methods resulted in bots which are competitive with bots tuned by a human with expert knowledge of the game. Related work was done by S. Zanetti et. al. who used the bot from the FPS game Quake 3, and demonstrated the use of Feed Forward Multi-Layer Neural Network trained by a Genetic Algorithm to tune the parameters tuned by N. Cole at. al.[12].

Widely used AI techniques include RL, neural networks, genetic algorithm, decision tree, Bayesian networks, and flocking [13][14].



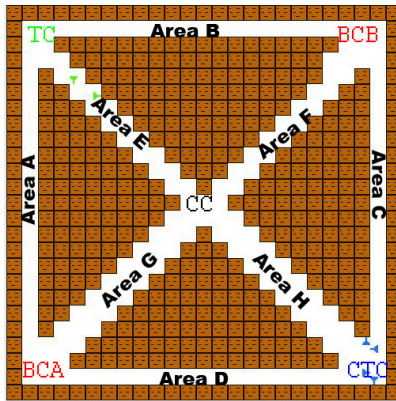


Fig. 3. MAP DIVIDED INTO AREAS

#### IV. APPROACH

##### A. Simulation Environment

We developed a scaled down abstraction of Counter-Strike in Java, and simulated the bots in this environment. The miniature version of a Counter-Strike 2D map is shown in figure 3. Herein, bricks are visible which form the boundary of the map and act as obstacles for the agents. There are two kinds of agents, blue and green, which navigate through the map formed by the bricks. Each of the blue and green agents imitates the behavior of terrorists and counter-terrorists respectively from Counter-Strike. Moreover, there are two sites labeled BCA and BCB in figure 3 which are similar to the bomb sites in Counter-Strike. In figure 3, sites labeled TC and CTC are green and blue base camps respectively. Before the start of a game, we specify the number of each type of agents. The green agents' goal is to plant the bomb in one of the sites (either BCA or BCB) or kill all blue agents. Blue agents defend these sites and kill all green agents. This provides us with an environment similar to a classic FPS game, where two autonomous sets of agents fight with each other.

##### B. Methodology

We investigate the Q-learning algorithm for improving the behavior of the *green agents*, while keeping the blue agents' behavior static.

The static Blue agents run a simple algorithm (Algorithm 2), in which if they spots a green agent they shoots a new missile, else they continue moving on map according to *plan*. A plan is a sequence of locations such as TC, BCB, etc. Each agent navigates on the map in the order specified in the plan. Agents randomly choose from six such manually configured plans.

---

##### Algorithm 2 Static Blue Agents

---

```

1: if s.hasError() then
2:   attack()
3: else
4:   move according to plan
5: end if

```

---

An agent using Q-learning learns a mapping for which action he should take when he is in one of the states of the environment. This mapping can be viewed as a table, called a *Q-table* -  $Q(s,a)$ , with rows as states of the agent and columns as all the actions an agent can perform in its environment. Values of each cell in a Q-table signify how favorable an action is given that an agent is in a particular state. Therefore, an agent selects the best known action, depending on his current state:  $\arg \max_a Q(s,a)$ .

Every action taken by an agent affects the environment, which may result in a change of the current state for the agent. Based on his action, the agent gets a reward (a real or natural number) or punishment (a negative reward). These rewards are used by the agent to learn. The goal of an agent is to maximize the total reward which he achieves, by learning the actions which are optimal for each state. Hence, the function which calculates quality of state-action combination is given by :  $Q : S \times A \rightarrow R$

Initially, random values are set in the Q-table. Thereafter, each time an agent takes an action; a reward is given to agent, which in turn is used to update the values in Q-table. The formula for updating the Q-table is given by:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_t \times [r_{t+1} + \gamma \arg \max_a Q(s_{t+1}, a_t) - Q(s_t, a_t)],$$

where  $r_t$  is reward at any given time  $t$ ,  $\alpha_t$  is the learning rate and  $\gamma$  is discount factor.

It is necessary to formulate the problem in terms of states and actions for applying q-learning algorithm. Hereby, figure 3 shows experimentation in which we have divided the map into 8 areas. For simplicity we plan on starting with eight *areas* with which will form the state: set  $A = \{A, B, \dots, H\}$  for the agents. The agents will select randomly one of the six manually configured plan. Agents' second state is *plans*: set  $P = \{0, 1, \dots, 5\}$  of size six. In addition, the agents use *enemy present* of size two as state: set  $E = \{0, 1\}$ , which signifies whether opponents are present in the individual agents' range(0) or not(1).

Hence, we have state space of 96 states ( $A \times P \times E$ ). An agent will be in one of these states at any period in time, will perform one of the following actions: Attack(0) or Ignore(1).

The game being highly dynamic, it is infeasible to predict the agents' future state and determining whether an action currently executed is fruitful or not (rewards) in order to recalculate utilities. Therefore, the an agent's reward is known in a future state. Hence, utilities of a state  $s_t$  is updated when an agent is in a state  $s_{t+1}$ . In state  $s_{t+1}$  we can determine an agent's rewards for the action attempted in the state  $s_t$  and  $s_{t+1}$  is treated as the future state for updating the utilities for state  $s_t$ . Similarly, if suppose an agent fired a missile then we cannot determine the rewards until state  $s_{t+x}$ , where  $x > 1$ , is the state when missile actually hits a blue agent or explode without hitting anyone. In such a scenario, we ignore the intermediate states between state  $s_{t+x}$  and  $s_t$ , and directly update values of state  $s_t$  based on values of state  $s_{t+x}$ .

We used an 'exploration rate( $\epsilon$ )' as probability for choosing



the best action. Suppose, if the exploration rate of agent is 0.2, then an agent will choose action with greater utility value with probability of 0.8 and any other actions with probability of 0.2. Usually, low exploration rates, between 0.0 to 0.3, are used. Therefore, an agent selects an action with greater utility most of the time and  $\epsilon$  determines the probability of exploring other actions.

---

**Algorithm 3** Dynamic Green Agents
 

---

```

1: currentState = getCurrentState()
2: prevState = getPreviousState()
3: action = selectAction(currentState)
4: if action = 0 then
5:   attack()
6: else
7:   ignore()
8: end if
9: updateQtable(prevState, currentState, rewards)
10: setPreviousState(currentState)

```

---

Algorithms 3 summarize the algorithms used by green agent wherein it is learning the best action i.e. attack or ignore(0 or 1). In algorithm 3, during the first two steps agent retrieves its current state and previous states. Then the agent selects an action based on its current state. Next, if the action is 0(mnemonics for attack action), then agent shoots a missile, else it just continues according to its plan. Finally, the agent updates its Q-table based on current and previous state, and stores the current state as previous state to use for the next iteration.

## V. EXPERIMENT

We examine designing agents using Q-learning which can learn different behaviors based on the rewards they are getting. Key issues with any learning techniques is setting various parameters, which in case of Q-learning are learning rate( $\alpha$ ), discount factor( $\gamma$ ), and exploration rate( $\epsilon$ ). Therefore, our preliminary experiments are to determine the right combination of parameters.

Provided a flexible simulation environment, inspired from environment in the Counter-strike game, varieties of experiments are possible. Albeit, bots in Counter-strike, as with most other FPS games, need to learn basic behaviors such as combat and planting the bomb. Therefore, we experimented with rewards function in order for bots to learn these basic behaviors i.e. learning to fight and plant the bomb. Apart from this, a model of a actual game will have a large number of states. Moreover, as the number of states grows in Q-learning the size of the Q-table grows; simultaneously slowing down the speed of learning. Hence, we propose to train the bots with a small number of abstract states which are a superset of the more detailed states used by actual game. Finally, these learned utility values are distributed among large number of detailed states in the actual game and an agent continue online learning thereafter.

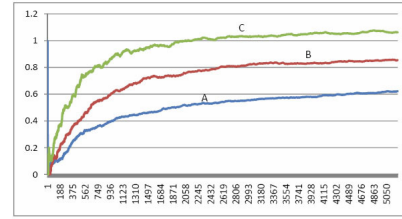


Fig. 4. LEARNING CURVES FOR LEARNING TO FIGHT: (A)  $\alpha = 0.10$ ,  $\gamma = 0.90$  AND,  $\epsilon = 0.1$ , (B)  $\alpha = 0.30$ ,  $\gamma = 0.95$  AND,  $\epsilon = 0.1$ , AND (C)  $\alpha = 0.56$ ,  $\gamma = 0.99$  AND,  $\epsilon = 0.1$

Evaluation of these agents is based on the maximum fitness green agents would reach against static blue agents. In all the experiments, fitness of agents is measured by the ratio of number of rounds won by green agents against the number of round won by blue agents. By round, we mean a single game cycle where one team wins and another loses. Green agents won by killing all the blue agents or planting the bomb. Blue agents won the round by killing all the green agents. For each experiment we modified reward function so that agents can learn differently. The remaining section provide detailed experimental setup and results.

### A. Learning to fight

For the first experiment, we wanted to train the agents to learn combat behavior. The agents had only two actions to choose from: Attack or Ignore opponents. In the attack action, the agents shoot a missile, while in the ignore action agents just ignore the presence of a nearby enemy and continue their current plan.

In order for the agents to learn that shooting a missile is costly, if it is not going to be effective, we gave small negative reward of -0.1 if agent shoots a missile. If the agent gets hit by an enemy missile, the agent gets a small negative reward of -0.2. Agents were given a large positive reward of +10 if they kill an enemy agent. All the values of Q-table were set to zero before training.

Figure 4 shows the learning curve for three different combination of  $\alpha$ ,  $\gamma$ , and,  $\epsilon$ . Similar curves are also observed for remaining combination of parameter. Unexpectedly, the combination with  $\alpha = 0.56$ ,  $\gamma = 0.99$  and,  $\epsilon = 0.1$  produced agents with maximum fitness. A high value of  $\gamma$  signifies that future states are playing an important role in determining the selection of current actions. Reinforcement learning technique tend to produce curves with high fluctuations if learning rate is high. But, in our experiment we observed a very steady learning curve, as seen in figure 4. Exploration rate of 0.1 is normal for this type of experiments. Notice that the curve crosses the fitness level of 1.0 around 3000 rounds and then curve becomes steady showing very little improvement and reaching an asymptote of 1.0620. Fitness value greater than 1 here means that agents are outperforming static agents.

Hence, with this experiment we were able to evolve agents which successfully learned a combat behavior.

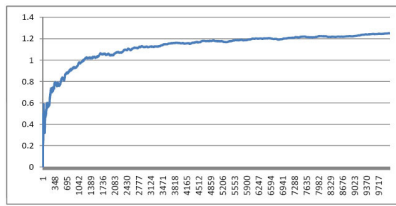


Fig. 5. LEARNING CURVE FOR LEARNING TO PLANT THE BOMB

### B. Learning to plant the bomb

Next experiment was to train the agents for planting the bomb. In the first experiment, we used expert knowledge that killing opponents is a better action. Now, we want to explore whether the bots can evolve to learn similar behavior (or better) if they are focused on planting the bomb. Again, the agents had two actions to choose from: attack and ignore. But, now planting the bomb action was part of ignore action. An agent would plant the bomb if it is in one of the bomb sites as a part of ignore action, else it will continue on its current plan.

The agents did not receive any rewards for killing enemy agents. Instead, rewards are given only when an agent plant the bomb: +4 reward for each unit of bomb planted. Similar to the first experiment, the agents are given -0.1 rewards for shooting a missile and -0.2 rewards for being hit by an enemy missile. Also, all the values of Q-table are set to zero before training.

The best fitness ratio is for the combination of  $\alpha = 0.94$ ,  $\gamma = 0.99$  and,  $\epsilon = 0.20$ . Again, the learning rate is high even though the graph shown in figure 5 is quite smooth. The discount rate remains the same. Herein, exploration rate is high due the fact that agents are not receiving any rewards for killing the opponent agents. Yet, in order to successfully plant the bomb an agent has to kill the opponent agents otherwise it will get killed by them. In order learn to kill blue agents it should actually fire a missile more often than in the previous experiment. The utilities require more time to propagate than before because the only location agents are getting positive rewards are in the two corners (bomb sites). It also evident from the figure 5 that it is required to run this simulation for more number of rounds.

Table I shows the Q-values learned from the experiment. A state in the table is represented by a triplet  $[PAE]$  where  $P = 0, 1, \dots, 6$  is the plan number,  $A = A, B, C, \dots, H$  is the area and,  $E = p$  if enemy is present or  $n$  if enemy not present. Values in remaining two columns: attack and ignore are the utility value of taking a particular action. Agent selects the action with greater utility value with probability of  $1-\epsilon$  else selects the other section.

Few rows in the Q-table have value 0 or very small value like -0.1, for example state: (0 H N). These are states where agents were not trained because agents rarely used these areas while using a particular plan. Similar is the case with all the

TABLE I  
Q-TABLE FOR PLAN 0 AND 4

State	Attack	Ignore	State	Attack	Ignore
0 A n	14.72	19.33	4 A n	-0.10	0.00
0 A p	2.10	1.83	4 A p	0.00	0.00
0 B n	26.89	35.14	4 B n	-0.10	0.00
0 B p	41.38	32.92	4 B p	-0.10	0.00
0 C n	33.67	33.65	4 C n	-0.10	0.00
0 C p	41.02	33.10	4 C p	-0.10	0.00
0 D n	1.25	1.19	4 D n	0.00	0.00
0 D p	-0.09	1.89	4 D p	0.00	0.00
0 E n	18.54	18.47	4 E n	-0.10	0.00
0 E p	34.33	23.51	4 E p	-0.10	0.00
0 F n	19.58	19.57	4 F n	-0.10	0.00
0 F p	19.99	19.98	4 F p	-0.10	0.00
0 G n	1.47	1.58	4 G n	0.00	0.00
0 G p	2.26	1.82	4 G p	0.00	0.00
0 H n	0.00	0.00	4 H n	0.00	0.00
0 H p	0.00	0.00	4 H p	0.00	0.00

states of plan 4 and 2(not shown in table) because green agent while using plan 4 and 2 never encounters the enemy agent. Therefore, all the enemy present state are having values zero. Remaining states have values -0.10 because every time a agent shoots a missile, it receives -0.1 reward. We can infer from the table that, for the majority of remaining states, agents learned the following two behaviors:

- To ignore or plant the bomb if enemy is not present. There is no need to shoot a missile if no enemy is present i.e. utility value of ignore action is greater than attack action, for example (0 A n).
- To attack if enemy is present. An agent learned to attack even though it is not getting any rewards for doing so i.e. utility value for attack action if greater then utility of ignore action if enemy is present, for example (0 A p).

We observe the second behavior due to the fact that rewards propagate from the state where agents plant the bomb to the state where agents shoot a missile. Also, there is a small difference in utility values of both the action in the majority of the states because the same rewards(for bomb planting) also propagate for ignore action. But as agents are able to plant more bomb units only if they killed an enemy agent in a previous state and hence, indirectly learned that killing is required to plant the bomb. Nevertheless, there are few states where even if enemy is present and utilities for ignore state are greater. These are the states with areas away from bomb sites so propagation for rewards might require more training or ignore action might actually be a better option to choose(to run away) because the ultimate goal is to plant the bomb.

The bots generated with this experiment outperformed the static bots and learned to attack even though they are not receiving any direct incentives. However, we cannot compare the results of the experiments in the previous section i.e. learning to fight with this experiment. In current experiment, i.e. learning to plant the bomb, definite goal of the agents is to plant the bomb. We modified plans to achieve this behavior such that final location of each plan is one of the bomb sites (BCA or BCB). This action affects the outcome of the game

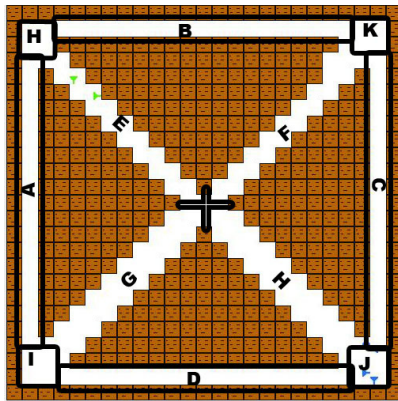


Fig. 6. 12 AREAS

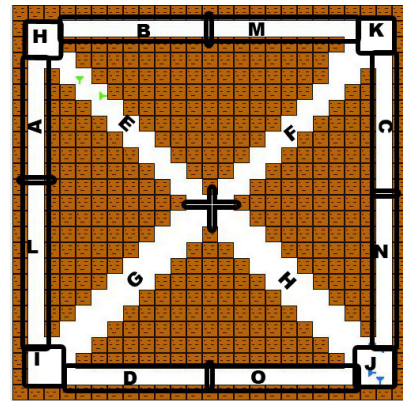


Fig. 7. 16 AREAS

because in the former case agents were moving randomly, but now they have a definite goal of going to a particular location and planting the bomb.

C. Learning for deployment

Above experiments showed that with current technique the competitive bots can be produced but, the bots with random initial values cannot be supplied with the actual games. So the bots need to be partially trained before they are actually supplied with a game. Also, training with more number of states, as in the case with actual game, also takes considerably more amount of time. In this experiment bots are trained for a small number of rounds with the agent having fewer states and then use those Q-values to train the agents with large number of states. Rewards and other settings for the experiment is similar to the experiment in section V.B and used parameter combination  $\alpha = 0.94$ ,  $\gamma = 0.99$  and,  $\epsilon = 0.20$  for experimentation which produced the bots with maximum fitness.

Until now, in all the experiments the map is divided into 8 areas. For subsequent experiments, the map is first divided into 12 areas and then into 16 areas. For a map divided into 8 areas the size of the Q-table is 96 which will increase to 144 for 12 areas and 192 for 16 areas. Figure 6 and Figure 7 shows the divided map for 12 areas and 16 areas respectively. Note that the new divisions are subset of atleast one division from the original map (8 area).

Agents are trained on the map with 8 areas for 500, 1000, 1500, and 2000 rounds and the utilities for the agents are stored. These stored utilities are then used as initial utilities for the agents to be trained on the map with 12 and 16 areas. Here, numbers of states for agents with 12 and 16 areas are more than the agents with 8 areas. Therefore, the utilities for new states are set equal to utilities of old states from which they are generated. For example, area I in figure 6 was part of area A in figure 3 therefore, utilities of all the states with area I is set equal to utilities of state with area A. For comparison purpose, we also ran simulation for 12 and 16 areas stating with all zeros in q-table (without fetching initial q-values from 8 areas), and called them the results with 0 initial utility value.

TABLE II  
EXPERIMENTAL RESULTS FOR LEARNING FOR DEPLOYMENT

Round in initial training with 8 area	Fitness with 12 areas	Fitness with 16 area
0	1.7488	1.6887
500	1.8299	1.7163
1000	1.8023	1.6931
1500	1.8327	1.7306
2000	1.7823	1.7337

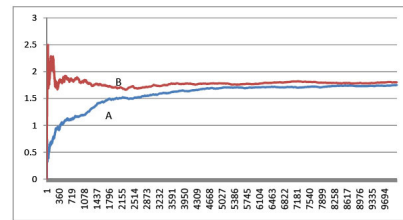


Fig. 8. LEARNING CURVE WITH 12 AREAS, WHERE (A) IS THE CURVE WITHOUT ANY INITIAL TRAINING AND, (B) IS THE CURVE WITH 1000 INITIAL TRAINING FROM 8 STATES

Table II shows the highest ratio achieved by agents in each setup, i.e. for 12 and 16 areas. Similar values are observed for different initial training which signifies that number of initial training does not play a significant role in determining agent’s ultimate performance. The interesting fact about this experiment is visible in graphs of figure 8 and figure 9. Both the figure shows the comparison graph between learning curve of agents with 0 initial utility values (A) and utilities from trained samples for 500 or 1000 rounds with 8 areas as initial utility values(B). Though both the graph almost converge at the end; notice that, initial fitness of the agents with initial training is high and remains high throughout. This result shows that initial training provided to the agents with fewer states is useful and the agents exhibit a sudden jumps to certain fitness levels and remain at those level with minor increment. The initial ups and downs seen in both graphs are due to the fact that our evaluation criteria is ratio of green wins versus blue wins which keeps on fluctuating due to less samples.

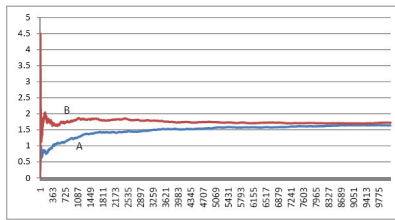


Fig. 9. LEARNING CURVE WITH 16 AREAS, WHERE (A) IS THE CURVE WITHOUT ANY INITIAL TRAINING AND, (B) IS THE CURVE WITH 500 INITIAL TRAINING FROM 8 STATES

TABLE III  
COMPARISON OF Q-VALUES

State	Initial At- tack	Initial Ig- nore	Final At- tack	Final Ignore
1 G p	283.44	313.69	940.71	886.04834
1 H p	0	288.57	997.46	916.6944
1 A p	318.27	327.12	1027.06	936.54224
3 C p	378.20	449.48	1343.67	1334.412

Finally, table III shows comparison between few selected samples from the Q-table before and after training in 12 areas for 20000 rounds. Before training, the utility values of the attack action, when an enemy was present, is less than the value of the ignore action. But after the training an agent's utility value of the attack action is greater than value ignore action. Here, an agent evolved to learn to shoot missiles at opponents when one is present. This demonstrated that an agent is capable of learning better and different actions than the initial utilities supplied from small number of abstract states.

We can conclude from this experiment that when partially learned values from abstract states are used as initial value for detailed states, provided a fitness boost to the agents. The agents thereafter remains at the competitive fitness level against the static agents and continue learning a better behavior.

## VI. CONCLUSION

It is evident from the results that the evolved bots are able to outperform their static counterparts. Moreover, by using the Q-learning algorithm bots were able to learn various behaviors based on the rewards they are getting. Also, having trained a bot with less number of states we are able to generate a competitive bot for large number of states. In this learning-based approach, the bots learned to attack or ignore the opponents based on their current state which comprises of location, plan, and enemy present or not. No hard coding of any parameters is required for the bots. The bots selected the actions based on its utility values which is updated dynamically. Hence, by using this approach we can not only reduce the efforts to engineer the

hard-coded bots, but also evolve them dynamically to improve their behavior and adapt to human player strategies.

Furthermore, the performance of the agents can be improved by devising a method through which agents can learn to select the plan. Currently, a plan is selected randomly after each round; instead, a plan based can be selected based on the past experience of the bots. The bots need to select a plan which in past has been proved to be most fruitful. This behavior can be achieved by interpreting their current utilities for using a particular plan. Along with this, a confluence of various learning technique can be used to improve the learning speed of agents. For example, after each round we can use genetic algorithm to mutate utility values in Q-table among the agents in order to generate a better population. Currently, all the five agents are learning separately without any integration among one another.

Most importantly, we need to test this approach in real simulation of the game, which was our initial attempt. Until then, we cannot judge the actual performance of these agents. Ultimately, the bots need to play against the human players. Although the dynamic bots were tested against static bots, yet human behavior is very unpredictable.

## REFERENCES

- [1] D.M. Bourg and G. Seemann. AI for Game Developers. *O'Reilly Media, Inc., 2004*
- [2] B. Scott. AI Game Programming Wisdom by Steve Rabin. *Charles River Media, Inc., 2002, pp. 16-20.*
- [3] P. Hingston. A new design for a Turing Test for Bots. *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on , 2010, pp. 345-350.*
- [4] N. Cole, S. J. Louis, and C. Miles. Using a Genetic Algorithm to Tune First-Person Shooter Bot. *In Proceedings of the International Congress on Evolutionary Computation, 2004, pp. 139-145 Vol. 1.*
- [5] A. Nareyek. Intelligent Agent for Computer Games. *In Proceedings of the Second International Conference on Computers and Games, 2000*
- [6] F. Tenc, C. Buche, P. D. Loor, and O. Marc. The Challenge of Believability in Video Games: Definitions, Agents Models and Imitation Learning. *GAMEON-ASIA'2010, France, 2010, pp. 38-45.*
- [7] R. Sutton and A. Barto. Reinforcement Learning: An Introduction. *The MIT Press Cambridge, Massachusetts London, England, 1998.*
- [8] M. McPartland, and M. Gallagher. Learning to be a Bot: Reinforcement learning in shooter games. *4th Artificial Intelligence for Interactive Digital Entertainment Conference, Stanford, California, 2008, pp. 78-83.*
- [9] M. McPartland, and M. Gallagher. Reinforcement Learning in First Person Shooter Games. *IEEE Transactions on Computational Intelligence and AI in Games, 2011, Vol. 3.1., pp 43-56. .*
- [10] H. Wang, Y. Gao, and X. Chen. RL-DOT: A Reinforcement Learning NPC Team for Playing Domination Games. *IEEE Transactions on Computational Intelligence and AI in Games, 2010, Vol. 2.1, pp. 17-26.*
- [11] D. Loiacono, A. Prete, P. Lanzi, and L. Cardamone. Learning to overtake in TORCS using simple reinforcement learning. *2010 IEEE Congress on Evolutionary Computation (CEC), 2010, pp. 1-8*
- [12] S. Zanetti and A. Rhalibi. Machine Learning Techniques for FPS in Q3. *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology, 2004, pp. 239-244.*
- [13] D. Johnson and J. Wiles. Computer Game with Intelligence. *Australian Journal of Intelligent Information Processing Systems, 7, 2001, pp. 61-68.*
- [14] S. Yildirim and S.B. Stene. A Survey on the Need and Use of AI in Game Agents. *In Proceedings of the 2008 Spring simulation multiconference, 2008, pp. 124-131.*



# Developing intelligent bots for the Diplomacy game

Sylwia Polberg

Warsaw University of Technology  
 Email: sylwia.polberg@gmail.com

Marcin Paprzycki, Maria Ganzha

Polish Academy of Sciences  
 Email: firstname.lastname@ibspan.waw.pl

**Abstract**—This paper describes the design of an architecture of a bot capable of playing the Diplomacy game, to be used within the *dip* framework—a testbed for multi-agent negotiations. The proposed *SillyNegoBot*, is an extension of the *SillyBot*. It is designed to be used in the level-1 negotiations (as defined within the *dip* framework) taking place during the Diplomacy game.

## I. INTRODUCTION; THE DIPLOMACY GAME

THE DIPLOMACY board game was created in 1954 by Allan B. Callhamer. The game takes players back to Europe from the beginning of the 20th century. The aim of each player is to eliminate opponents and gain control over the continent, by any means necessary. To gain more influence, players can negotiate, create alliances, lie, break alliances and/or promises, etc. Outcome of the game depends only on players' decisions and behavior (there is no element of chance). More information about the game can be found in the Wikipedia [1], the Diplomacy Archive [2], and the rule book [3].

Since the game depends only on strategy and negotiations it became of interest to AI researchers. There were many projects that tried to create a successful *bot* for the Diplomacy game. References to most of them are available on the DAIDE project website [4]. Unfortunately, most of them are currently halted, or already dead [5]. This is often not only because the topic is hard, but also because it requires knowledge outside of computer science. Moreover, in many cases source codes of bots were not made available, and hence the total contribution to the state of knowledge was relatively small. Currently, the most active project is pursued in the Spanish Artificial Intelligence Research Institute (IIIA). This project aims not only at developing the *dip* framework (an environment, in which agents will be able to compete against each other and humans), but also to develop negotiating bots (however, these are not available yet).

The aim of this contribution is to summarize the rationale behind and describe the architecture of the *SillyNegoBot*, which is to be capable of level-1 negotiations (within the *dip* framework). The *SillyNegoBot* is an extension of the *dip* 0-level Diplomacy playing *SillyBot* (see, [6]).

### A. Rules of the Diplomacy game

Let us start by briefly describing the rules of the Diplomacy game (for further details, see the rule book [3]). In order to explain how the game proceeds, we need to first introduce some basic notions. In the standard game there are 7 powers (players) — Austria, England, France, Germany, Italy, Russia and Turkey. The map of Europe is divided into 56 land and

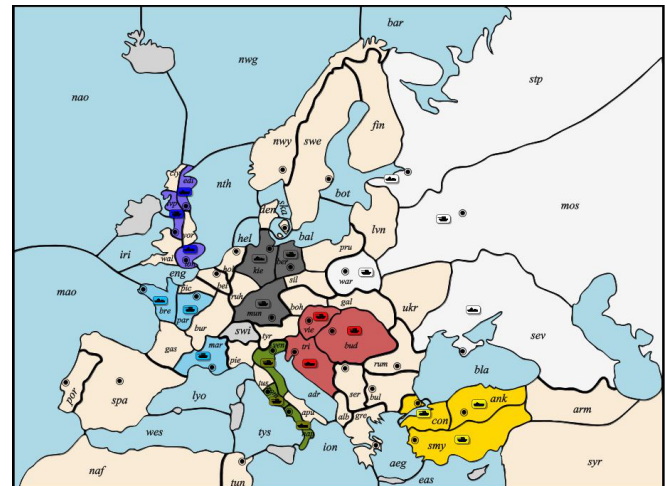


Fig. 1. Sample initial map from the dipgame [7].

19 sea provinces. Each province is further split into a certain number of coastal, sea and land regions. Among provinces, 34 are marked as Supply Centers. Controlling them allows a player to expand his army. Homes are supply centers that each player receives at the very start of the game. Every power has 3 of them, apart from Russia—in exchange for the longest battle front it controls 4 home provinces. Winning condition of the game is the control of at least 17 supply centers.

There exist two types of units: fleets and armies. Armies travel through the land, while fleets travel through seas and coasts. A unit of given type can only be built in a home province that contains a region that allows its movement (e.g. fleets can be only build in coastal regions). Only one unit can occupy a province at a time. All units are equally strong—during a “fight” only their number matters—more units supporting each-other beats less units supporting each-other, and results in capturing a given province.

### B. Orders

During each phase of the game (see, below), an order has to be issued for each unit. Depending on the specific phase, an order can be either of the ones presented on Figure 2.

Both, in the case of a movement and attacking, unit with bigger support “wins” and moves to the desired region (captures it). When the support size is equal, units do not move. Interestingly, most of known diplomacy bots do not implement the convoy order. Note that the game-server executes orders

Order	Abbr.	Description
Hold	<i>HLD</i>	unit stays where it is
Move	<i>MTO</i>	units moves to an adjacent (free) region, or attacks it
Support	<i>SUP</i> ( <i>supporting holding unit</i> ) <i>SUPMTO</i> ( <i>supporting moving unit</i> )	for unit <i>A</i> to support a stationary unit <i>B</i> , they have to be neighbors if unit <i>B</i> is moving, for <i>A</i> to support it, <i>A</i> has to reside next to <i>B</i> 's destination
Convoy	<i>CVY</i>	fleet order used to move an army from one coast to another
Retreat	<i>RTO</i>	move a unit that needs to be evacuated (escape) to a (free) adjacent region
Dislodge (disband)	<i>DSB</i>	used if a <i>Retreat</i> is not possible (dislodges the unit from the board)
Build	<i>BLD</i>	build a unit; if there are more resource centers than units, and the home province is free
Waive	<i>WVE</i>	sent if it is impossible, or undesired, to build a unit
Remove	<i>REM</i>	if there are more units than source centers player chooses unit to remove

Fig. 2. List of Diplomacy orders.

“simultaneously.” In other words, time of their arrival is of no importance to their results.

1) *Game play*: Game proceeds in turns (representing years). Each turn is separated into five phases that differ in purpose and possible orders; in summary:

- *Spring*  
The first movement season. Orders *MTO*, *HLD*, *SUP* and *CVY* can be issued. During this phase, one usually moves units to areas (s)he wants to annex soon, or build up defenses against incoming enemies.
- *Spring retreats*  
Only *RTO* and *DSB* orders can be issued. If during the Spring phase a units is attacked and loses the fight (i.e. the attacker had greater support), this unit has to move out of the region. If there are no such regions, the *RTO* order has to be issued. Otherwise, given army or fleet is automatically dislodged from the board.
- *Fall*  
Season very similar to the Spring. The only difference is that after it ends, “newly” occupied supply centers are

being annexed and become usable in the Winter phase.

- *Fall retreats*  
Exactly as Spring retreats.
- *Winter*  
At the end of the year players can expand their army (or fleet). In order to issue a *BLD* order, three conditions have to be met:
  - Province one wants to build a unit in, has to be an unoccupied home.
  - Province has to contain a region compliant with the type of unit to be build.
  - Amount of controlled supply centers has to be greater than the number of owned units.

Usually one can build at most 3 units during one Winter—reason is obvious, there are only 3 homes (4 for Russia). Should a player have more units than supply centers, (s)he has to remove some of them in order to restore the balance (the *REM* order is issued for specific units). When no removal is needed or no builds are possible/desired, the *WVE* order is issued.

## II. THE *dip* FRAMEWORK

In 2009, A. Fabregues and C. Sierra created the *dip* framework [8], which allows one to create Diplomacy playing bots and test their abilities against other bots (and humans). All libraries needed to write dip-bots can be found within the dip website [7]. The dip framework uses the *dip language* [8]. Currently, out of its 10 levels, we are interested in level-0 (*Order Issuing*), and level-1 (*Negotiations*). In Figure 3 we summarize how the latter is structured. Interestingly, the dip language not only differs from the earlier DAIDE language, but there is no 1-to-1 relation between them. This difference does not affect the game rules—only the level 0 communication is defined in the rulebook [3], while negotiations are independent and are not defined in the original board game. During negotiations, agents make proposals to other agents,

$$\begin{aligned} \mathcal{L}_1 &::= \text{propose}(\alpha, \beta, \text{deal}) | \text{accept}(\alpha, \beta, \text{deal}) | \text{reject}(\alpha, \beta, \text{deal}) | \\ &\quad \text{withdraw}(\alpha, \beta) \\ \text{deal} &::= \text{Commit}(\alpha, \beta, \varphi)^+ | \text{Agree}(\beta, \varphi) \\ \varphi &::= \text{predicate} | \text{Do}(\alpha, \text{action}) | \varphi \wedge \varphi | \neg \varphi \\ \beta &::= \alpha^+ \\ \alpha &::= \underline{\text{agent}} \end{aligned}$$

Fig. 3. dip level-1 language.

and accept or reject ones they receive. In order to end a negotiation (message exchange) with a given power, an agent sends a withdraw message (similar to a “bye” after a finished chat). Note that this ends only a specific negotiation, (other negotiations between these agents may ensue later). Agents can “talk” about agreeing on truthfulness of some facts—such as keeping a peace, or committing to do something. By an action we understand the 0-level orders.

Note that the predicates used at the level-1 are typically limited to the following offers (and thus define the scope of our work):

- PCE(power+)—peace between a group of powers
- ALY(power+, power+)—alliance between a group of powers against some other group of powers.

### III. SILLYBOT AGENT

Let us now briefly describe the SillyBot (a predecessor to the SillyNegoBot), which is capable of playing using the level 0 language. Out of many designs we have tested, this one proved to be the most successful one. For further details please refer to [6].

#### A. Bot design

Key elements of the design of the SillyBot were (see, also Figure 4):

- Phase Graders—responsible for placing *Requests*.
- Board Analyzer—consisting of Board Assistant, Union Manager and Threat Assistant.
- Request System—mechanism evaluating *Requests*.
- Heuristic—responsible for picking a *Request* solution, prioritizing *Requests*, and *Request* filtering.

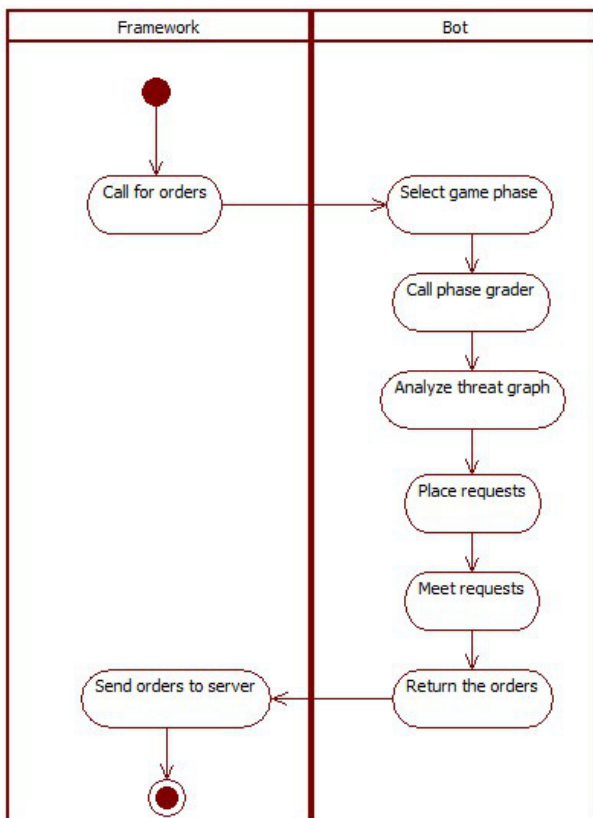


Fig. 4. Structure of the SillyBot agent.

To make the description more understandable, let us start from the *Request*, which is a simple entity that means “I want

someone to :an order: in a given province.” Each level 0 order has a respective *Request*. For example, let us assume that there is a province worthy annexing; we will want to issue an MTO order (for a specific unit). Therefore, we will place a MTO *Request* for that area. Moreover, each *Request* has to contain a list of possible units that can perform the desired order within a given province. When the SillyBot prepares actual orders to be issued, the *Graders* generate *Requests*, which are collected and sent to the *Request Meeter* (it is an element of the system responsible for satisfying *Requests*).

To handle the *Request* evaluation, an algorithm based on a tree structure has been developed. Here, the root of the tree is empty. We take first *Request* and expand the root—each node represents an actual sequence of orders satisfying the *Request*. There can be many ways to fulfill a *Request*, hence multiple nodes. For example, if we want to occupy Paris and have armies in Burgundy and Gascony, our root will have two children—one containing MTO(X,BURAMY,PARAMY), and the other MTO(X,GASAMY,PARAMY). Order sequences with more than just a single order can occur only in the case of a SUP or SUPMTO *Requests*. This should be obvious, as we can need help of more than just one unit. Created nodes are later filtered for collisions—we will not want to have an order issued for a unit that was earlier assigned a different order. Another filter is to establish whether satisfying the *Request* with orders in the node will later give us the chance to find a node fulfilling the next *Request*. This basically means that if among a group of nodes some leave at least one option for the next *Request*, we will not bother with the rest and apply the heuristic only to them. To each remaining node layer, another heuristic is applied that attempts to pick the best node. Its focal point is how “shared” node is—e.g. how many other *Requests* the unit we have picked can satisfy. With already sorted and grouped *Requests*, we aim to fulfill as many as we can. Here, result matters most, the fact who will go where is of secondary importance. In the final stage of the algorithm, data from all chosen nodes is merged into a single order list and returned to the *Graders*. There are several *Graders*—one for the building phase, one for retreating phase and three for the movement phase. This is due to the fact that during the start, middle and end of the game we may want to apply different *Request* placing, as priorities can change.

Results of initial tests have been reported in [6]. They were promising enough to proceed with further developments.

### IV. THE SILLYNEGOBOT

The SillyNegoBot consists of 9 agents (for the reason behind its structure, see below). It is built on top of the SillyBot, to handle the level 1 dip language. Specifically, the framework agent (further referred to as, the *Mother*) extends the SillyBot and uses its functionalities to handle the level 0 functions (recall that the language levels are inclusive—every higher level contains previous ones). It adds functions and classes necessary from the point of view of framework, in order to handle the technical side of negotiation messages. Moreover it contains means of communicating with the remaining 8



agents. Logically, the SillyNegoBot consists of the Emotional System and Rational System. Although they share multiple functionalities, they are kept separated in order to be turned on and off for testing (e.g. to represent the emotion—rationality conflict). The rationale for including emotions in the design of the SillyNegoBot has been discussed in detail in [5].

For the design and implementation of the SillyNegoBot, we have decided to use an already existing knowledge driven model—JADEX, that is based on the BDI approach [9]. As discussed in [5], we have decided to create a bot consisting of eight separate (sub)agents: the *Mother*, six *Ambassadors*, and an *Arbitrator*. In the current version we have extended the system by the *MotherMessenger* agent, responsible for transferring all level 1 reasoning to JADEX. This encapsulates the negotiation-oriented reasoning (thus allowing to change the negotiation “brain” without touching the remaining parts of the system). After discussing the purpose behind all agents we will describe the communication between them and the complete architecture of the SillyNegoBot (for more details, see, also [5]).

#### A. Agents

1) *Mother*: The *Mother* agent (a slightly modified SillyBot) has two main tasks—the game (dip) server communication, and specification of dip level 0 orders. Furthermore, it initializes the platform for other agents and launches the *MotherMessenger*. Finally, it controls passing messages between the *MotherMessenger* and the server.

2) *MotherMessenger*: This agent, upon receiving appropriate message from the *Mother*, launches all other agents. It is responsible for distributing messages, e.g. *Ambassador of Germany* might not be interested in messages meant for the *Ambassador of Italy*. At the end of the game it is responsible for platform shutdown.

3) *Ambassadors*: The six *Ambassadors* are the main “thinking” part of the system. Note that, using a single agent to handle all “reasoning,” would result in a single, very large, belief base, while requiring only a part of it for each decision. Moreover, one would have to keep track of six communication “channels” at the same time (recall that there are six opposing powers). This would complicate the implementation, and could lead to decreased efficiency. Therefore, we decided to use six *Ambassadors* (JADEX agents; one per opponent). The *Ambassadors*, facilitate level 1 messaging, and reasoning based on their exchange (Negotiations). Note that, due to this separation, we have created a layer of negotiations internal to the bot; e.g. one *Ambassador* may need a piece of knowledge owned by another *Ambassador*. To handle such situation, and to deal with conflicting recommendations originating from independently working *Ambassadors*, we have created the *Arbitrator* agent.

4) *Arbitrator*: The six *Ambassadors* will sometimes require acceptance of a goal or an action. For example, when all (or at least some) *Ambassadors* decide to attack powers they are assigned to, a decision has to be made, which attack should materialize (while others are dropped or postponed).

To deal with this problem we introduced another agent—an *Arbitrator*—responsible for mediating conflicting recommendations obtained from the *Ambassadors*. Obviously, this task could have been placed within the *Mother* agent (or the *MotherMessenger*), but this would result in putting too many unrelated functions within a single agent. Therefore, we have followed a “single agent per major functionality” paradigm of agent system design.

#### B. Architecture of the SillyNegoBot

Here, we start from key elements that are independent of the JADEX platform. Next, we describe how we use the BDI model. For further details concerning the reasoning behind each element, please refer to [5].

1) *Messages*: Messages passed between agents will be encapsulated using FIPA standards. Their mapping is presented in Figure 5. In the current version of the bot we have simplified some messages and used the functionality taken away from them to create new ones. Internal messages in the system can be of the following types:

- *START* (*START((POWER, AGENT), (POWER, AGENT),...)*)— sent by the *MotherMessenger* to all *Ambassadors* (as the first message). It and defines the power that the given *Ambassador* is responsible for. The *Ambassador* agent is supposed to extract its own assignment and remember the powers of others in order to make the internal communication easier.
- *INFO* (*INFO(ORDER), INFO(MESSAGE)...*)— exchanged between all agents in the bot to inform about data, e.g. the *MotherMessenger* agent passes an incoming message to the *Ambassador*. It can also be the case that an *Ambassador* sends to the *Mother* (through *MotherMessenger*) information as to who is allied with whom (level 0 requires such data for proper threat estimation).
- *PASS* (*PASS(MESSAGE)*)—used to ask the *MotherMessenger* and the *Mother* agents to pass messages from the *Ambassador* to the given power. Note that a direct communication is impossible without writing our own server.
- *FETCH* (*FETCH(POWER), FETCH(ORDER), FETCH(PREDICATE),...*)—asks given agent to provide specific information (beliefs, emotions and so on) concerning a given opponent, relations between powers, given order or belief, feeling, etc.
- *ARB ID* (*ARB ID(AGENT)*)—contains the identifier of the *Arbitrator* agent.
- *BOARD* (*BOARD(GAME)*)—message sent from the *Mother* to the *MotherMessenger*. State of the board is read and transformed into *OWN* predicates, which are later passed to the *Ambassadors*. If it is received for the first time, Power information is read and used to launch other agents in the platform. It requires no FIPA encapsulation, as it sent through a socket.
- *END*—end of the game, shut the platform down.

- *NEXTPHASE* (*NEXTPHASE(TIMELINE)*)—message marking a new phase in the game.

Internal message	FIPA encapsulation
START	INFORM
INFO	INFORM
PASS	INFORM
FETCH	REQUEST
ARB ID	INFORM
END	INFORM
NEXTPHASE	INFORM

Fig. 5. FIPA encapsulation of internal messages.

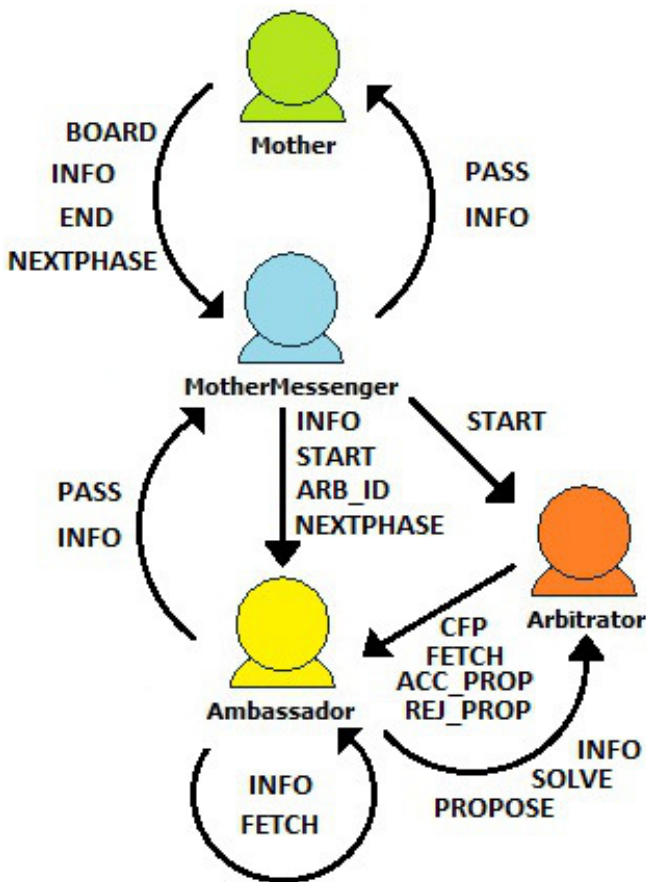


Fig. 6. Message flow diagram.

We drop the internal parsing in the following messages:

- *SOLVE* (*SOLVE(MESSAGES)*)—sent by the *Ambassador* to the *Arbitrator*, in order to receive acceptance of moves. In response the *Arbitrator* sends out a *CFP* message, which should be answered with *PROPOSE* messages that *Ambassadors* want to send to their respective powers. In other words—an initial message for conflict resolution.
- *ACCEPT PROPOSAL*, *REJECT PROPOSAL*—to be used as an answer for the *PROPOSE* message.

2) *Emotions*: Emotions impact the probability of other powers actions—how will they react to negotiations (offers and counter-offers), whom will they help, and who not. Simple analysis of “what action benefits one most” is not enough in the case of a game like Diplomacy, and often fails to lead to the optimal strategy (in particular, in case of bots playing against humans). Typical primary emotions include fear, sadness, anger and happiness [10]. There is quite a number of models allowing introducing emotions into a system like ours, some more complex than others. When it comes to computers, HUMAINE Emotion Annotation and Representation Language EARL is worth mentioning [11]. However, it distinguishes a total of 48 emotions, out of which some are of no interest to us. Moreover attempting at using it would move the project to a completely new complexity level, and change the focus too far into the human emotions. The level 5 dip language provides us with the following emotions: *Very Happy*, *Happy*, *Sad* and *Angry*, which is partially related to the theoretical models (e.g. the James model, the Weiner and Graham, and the main categories of the Parrot model [10]). However, fear is not included in our set—basically because it is a weakness when felt by our bot, and a fact impossible to verify when felt by other players (see, also [5]).

From basic emotions we create two aggregated values, *Like*, and *Emotional Trust*. The first one represents the general outcome and how we feel about the opponent, while the latter represents the comfort in/confidence towards someone.

Emotion facts are stored as triples *Power—Feeling—Reason*, where *Power* represents who “felt,” *Feeling* can be: *Very Happy*, *Happy*, *Sad* and *Angry*, while *Reason* is a list of *Expressions* that caused given emotion to arise—basically it is the left hand side of rules that are responsible for creating this fact and perhaps increasing its probability. For example “Germany is happy with Italy offering peace” will be represented as GERMANY—HAPPY—PROPOSE(ITA, GER, COMMIT(ITA, GER, PCE(ITA,GER))). Basic emotions used in the dip framework are extended by two new numerical variables—*Liking* and *Emotional Trust*—that represent how “in general” we feel about our opponent.

3) *Personality*: Players differ—some cheat and lie, some don’t. Some forgive and forget, while others hold grudges till the end of the game (and sometimes, beyond). They can like some actions more than others. This all calls for creating a structure that will represent such characteristics of a bot, its “personality.” In our approach we express personalities in a *.pers* file that is loaded into a belief set. Note that they impact almost every process in the bot—from creating conclusions, relationships, to decision making. Personality contains also additional information—how emotionally we react to some actions, whether some we despise more than others. Therefore, we can model how *Like* and *Emotional Trust* are decreased or increased, based on events in the game. Here, we present sample characteristics:

- Boolean *canLie*.
- List  $\langle \text{int} \rangle$  *initialLike*—initial value of like for opponents.
- List  $\langle \text{int} \rangle$  *likeWeight*—here, we can define how much

weight is put to different types of actions or emotions; this is useful for expressing a person that “only looks at good sides,” that gains “liking” fast when made *Happy*, but loses it slowly when made *Sad* or *Angry*, etc. Same applies to changes of opinion about a player that attacks him or supports him.

- int *likeLimit*—value of liking below which we stop negotiation (“I hate you, I’m not talking to you”).

All entries are in a form *Class—Name—Values*. One line is one such triple, with elements separated with the — character. Thanks to this we can easily use Java Reflection to parse the file and load all the necessary data. Here is a fragment of the .pers file:

- Boolean—canLie—0
- Vector <Integer>—initialLike—50 50 50 50 50 50
- int—memlevels—5
- Vector <Integer>—memvals—10 9 7 6 4

4) *Reasoner*: Our bot is capable of creating conclusions from facts thanks to a rule based reasoner. Based on the rules defined in an external .rf file, our agents can create new facts. Rules are in a form  $\{Expression+\} \rightarrow \{Expression+\}|p$ , where the Expression is a helper class containing either an order, a message, a predicate, or a function. *p* stands for initial probability that the drawn conclusion is correct. Some rules might have 100% probability. In other cases, we treat them as some initial data that can undergo many changes during computations. Here we can see the predicates connected to the level-0 language, and where do they come from:

- ATK(unit owner, region owner)—attack—when a destination of MTO(unit, region) is occupied by opponent’s unit, or it is a supply center controlled by an opponent.
- DEF(power init, power aim)—aim power defended from init power (symmetric to ATK).
- PTAH(unit owner, power,x)—in position to attack/help in x rounds—currently distance x is set to not bigger than 2, can be generated by MTO, HLD, RTO and BLD orders. This value was found empirically as being not too far for conclusion to be insignificant, and not too close to trigger the reasoning too late.
- HLP(unit owner, help receiver)—if an opponent issued a support order towards rival’s unit, we can say it was *helping* it. Generated by SUP(unit, HLD(unit)) and SUP(unit,MTO(unit,region)).
- DSTR(power,power)—a power *destroyed* opponent’s unit—generated directly by forced DSB(unit), or indirectly by REM(unit) by taking away resources.
- STR(power,power)—one power is *stronger* than the other—comes from calculating resources and units.
- OWN(power, region)—power owns given region—created from the current board state.
- NONE(power, power)—nothing occurred between two powers.

For level 1 following predicates were introduced:

- WAR(Power, Power)—given powers are at war—can be generated for several reasons, e.g. attack, helping the

attacker, being allied with the attacker.

- ALY(Power, Power+)—given powers are allied/at peace—can be caused by received messages, observed help or affiliation.
- LIKE(Power, Power)—needs to be used with the EVAL function, stands for *Like*.
- ETRUST(Power, Power)—needs to be used with the EVAL function, stands for *Emotional Trust*.
- TRUST(Power, Power)—needs to be used with the EVAL function, stands for *Trust*.
- REP(Power)—needs to be used with the EVAL function, stands for *Reputation*.
- EREP(Power)—needs to be used with the EVAL function, stands for *Emotional Reputation*.
- HAPPY(Power, Power, expression), VHAPPY(Power, Power, expression), ANGRY(Power, Power, expression), SAD(Power, Power, expression)—used to express how given power felt about opponents action/sent message/etc.

We also have the five functions used for the *Time line* related analysis:

- FUT(expression)—reasoning about future.
- PAST(expression)—reasoning about past.
- INC(expression)—increase value by a modifier defined in the .pers file.
- DEC(expression)—decrease value by a modifier defined in the .pers file.
- EVAL(expression)—used for emotion evaluation, it fetches the numerical value of given emotion.

As stated above, beliefs contain a variable *Time line* and can be grouped by it. To switch between such groups we use the FUT and PAST functions. We use such to express opinions like “Opponents units are close to me, he might attack me in the next round”. A simple example of when can that be useful—“German units are getting close to the Italian units, and hence Germany might attack Italy”, represented as  $PTAH(GER, ITA, 2) \rightarrow FUT(ATK(GER, ITA))|30$  (value 30% is just a sample probability).

The *INC* and *DEC* functions are used to deal with numerical variables such as *Trust*, *Reputation*, their emotional equivalents, and *Liking*. They provide a signal that a value should be decreased or increased, e.g. “Germany attacked Italy hence Italy likes Germany less” expressed as:  $ATK(GER, ITA) \rightarrow DEC(LIKE(ITA, GER))|100$ . The quantity by which we should increase or decrease the value is defined in the *Personality* of the bot and can be further affected by the current situation; e.g. if we are close to losing, we might care less or care more about specific situations surrounding us.

The general structure of the *Reasoner* is depicted in Figure 7. As it can be easily seen, the .rf file is first analyzed by a *Rule Reader*, based on a tree parser (thanks to it we can handle any level of nesting). The *Rules* are then passed to the *Rule Manager* that handles firing, matching and substitution. Whenever the belief base is extended, we fire the rules, generate all possible facts and add them to the belief base.

**Algorithm IV.1:** REASONER(*newbelief*)

```

FireRules(newbelief);
PerformMatching();
List < Expression > out = Substitution();
if out! = null
then GenerateNewBeliefs(out);

```

Fig. 7. Structure of the Reasoner.

Rule matching is performed as follows: we read all variables in the rule premises, and all variables in the beliefs sequence, and we check them for application. If we managed to assign them in a one-to-one manner, such that the order is preserved, it means that we can draw conclusions. Variables in the created expressions, such as Power  $x$ , are substituted by real values (e.g. France, Russia), and finished elements can be added into the belief base. Thanks to the Java Reflection, we can easily extend predicate, function, etc., without any further need of modifying the mechanism. For an example illustrating how the reasoner works, please refer to [5].

Here is an extract from the .rf file:

- MTOOrder(Power  $x$ , Region unit, Region dest) + OWN(Power  $y$ , Region dest) = ATK(Power  $x$ , Power  $y$ ) — 100
- ATK(Power  $x$ , Power  $y$ ) = WAR(Power  $x$ , Power  $y$ ) — 100
- DSTR(Power  $x$ , Power  $y$ ) = WAR(Power  $x$ , Power  $y$ ) — 100
- WAR(Power  $x$ , Power  $z$ ) + ALY(Power  $y$ , Power  $x$ ) = WAR(Power  $y$ , Power  $x$ ) — 60

Please note that here “+” and “=” do not stand for arithmetic operations. They are simply used to express “and” and implication. It is just a personal choice of symbols.

5) *Making a decision:* The *Ambassador* and the *Arbitrator* choose which recommendations to follow and which should be (at least temporarily) abandoned, in the following way:

- Compute all consequences—this basically means that using the reasoner we estimate the effects of our decisions, e.g. taking into account consequences for the owned land, feelings of our opponents, possible opponent actions, etc.
- Compute likelihood of given outcome—is evaluated on the basis of probabilities of specific beliefs that form it.
- Filtering—we remove decisions that would lead us to something we cannot, or do not want to, do, e.g. betrayal (breaking a promise, or attacking an ally without prior warning)
- Evaluate the benefits of decisions—whether we get some land, if yes, how good it is; do we make enemies, if yes, how bad it is; do we do something we like and approve, if yes, how much; and so on. Very important is the fact whether our decision gets us any closer to our goals.

The outcome of the algorithm is an assignment of a numerical value to a decision. This allows us to rank possibilities and as a

result the highest ranked one is the winner. However, we send to the *Arbitrator* three best results, just in the case it rejects a choice we established to be the best (from our perspective).

The decision cycle of the *Arbitrator* is very similar when it comes to computing consequences and judging them. However, its main aim is to agree to the most beneficial combination of *Ambassadors*’ proposals. In this way, we can say that while *Ambassadors* think locally, the *Arbitrator* thinks globally.

6) *Embedding it all into BDI:* All knowledge our bot has—Personality, power assignments, facts etc., is held in the belief base. It consists of four main belief sets, all represented as JADEX Tuples:

- Assignment Belief Set—stored as pairs *Agent—Power*
- Setup Belief Set—holds personality elements and technical information for the bot
- Emotion Fact Belief Set—holds triples *Time line—Emotional Fact—Memory time*. *Time line* is the time given belief was generated, *Memory time* is the number of rounds it should be stored. The *Emotional fact* is an emotional predicate (possibly combined with functional) such as DEC(LIKE(GER, ITA)). It is accompanied by the reason for generation.
- Rational Fact Belief Set—as above, for rational facts.
- Relationship Belief Set—stores powers and their relationships with each other: *Trust, Reputation, Emotional Reputation, Like*. Note that it was separated from other belief sets, in order to reduce the complexity of computation—we are going to use them often, and it is handy to save them separately, rather than having to repeatedly search through the belief set.

In our bot, we make an extensive use of triggers and goals. The first category is the events caused by an incoming message—they are used to launch the FIPA Arbitrator/ Ambassador message plans (JADEX plans that are meant to react to messages). Next, content is read and an appropriate event dispatched. It fires the internal message plan that performs the desired action—what interests us most at this point, is the INFO received from the *MotherMessenger*, and what is happening in the *Arbitrator* during the decision agreement. Other messages mainly operate on the belief base—add, remove, fetch, etc. Manipulating beliefs can fire the reasoner plan (JADEX plan that launches the reasoner and adds created beliefs to the belief base) that is meant to draw all possible conclusions from the provided data. Message sending is fired with a send event, dispatched from any plan. The schema of message exchange between the *MotherMessenger* and the *Arbitrator* is represented in Figure 8.

After receiving a negotiation message passed by the *Mother* and the *Mother Messenger*, we can finally start thinking and planning. As mentioned before, we evaluate the consequences. In the meantime it might happen that one of the JADEX agents will demand additional information or some confirmation from another agent. We “pause” the fire plans, mentioned above, and wait for an answer (for the time defined in the *Setup Belief Set*). After deciding what to do, we hold 3 best possible choices and call the *Arbitrator* for an approval, as

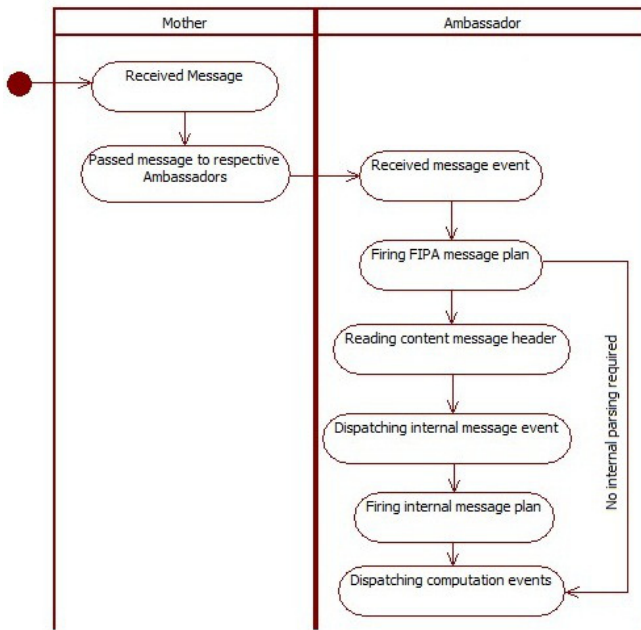


Fig. 8. Receiving messages diagram.

depicted in Figure 9. The received answer is then passed to the *MotherMessenger* agent.

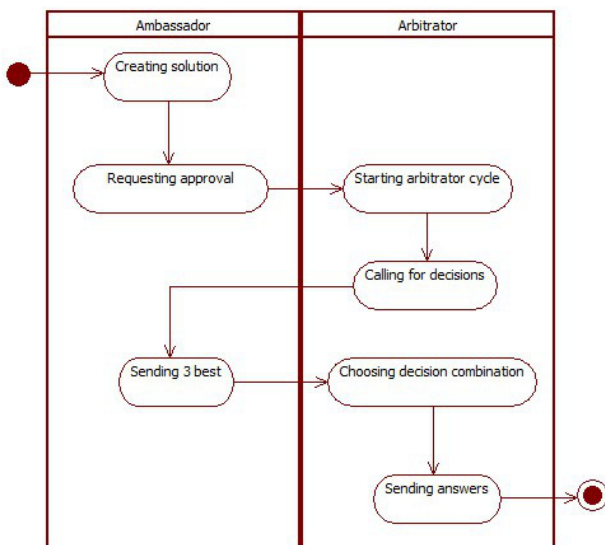


Fig. 9. Arbitrator cycle.

However, with the current plans and events, we have a problem—the first act of negotiation from the *Ambassadors* side can start only as an effect of a received message. It means that we need a plan that is “always running in the background,” and performing part of the computations mentioned above to check whether we need some interaction with opponents. If

such a need is established, we create an appropriate goal and fire the grading plan that will choose what should be done. This is the default behavior. Of course, it can happen that the previous step will precisely know what is required and pass such data to the *Ambassador*. For technical issues, such as how precisely the triggers work, see [9].

#### V. CONCLUDING REMARKS AND PRELIMINARY TESTS

In this paper we have described the background of the Diplomacy game and the architecture of the bot that is to play it using Negotiations. After correcting some minor implementation errors, the bot will be tested against humans, and the IIIA—CSIC bot. One of the important directions of experimental testing of the proposed design will be use of different personality setups. Most interesting should be the tests with different emotional – rational ratios, e.g. 0:100, 50:50, 100:0. Finally, we expect to use the experimental data to tune up the bot’s heuristic and in this way complete the work with dip level-1.

Initial tests with the 50:50 ratio included 10 games with 3 human players (3, 3 and 4 games), in proportion 1 SillyNegoBot, 6 DumbBots and 1 human. Such choice was made to analyze the behavior of the SillyNegoBot. We used DumbBots to have clear view at the SillyNegoBot—human negotiations. Out of the conducted games, 3 ended prematurely due to technical errors, out of the remaining 7, our bot won 4. Results gave us more information on what needs to be corrected both from the implementation and the behavior side. However, the overall conclusion is that the current version of the SillyNegoBot is too trustful and too nice towards other players.

#### ACKNOWLEDGMENT

We would like to thank A. Fabregues and C. Sierra for support with the framework and the bot, and the COST Action IC0801 for funding the STSM visit that made creation of the architecture design possible.

#### REFERENCES

- [1] Wikipedia, the free encyclopedia. [http://en.wikipedia.org/wiki/Diplomacy\\_\(game\)](http://en.wikipedia.org/wiki/Diplomacy_(game)).
- [2] Diplomacy Archive <http://www.diplom.org/~diparch/god.htm>
- [3] Calhamer, Allan B. The Rules of Diplomacy 4th Edition. s.l. : Avalon Hill Game Co., 2000.
- [4] Diplomacy AI Centre. <http://www.daide.org.uk/>.
- [5] Sylwia Polberg, SillyNegoBot Architecture, to appear in: M. Essaïdi, M. Ganzha, M. Paprzycki (eds.), Software Agents, Agent Systems and Applications, IOS Press, 2011
- [6] Sylwia Polberg, Shupantha Kazi Imam, Developing Bots for the Diplomacy Game, submitted for publication.
- [7] Fabregues, Angela and Sierra, Carles. Dipgame. [www.dipgame.org](http://www.dipgame.org).
- [8] Fabregues, Angela and Sierra, Carles. Testbed for Multiagent Systems. <http://www.iiia.csic.es/files/pdfs/DiplomacyTestBed.pdf>.
- [9] Braubach, Lars, et al. JADEx - BDI Agent System. <http://jadex-agents.informatik.uni-hamburg.de/xwiki/bin/view/About/Overview>.
- [10] Emotions [http://changingminds.org/explanations/emotions/primary\\_secondary.htm](http://changingminds.org/explanations/emotions/primary_secondary.htm) <http://changingminds.org/explanations/emotions/basic/%20emotions.htm>
- [11] EARL <http://emotion-research.net/projects/humaine/earl>



# Computing Equilibria for Constraint-based Negotiation Games with Interdependent Issues

Mihnea Scafes  
University of Craiova  
Software Engineering Department  
Bvd.Decebal 107, Craiova, 200440, Romania  
scafes\_mihnea@software.ucv.ro

Costin Bădică  
University of Craiova  
Software Engineering Department  
Bvd.Decebal 107, Craiova, 200440, Romania  
badica\_costin@software.ucv.ro

**Abstract**—Negotiation with interdependent issues and nonlinear, non-monotonic utility functions is difficult because it is hard to efficiently explore the contract space. This paper presents a new result in automated negotiations with interdependent issues, complete information and time constraints. We consider that agents express their preferences using constraints defined as one interval per issue and that we represent their constraint sets as intersection graphs. We model negotiations as a bargaining game and we show that the equilibrium solution is one of the maximal cliques of the constraint graph. Consequently, we find that the problem of computing the equilibrium solution has polynomial-time complexity when the number of issues is fixed.

## I. INTRODUCTION

**M**OST of the research in negotiation has considered that the negotiation issues are independent, i.e. the value of one issue does not depend on the values of the other issues. In this case the utility functions are typically linear and monotonic. As the agents often have to make concessions to the other agents, while also maximizing their utility, they must determine contracts that, depending on a given situation, will either increase or decrease the utility values. If the utility functions are monotonic, it is computationally easy to decide for either increasing or decreasing the value of an issue in a potential contract in such a way that the value of the utility function will be increased or decreased. But, if there are multiple possible trade-offs between issues, computing the agreement is computationally more expensive.

Real-world applications may require negotiation with interdependent issues which are typically aggregated in a more complex way than simple linear utility functions. Examples can be given in areas as meeting scheduling [1], [2], cooperative design [3], and energy markets [4]. In these cases utility functions are nonlinear and non-monotonic, so searching the contract space for contracts that score above a given utility value requires an efficient exploration method because the number of contracts is exponential.

The interdependencies between issues can be represented in several ways: utility graphs [5], [6], interval constraints [7]–[9] and influence matrices [10]. Most of the approaches for finding the optimal outcome of the negotiation implement simulated annealing algorithms [6]–[10]. This approach has the disadvantage that agents may accept contracts of a lower utility value, rather than accepting only contracts of a higher

utility value as with the hill-climbing method. Thus, an agent employing a simulated annealing strategy will get a lower outcome from a negotiation with an agent employing a hill-climbing strategy [10]. The simulated annealing approach was implemented using a mediator for computing contracts and most of the time it determines near-optimal solutions because computing an optimal contract is time expensive.

In our work we focus on bargaining with complete information, under time constraints, about interdependent issues. When modelling such negotiations, one of the most important things is to study the equilibrium solution. A general framework for finding the equilibrium solution for this class of problems has already been proposed in the literature [11]. The application of this framework produced a series of interesting results, under certain conditions regarding the negotiation configuration [11]–[13].

We build our work on the same framework for finding the equilibrium solution of our negotiation problem. By using a preference model suited for interdependent issues [8] and modelling negotiation as a bargaining game under time constraints, we define and solve a problem that leads us to the equilibrium solution. But probably the most important aspect is that we prove that this problem can be solved in polynomial time when the number of negotiation issues is fixed.

According to our literature review, there is no other work that gives the same important results in the context of bargaining. However, the preference model that we use here has been already considered in previous works [8], although in different negotiation settings. Here we are studying negotiation as bargaining game with complete information, differently from the mediated, auction-based mechanisms used in other works [8]. Moreover, our results can be successfully applied to other negotiation settings.

The paper is structured as follows. We start in Section II with an analysis of the preference model with interdependent issues. We briefly describe in Section III the negotiation protocol and the structuring of the offers. In Section IV we model negotiation as a bargaining game of alternating offers and we prove that the equilibrium solution can be computed in polynomial time. Section V provides an overview of related research on negotiation with interdependent issues. In Section VI we draw conclusions and we point to future work.

## II. PREFERENCES AND UTILITY

There is a set of  $m$  negotiation issues  $X = \{x_1, x_2, \dots, x_m\}$ . Issues can be assigned values only from their domain. There are  $m$  domains,  $D_1, D_2, \dots, D_m$ . A *contract* is an  $m$ -dimensional variable defined over  $D_1 \times D_2 \times \dots \times D_m$ , i.e. a combination of issue values. The set of possible contracts is therefore  $D_1 \times D_2 \times \dots \times D_m$ .

Agents have preferences over the issues in the form of constraints.

*Definition 1:* Let  $X = \{x_1, x_2, \dots, x_m\}$  be the set of issues under negotiation and  $D_1, D_2, \dots, D_m$  their domains. A constraint  $C$  is a boolean function defined over the set of possible contracts, i.e.  $C : D_1 \times D_2 \times \dots \times D_m \rightarrow \{0, 1\}$ . We say that an arbitrary constraint  $C$  is satisfied by a contract  $x$  when  $C(x) = 1$ .

A constraint restricts the domains of the issues to smaller sets of values and introduces a set of contracts  $Dom(C) = \{x \in D_1 \times \dots \times D_m \mid C(x) = 1\}$ . We refer to  $Dom(C)$ , i.e. the set of contracts  $x$  for which  $C(x) = 1$ , as the *domain of constraint*  $C$ .

But as agents usually have complex preferences that are modeled by combining several constraints, we need to know if these constraints are compatible.

*Definition 2:* A constraint set  $C$  is *consistent* if  $\exists x \in D_1 \times D_2 \times \dots \times D_m$  such that  $C(x) = 1, \forall C \in C$ .

A consistent constraint set defines a domain equal to the intersection of the individual domains defined by each constraint of the set. For example, if we have a constraint set  $C = \{C_1, C_2, C_3\}$ , then the *domain of constraint set*  $C$  is  $Dom(C) = Dom(C_1) \cap Dom(C_2) \cap Dom(C_3)$ . An inconsistent constraint set defines an empty domain that contains no contracts.

We can associate an intersection graph to the constraint set of an agent. We associate a vertex to each constraint. Two vertices are connected by an edge if the domains of their associated constraints intersect. If  $C$  is the set of constraints then the associated constraint graph is denoted by  $G_C$ .

In what follows we will refer to this intersection graph as the constraint graph. If we assume that each agent is expressing his preferences using a constraint set (which might be consistent or not as a whole), he will have such a constraint graph.

Please note that a constraint graph, say  $G = (V, E)$ , is composed by intersecting  $m$  graphs  $G_i = (V, E_i)$ ,  $1 \leq i \leq m$ , one for each issue, which share the same vertex set, but not the edges. A vertex in graph  $G_i$  corresponding to constraint  $C$  is assigned the domain of issue  $i$  which is part of constraint  $C$ . The constraint graph  $G$  has the same vertex set as  $G_i$ ,  $1 \leq i \leq m$ . Two vertices in graph  $G$  are connected by an edge if the two vertices are connected in all graphs  $G_i$ , i.e.  $(x, y) \in E$  if and only if  $(x, y) \in E_i, \forall x, y \in V, \forall 1 \leq i \leq m$ . We refer to graphs  $G_i$  as *issue graphs*.

The following proposition describes how a consistent constraint set looks like in the constraint graph.

*Proposition 1:* Any consistent constraint subset defines a clique in the constraint graph.

*Proof:* Let  $\{C_1, C_2, \dots, C_k\}$  be a consistent constraint subset. It follows that  $Dom(C_1) \cap Dom(C_2) \cap \dots \cap Dom(C_k)$  is non empty. Then for each  $i \neq j$ ,  $Dom(C_i) \cap Dom(C_j) \neq \emptyset$ , so the constraint graph associated to the constraint set  $\{C_1, C_2, \dots, C_k\}$  is a clique. Therefore, a consistent constraint subset defines a subgraph of the constraint graph in which all the vertices are connected with each other, i.e. a clique. ■

The reverse is not always true, i.e. not every clique corresponds to a consistent constraint set. Note that as a consistent constraint set defines a non-empty domain, we can say that a clique in a constraint graph also defines a non-empty domain.

Agents negotiate about a set of issues, seeking to increase their outcomes. The following definition introduces the utility function that agents use to evaluate contracts. We assume that the agents use the preference model described earlier in this section. This model of utility functions is adopted from [8].

*Definition 3:* Let  $C_A$  be the constraint set of agent  $A$  (the set of all constraints used by agent  $A$  to express his preferences). Each constraint  $C \in C_A$  has an associated weight (a strictly positive real number)  $\omega_C > 0$ , which the agent uses to build a preference relation over constraints. These weights are normalized, i.e.  $\sum_{C \in C_A} \omega_C = 1$ . The function  $u_A : D_1 \times D_2 \times \dots \times D_m \rightarrow \mathbb{R}$ ,  $u_A(x) = \sum_{C \in C_A} \omega_C \times C(x)$  is the utility function of agent  $A$ .

The utility function corresponding to outcome  $x$  is the weighted sum of all constraint evaluations in  $x$ . Throughout the paper we assume that agents try to maximize their utility functions [14].

Note that the utility function for independent preferences is usually modeled as a weighted sum of issue values. However, even if it has a similar form (i.e. a weighted sum), the utility function from Definition 3 defines a complex aggregation of constraints and usually has a non-monotonic shape [8].

An agent must determine consistent combinations of constraints in order to find contracts that score above a particular utility threshold. Please note that not every value in  $[0, 1]$  can be scored by the utility function, as the set of all possible values scored by the utility function is finite. Compared to the linear programming problems studied earlier for independent utilities [11], [13], exploration of the contract space becomes a combinatorial optimization problem.

For the rest of the paper we make a very important assumption, that drives the result of our work. We assume that a constraint defines a single interval (open or closed) on the real line for each negotiation issue. Example 1 illustrates a constraint set with such constraints. A constraint is therefore defined as a conjunction of interval memberships on the real line, with one interval per issue. Please note that in this case the issue graphs are interval graphs [15] and a constraint graph is composed by intersecting  $m$  interval graphs. Therefore, under this assumption, the reverse of Proposition 1 is true, i.e. each clique is a consistent constraint subset<sup>1</sup>.

<sup>1</sup>This statement results from the fact that real intervals have the Helly property: if a set of real intervals is such that each two of them intersect then all of them have a non-empty intersection.



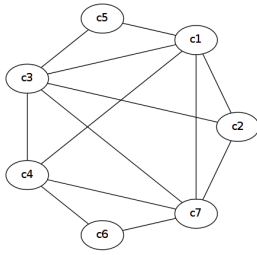


Fig. 1. Example of constraints. Weights are not illustrated, for simplicity. Edges connect vertices whose domains intersect

*Example 1:* Let's suppose that an agent has the following simple constraints over 2 issues denoted  $x, y$ , defined over  $[0, 100]$ :  $c1$  ( $x \in [37, 84]$ ),  $c2$  ( $x \in [79, 96]$ ),  $c3$  ( $x \in [41, 86] \wedge y \in [5, 78]$ ),  $c4$  ( $x \in [5, 51] \wedge y \in [33, 98]$ ),  $c5$  ( $x \in [61, 69] \wedge y \in [36, 40]$ ),  $c6$  ( $x \in [14, 25]$ ),  $c7$  ( $y \in [62, 69]$ ). When the interval for an issue is not specified, it is assumed that the domain of the issue is not restricted. The constraint set is depicted in Figure 1.

### III. PROTOCOL

We model negotiation as a bargaining game of alternating offers [16]. In this paper we study only bilateral negotiations, i.e. negotiations with 2 players denoted by  $A$  and  $B$ .

In the general case, an *offer* consists of a set of values for each issue. The set assigned to an issue must be included in the domain of the issue. An agreement is represented by the last offer that has been accepted, so it also consists of a set of values for each issue. We consider that agents are indifferent about the issue values chosen from these sets and an exact agreement (i.e. that consists of exactly one value from each of these sets for each issue) can be chosen randomly by one of the players.

Following our preference model described earlier, an offer contains a list of *offered constraints* with one interval per issue<sup>2</sup>. The offered constraints can be constraints from the constraint set of the agent or constraints formed from constraints in the constraint set by shrinking their intervals. Shrinking of an interval happens when an interval is intersected with another interval, as a result of combining constraints into consistent constraint sets. In other words, the constraints in the offers are composed of one interval per issue and these intervals represent domains of consistent constraint subsets (or cliques in the constraint graph) of the agents.

The agent receiving an offer can accept the offered constraints or he can accept only parts of offered constraints obtained by shrinking their domains (i.e. by shrinking one or several issue intervals). This operation makes sense as agents try to satisfy as many of their constraints as possible,

<sup>2</sup>We are consistent with the assumption made earlier. In the most general case an offer can contain a more complex specification of a set of values per issue, for example a union of intervals, rather than a single interval. However, we do not deal with this case in this paper. Our results are based on the assumption that there is only a single interval assigned to one issue in a constraint, as in [8].

and sometimes shrinking domains results in contracts that satisfy more constraints. In this case, shrinking happens when the agent makes combinations of constraints to determine consistent constraint sets and intersects their domains with the domains in the opponent's offer. In this way he can accept only a part of the offer that maximizes his utility.

Agents seek to maximize their private utilities, but it is required that the opponent's utility is not lost, i.e. Pareto agreements are preferred. An agreement is Pareto if there is no other agreement that is at least as good for both agents and strictly better for at least one of them.

### IV. NEGOTIATION WITH COMPLETE INFORMATION

Let  $A$  and  $B$  be two agents that negotiate in order to reach an agreement over their sets of preferences. They would like to allocate values to multiple issues, but the issues are constrained to subsets of their domains. The issues are interdependent, i.e. the subset of values that an issue is allowed to take depends on the subsets of values that the other issues can take. These interdependencies are modeled using multi-issue constraints, as described in Section II. Each agent has a constraint set containing possibly many such constraints. Agents  $A$  and  $B$  have constraint sets  $C_A$  and  $C_B$  respectively.

Agents have time constraints given as deadlines and discount factors similarly to [11]–[13]. We represent deadlines as time steps in negotiation. For simplification (but without losing the generality of the algorithm), we make the assumption that both agents have the same deadline, denoted with  $n$ . We assume that agents have no gain if the agreement is reached after the deadline and so agents wish to reach agreements before their deadlines. Moreover, we assume that the utility of each agent decreases as negotiation advances in time. We model this with the help of a discounted utility function that depends on two variables: the contract and the negotiation step. By taking into account deadlines and discount factors, the agents  $i \in \{A, B\}$  have the discounted utility function:

$$U_i(x, t) = \begin{cases} u_i(x) \times \delta_i^{t-1} & \text{if } t \leq n \\ 0 & \text{if } t > n \end{cases}$$

with  $\delta_A \in [0, 1)$ ,  $\delta_B \in [0, 1)$  and  $t \geq 1$ . Again, to reduce the complexity of the representation (but not the generality), we assume that both agents have the same discount factor  $\delta$ . Thus, the utility an agent gets from a contract in the first negotiation round is greater than the utility it gets from the same contract in subsequent negotiation rounds.

Please note that the negotiation game we study is not the *split-the-pie* game [17] previously studied for independent issues. There is no object that the participants want to split and that shrinks over time. Participants want to reach an agreement over a set of values (that are constant during the negotiation), not over shares of an object.

We consider that agents have complete knowledge about the deadline and the discount factor and they know each other's preferences. Our negotiation model builds on existing negotiation models with complete information [11]. We adapt existing equilibrium strategies for negotiation with complete

information, under time constraints, with multiple issues and monotonic utility functions [11] to negotiation with complete information, under time constraints, with multiple interdependent issues and non-linear, non-monotonic utility functions. We show that under specific circumstances (the preferences are expressed using constraints that restrict issue values to intervals and the utility functions are weighted sums of constraint evaluations) the equilibrium solution is not as hard to compute as it was recently considered. Although the same preference model under the same circumstances had been studied before, only approximate solutions were proposed, using heuristic methods [8], [10]. No efficient algorithm that finds the exact solution was proven to exist.

Finding equilibrium strategies for negotiation with multiple divisible issues under time constraints has been proven to be equivalent to solving a particular maximization problem, the *fractional knapsack problem* [11]. The solution can be computed in linear time, in the first negotiation round and the strategy set is a *Nash equilibrium*. We adapt these results to the preference model discussed in this paper. We use a notation similar to [11]. Let  $S_i(t)$  be the equilibrium strategy of agent  $i \in \{A, B\}$  for  $1 \leq t \leq n$ .

The following proposition defines *Nash equilibrium strategy* for our bargaining game with deadlines and discounted utility functions.

*Proposition 2:* For  $t = n$ , the equilibrium strategy is:

$$S_A(n) = \begin{cases} \text{OFFER } O_A(n) = \{x | x = \arg \max_y U_A(y, n)\} & \text{if } A\text{'s turn} \\ \text{ACCEPT } ACC_A(n) & \text{if } B\text{'s turn} \end{cases}$$

$$S_B(n) = \begin{cases} \text{OFFER } O_B(n) = \{x | x = \arg \max_y U_B(y, n)\} & \text{if } B\text{'s turn} \\ \text{ACCEPT } ACC_B(n) & \text{if } A\text{'s turn} \end{cases}$$

For  $t < n$ , the equilibrium strategy is:

$$S_A(t) = \begin{cases} \text{OFFER } O_A(t) & \text{if } A\text{'s turn} \\ \text{IF } \exists x \in O_B(t) \text{ s.t. } U_A(x, t) \geq U_A(t+1) \\ \text{THEN ACCEPT } ACC_A(t) & \text{if } B\text{'s turn} \\ \text{ELSE REJECT} \end{cases}$$

$$S_B(t) = \begin{cases} \text{OFFER } O_B(t) & \text{if } B\text{'s turn} \\ \text{IF } \exists x \in O_A(t) \text{ s.t. } U_B(x, t) \geq U_B(t+1) \\ \text{THEN ACCEPT } ACC_B(t) & \text{if } A\text{'s turn} \\ \text{ELSE REJECT} \end{cases}$$

where  $O_i(t)$  with  $i \in \{A, B\}$  are the equilibrium offers of the agents at time  $t$ .

$O_A(t)$  is the set of offered constraints satisfied by all solutions  $x$  of the following maximization problem:

$$\begin{aligned} & \text{maximize } U_A(x, t) \\ & \text{such that } U_B(x, t) \geq U_B(t+1) \end{aligned} \quad (1)$$

$O_B(t)$  is the set of offered constraints satisfied by all solutions  $x$  of the following maximization problem:

$$\begin{aligned} & \text{maximize } U_B(x, t) \\ & \text{such that } U_A(x, t) \geq U_A(t+1) \end{aligned} \quad (2)$$

$U_i(t)$  with  $i \in \{A, B\}$  is the maximum utility that agent  $i$  can get from his equilibrium offer at time  $t$ .

$$ACC_A(t) = \{x \in O_B(t), U_A(x, t) \geq U_A(t+1) | x = \arg \max_y U_A(y, t)\}$$

is the part of the offer of agent  $B$  accepted by agent  $A$  at time  $t$ , if he decides to accept.

$$ACC_B(t) = \{x \in O_A(t), U_B(x, t) \geq U_B(t+1) | x = \arg \max_y U_B(y, t)\}$$

is the part of the offer of agent  $A$  accepted by agent  $B$  at time  $t$ , if he decides to accept.

*Proof:* The equilibrium strategy is similar to [11], but slightly adapted for the preference model with constraints. At the last step ( $n$ ), the agent making the offer (say  $A$ ) is in a strong position and offers a consistent constraint set that gives him the highest utility (diminished with time), i.e.  $\max(U_A) = \max(u_A) \times \delta^{n-1}$ . All the values in the domain of the offered consistent constraint set score the same utility. The other agent's ( $B$ ) best response is to accept, but as specified in Section III, he can accept only a part of the proposal to obtain as much utility as possible. He will intersect the domain of the offered consistent set with one of the consistent constraint subsets of his own constraint set and he will select the intersection that gives him maximum utility,  $ACCB(n)$ . This intersection of the domains might shrink the domain of the offered constraint set. As agents have different preferences, it might be the case that  $B$  gets a high utility, even maximum, making this a key difference from the *split-the-pie* games. At step  $t = n - 1$ , the agent that must make a proposal ( $B$ ) reasons backwards to  $t = n$  and thinks that  $A$  will be able to get  $\max(u_A) \times \delta^{n-1}$ . Therefore, at step  $t = n - 1$ ,  $B$ 's best action is to make an offer that will give to  $A$  an utility at least as high as  $\max(u_A) \times \delta^{n-1}$ , such that  $A$  will immediately accept. From all the possible offers that satisfy this condition,  $B$  chooses those that maximize his utility by solving maximization problem (2). The agent that makes the first proposal has complete information about the negotiation setting and therefore at step  $t = 1$  he can compute the best offer such that the other agent will accept (the whole offer or only a part of it) immediately by reasoning backwards to  $t = n$  and solving maximization problems (1) and (2) for each time step. Both agents play their best response strategies and the strategy set is a Nash equilibrium. ■

The equilibrium solution depends on the agent that makes the first move, so it is neither symmetric, nor unique. As there are two players,  $A$  and  $B$ , there are two equilibrium solutions. Moreover, the equilibrium solution is a set of multiple possible contracts and the final agreement is established randomly by one of the agents, so there may be many possible agreement contracts. However, as an agent equally prefers any contract from the set of the equilibrium solution, this step is trivial and not of critical importance.

Note that our equilibrium strategy is Pareto optimal, i.e. there is no loss of utility points. At each step, an agent offers to the other agent as much as needed such that he can accept his offer. At the same time he tries to maximize his own utility. The agent that receives an offer is able to accept a part of the offer that gives him the highest utility.

Solving maximization problems (1) and (2) for each time step is a difficult process. Agents must perform suitable combinations of constraints to aggregate a certain utility value. So far in the literature the solutions were to sample the contract space by using heuristic techniques like simulated-annealing [8] thus obtaining near-optimal solutions. In this paper we describe a method to find an exact solution to this problem. The next theorem helps us to reduce the number of combinations in order to find the equilibrium solution.

*Theorem 1:* If  $G_{C_i}$  (corresponding to constraint set  $C_i$ ) are constraint graphs of agents  $i \in \{A, B\}$  then the equilibrium solution is one of the maximal cliques of the constraint graph corresponding to the union of the constraint sets of agents  $A$  and  $B$ , i.e.  $G_{C_A \cup C_B}$ .

*Proof:* The agent that makes the first offer, say  $A$ , tries to satisfy as many of his constraints as possible in order to maximize his utility, i.e. a maximal set of constraints. If the offer is constructed according to the method described in Section III, i.e. using the constraints themselves or subintervals of the issue intervals of the constraints, then no additional constraints from his set of constraints,  $C_A$ , can be satisfied by the values included in the offer. In conclusion, no other nodes from  $G_{C_A}$  are taken into consideration and the equilibrium solution is maximal with respect to the nodes of  $G_{C_A \cup C_B}$  that are part only of  $C_A$ . The agent receiving an offer,  $B$ , can accept a part of the offer that maximizes his own utility, i.e. the one that satisfies as many constraints as possible from  $C_B$ . The part of the offer is computed by intersecting the issue intervals in the offer with the issue intervals from his own constraint set. Because no additional constraints can be satisfied by the part of the offer he accepts, it follows that the solution is maximal with respect to  $C_B$  (and with respect to the nodes in  $G_{C_B}$ ). Because no other nodes can be taken into consideration either from  $C_A$  or from  $C_B$ , it results that the equilibrium solution contains the domain of a maximal consistent set of constraints from  $C_A$  and  $C_B$  – a maximal clique in  $G_{C_A \cup C_B}$ . ■

Even if this theorem shows that the search space can be drastically reduced, finding all the maximal cliques of the constraint graph is still a difficult problem. Note however that according to our assumption, constraints are single intervals and the constraint graph is a special type of graph formed by intersecting the issue graphs. Consequently, the following proposition shows that the number of maximal cliques of the constraint graph can be further reduced.

*Proposition 3:* A clique is maximal in the constraint graph iff its vertex set is the intersection of the vertex sets of maximal cliques in each of the issue graphs.

*Proof:* Let  $G_C = (V, E)$  be the constraint graph and let  $G_i = (V, E_i)$ ,  $1 \leq i \leq m$  be the issue graphs. From the definition  $E = \bigcap_{i=1}^m E_i$ .

⇐ Let  $S_i$  be maximal cliques in  $G_i$  and let  $V_i$  be their vertex sets,  $1 \leq i \leq m$ . We define  $V_S = \bigcap_{i=1}^m V_i$ . Let  $x \neq y$ ,  $x, y \in V_S$ . It follows that for all  $1 \leq i \leq m$  we have  $x, y \in V_i$ . From the fact that  $S_i$  is a clique of  $G_i$  it follows that  $(x, y) \in E_i$ , so  $(x, y) \in E$  that clearly shows that  $V_S$  induces a clique  $S$  of  $G_C$ . Let us now prove that clique  $S$  is maximal. Assuming by refutation that clique  $S$  is not a maximal clique of  $G_C$  we would find a vertex  $x \in V \setminus V_S$  s.t.  $S \cup \{x\}$  is also a clique of  $G_C$ . It follows that  $(y, x) \in E$  for all  $y \in V_S$ , i.e.  $(y, x) \in E_i$  for all  $1 \leq i \leq m$ . Therefore the set of vertices  $V_i \cup \{x\}$  would define a clique of  $G_i$ , that contradicts the hypothesis that  $S_i$  is a maximal clique of  $G_i$ .

⇒ Let  $S$  be a maximal clique of  $G_C$  and let  $V_S$  be its vertex set. From the definition of  $E$  as  $\bigcap_{i=1}^m E_i$  it follows that  $S$  is also a clique of each  $G_i$ . We expand  $S$  to a maximal clique  $S_i$  of  $G_i$  and let  $V_i$  be its vertex set for all  $1 \leq i \leq m$ . Clearly  $V_S \subseteq \bigcap_{i=1}^m V_i$ . If we assume by refutation that the inclusion is strict, we would be able to find another clique  $S'$  of  $G_C$  with vertex set  $V_{S'} = \bigcap_{i=1}^m V_i$  s.t.  $V_S \subset V_{S'}$ . But this contradicts that  $S$  is a maximal clique of  $G_C$ . ■

In other words, the maximal cliques of graphs  $G_{C_A}$  ( $G_{C_B}$ ) are obtained by intersecting the vertex sets of the maximal cliques of issue graphs that compose  $G_{C_A}$  ( $G_{C_B}$ ). Note that issue graphs are interval graphs, and interval graphs are also *chordal graphs*<sup>3</sup>. The number of maximal cliques of a chordal graph is at most equal to the number of its vertices [15] ( $|C_A|$  for agent  $A$  and  $|C_B|$  for agent  $B$ ). It follows that  $G_{C_A}$  has at most  $|C_A|^m$  (i.e. the total number of intersections) maximal cliques, while  $G_{C_B}$  has at most  $|C_B|^m$  maximal cliques. This follows from the fact that there are  $m$  issue graphs that compose each of the constraint graphs  $G_{C_A}$  and  $G_{C_B}$ .

Chordal graphs can be recognized in linear time using procedures *Lexicographic Breadth First Search (LexBFS)* [19] and *Maximum Cardinality Search (MCS)* [20]. Both procedures, when applied to chordal graphs, generate a particular ordering of vertices called *Perfect Elimination Ordering (PEO)* (which every chordal graph has [15]). This ordering has the property that any vertex  $x$  together with its neighbors that are placed to its right in the ordering ( $RN(x)$ ) form a clique. That is,  $x \cup RN(x)$  is a clique. With the help of a PEO it is possible to collect the maximal cliques in linear time [21].

Algorithm 1 computes the equilibrium solution for the case when  $A$  is the first mover and runs in polynomial time. The next theorem gives the complexity.

*Theorem 2:* If each constraint defines a single interval on the real line per issue, the time complexity of finding the equilibrium solution in the first round is  $O((n+1) \cdot (|C_A| + |C_B|)^m + m \cdot (|C_A| + |C_B|)^2 + 2 \cdot m \cdot (|C_A| + |C_B| + E))$ , where  $m$  is the number of issues,  $E$  is the maximum number of edges in the issue graphs that compose  $G_{C_A \cup C_B}$  and  $n$  is the negotiation deadline (the number of steps).

*Proof:* The result follows by summing up the complexities of individual sections of Algorithm 1.

<sup>3</sup>A graph is chordal if any of its cycles with more than 3 vertices has a chord, i.e. an edge connecting two vertices non-adjacent in the cycle [18].

**Algorithm 1** COMPUTE SOLUTION – A MOVES FIRST

---

```

COMPUTE_A_FIRST( $U_A, C_A, U_B, C_B, n$ )
1:  $G_{C_A \cup C_B} \leftarrow$  transformation of  $C_A$  and  $C_B$  into constraint graph
2: for  $i = 1$  to  $m$ 
3:    $peo[i] \leftarrow$  ComputePEO( $i, G_{C_A \cup C_B}$ ) for the issue graph  $i$ 
4:    $mc[i] \leftarrow$  ComputeMaximalCliques( $peo[i], G_{C_A \cup C_B}$ )
5:    $solution \leftarrow$  empty clique
6:    $MAXCG \leftarrow$  compute maximal cliques of  $G_{C_A \cup C_B}$ 
7:    $UAMAX \leftarrow 0, UBMAX \leftarrow 0$ 
8:    $LASTUA \leftarrow 0, LASTUB \leftarrow 0$ 
9:   for  $t = n$  to 1
10:    if ( $t$  is odd) then //A's turn
11:      for each clique in  $MAXCG$ 
12:        if ( $U_A(clique, t) \geq UAMAX$  and
13:            $U_B(clique, t) \geq LASTUB$ ) then
14:            $UAMAX \leftarrow U_A(clique, t)$ 
15:            $LASTUB \leftarrow U_B(clique, t)$ 
16:            $solution \leftarrow clique$ 
17:        else // B's turn
18:          for each clique in  $MAXCG$ 
19:            if ( $U_B(clique, t) \geq UBMAX$  and
20:                $U_A(clique, t) \geq LASTUA$ ) then
21:                $UBMAX \leftarrow U_B(clique, t)$ 
22:                $LASTUA \leftarrow U_A(clique, t)$ 
23:                $solution \leftarrow clique$ 
24: return  $solution$ 

```

---

Transformation of constraint sets into a single constraint graph takes  $O(m \cdot (|C_A| + |C_B|)^2)$  – line 1. First we create a larger constraint set from the constraint sets of the two agents. The larger constraint set will have size  $|C_A| + |C_B|$ . Then we create the graph  $G_{C_A \cup C_B}$  from this constraint set. The existence of an edge will be tested for every pair of constraints in the set and this operation costs  $O(m \cdot (|C_A| + |C_B|)^2)$ . The multiplication with  $m$  in the equation means that the domains of all issues will be tested for intersection, for each pair of constraints.

LexBFS or MCS algorithms (line 3) take  $O(|C_A| + |C_B| + E)$  to produce a PEO that is stored into  $peo[i]$  [19], [20]. Maximal cliques are saved into  $mc[i]$  in time  $O(|C_A| + |C_B| + E)$  [21]. This happens  $m$  times, once for each issue graph, so lines 2 to 5 take  $O(2 \cdot m \cdot (|C_A| + |C_B| + E))$ . Storing maximal cliques of  $G_{C_A \cup C_B}$  into  $MAXCG$  costs  $O((|C_A| + |C_B|)^m)$  by taking the intersections among all possible combinations of maximal cliques of issue graphs. The search for equilibrium solution for each step is carried out in time  $O(n \cdot (|C_A| + |C_B|)^m)$ . For each step ( $n$  in total) we must check all the maximal cliques (at most  $(|C_A| + |C_B|)^m$ ). Summing up, we get  $O((n+1) \cdot (|C_A| + |C_B|)^m + m \cdot (|C_A| + |C_B|)^2 + 2 \cdot m \cdot (|C_A| + |C_B| + E))$ . ■

We can observe that if the number  $m$  of issues is fixed then the complexity is polynomial with respect to the number  $n$  of constraints.

## V. RELATED WORK

According to our literature review, there is little work on negotiation about interdependent issues. Probably the first result to appear in the literature is the negotiation model of Klein [10]. After the authors observed that hill climbers perform better when paired with annealers, they proposed a model with an annealer as a mediator and a voting mechanism for agents. The model performs quite well, but the solutions

are not exact and a mediator that performs the annealing is required. Differently from our work, there is no game-theoretic analysis of the model and there is no theoretical study of the preference model.

There are works that build on [10], such as [7]–[9]. All of them use simulated annealing, again without any game-theoretic analysis of the model. Moreover, their model has some disadvantages: the number of bids per agent is restricted because of performance limitations; the achieved optimality decreases with the number of issues. An important thing that worths mentioning is that, while these works used the same preference model as we did (in fact we have borrowed their preference model), they did not derive the same conclusions.

Another work that employs simulated annealing is [6]. Simulated annealing is used by agents to accept contracts and the accepted contracts are mutated in the next step. The contract is thus improved until the deadline is reached. In our work we use the hill climbing strategy for accepting contracts and we compute the exact solution of the negotiation problem.

There are results in the literature that approach the problem differently. By using utility graphs, authors of [5] achieve an exponentially decrease of the complexity of the problem, though the problem remains complex. In [12], the authors consider approximations of generic separable nonlinear utility functions and show that the equilibrium can be computed in linear time. A comparison of the outcomes for different negotiation procedures is also provided.

## VI. CONCLUSIONS AND FUTURE WORK

We have shown that, under certain assumptions, the equilibrium solution of negotiations with nonlinear utility functions can be computed in polynomial time.

We are not aware of a similar work that derives the same powerful conclusions, although a similar preference model (constraints with intervals assigned to issues) has been studied before [7]–[9]. Compared to previous related works, we provide a game-theoretic analysis of the negotiation model using a method inspired from [11] and we show that the equilibrium solution can be computed in polynomial time. The agents are using the hill-climbing approach of accepting contracts. Therefore, we consider that our approach for negotiation with interdependent issues is novel.

As future work, we plan to experimentally evaluate the algorithm in order to see how it performs against real-world complex negotiation scenarios. In particular, we are interested in the average computation time of the algorithm when applied to various realistic scenarios.

## REFERENCES

- [1] E. Shakshuki, H.-H. Koo, and D. Benoit, "Negotiation strategies for agent-based meeting scheduling system," in *Proceedings of the International Conference on Information Technology*, ser. ITNG '07, Washington, DC, USA: IEEE Computer Society, 2007, pp. 637–642.
- [2] J. Wainer, P. R. Ferreira Jr., and E. R. Constantino, "Scheduling meetings through multi-agent negotiations," *Decis. Support Syst.*, vol. 44, no. 1, pp. 285–297, November 2007.
- [3] Y. Jin and M. Geslin, "Roles of negotiation protocol and strategy in collaborative design," in *Design Computing and Cognition '08*, J. S. Gero and A. K. Goel, Eds. Springer Netherlands, 2008, pp. 491–510.

- [4] K. P. Clark, S. van Splunter, M. Warnier, and F. M. T. Brazier, *Expressing intervals in automated service negotiation*, ser. Grids and Service-Oriented Architectures for SLAs. Springer-Verlag, 2010, ch. 7, pp. 67–75.
- [5] V. Robu, D. J. A. Somefun, and J. A. La Poutré, “Modeling complex multi-issue negotiations using utility graphs,” in *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2005, pp. 280–287.
- [6] A. Kardan and H. Janzadeh, “A multi-issue negotiation mechanism with interdependent negotiation issues,” in *ICDS '08: Proceedings of the Second International Conference on Digital Society*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 55–59.
- [7] H. Hattori, M. Klein, and T. Ito, “A multi-phase protocol for negotiation with interdependent issues,” in *IAT '07: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 153–159.
- [8] T. Ito, H. Hattori, and M. Klein, “Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces,” in *IJCAI, M. M. Veloso, Ed., 2007*, pp. 1347–1352.
- [9] K. Fujita and T. Ito, “A preliminary analysis of computational complexity of the threshold adjusting mechanism in multi-issue negotiations,” in *WI-IATW '07: Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 383–386.
- [10] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam, “Negotiating complex contracts,” *Group Decision and Negotiation*, vol. 12, pp. 111–125, March 2003. [Online]. Available: <http://jmvidal.cse.sc.edu/library/klein03a.pdf>
- [11] S. Fatima, M. Wooldridge, and N. R. Jennings, “Multi-issue negotiation with deadlines,” *Journal of Artificial Intelligence Research*, vol. 27, pp. 381–417, 2006. [Online]. Available: <http://eprints.ecs.soton.ac.uk/13079/>
- [12] S. S. Fatima, M. Wooldridge, and N. Jennings, “An analysis of feasible solutions for multi-issue negotiation involving non-linear utility functions,” in *Proc. 8th Int. Conf on Autonomous Agents and Multi-Agent Systems*, 2009, pp. 1041–1048. [Online]. Available: <http://eprints.ecs.soton.ac.uk/17065/>
- [13] S. Fatima, M. Wooldridge, and N. R. Jennings, “Approximate and online multi-issue negotiation,” in *6th International Joint Conference on Autonomous Agents and Multi-agent Systems.*, 2007, pp. 947–954. [Online]. Available: <http://eprints.ecs.soton.ac.uk/14219/>
- [14] J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton, NJ: Princeton University Press, 1947.
- [15] D. Fulkerson and O. Gross, “Incident matrices and interval graphs,” *Pacific J. Math*, pp. 835–855, 1965.
- [16] M. Osborne and A. Rubinstein, *A Course in Game Theory*. Cambridge, MA: MIT Press, 1994.
- [17] A. Rubinstein, “Perfect equilibrium in a bargaining model,” *Econometrica*, vol. 50, no. 1, pp. 97–109, January 1982. [Online]. Available: <http://ideas.repec.org/a/ectm/emetrp/v50y1982i1p97-109.html>
- [18] M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs (Annals of Discrete Mathematics, Vol 57)*. Amsterdam, The Netherlands, The Netherlands: North-Holland Publishing Co., 2004.
- [19] D. J. Rose, R. E. Tarjan, and G. S. Lueker, “Algorithmic aspects of vertex elimination on graphs,” *SIAM J. Comput.*, pp. 266–283, 1976.
- [20] R. Tarjan and M. Yannakakis, “Simple linear-time algorithm to test chordality of graphs, test acyclicity of hypergraphs, and selectivity reduce acyclic hypergraphs,” *SIAM J. Comput.* 13, p. 566579, 1984.
- [21] J. R. Blair and B. Peyton, “An introduction to chordal graphs and clique trees,” *IMA*, vol. 56, pp. 1–29, 1993.



# Agent-Oriented Knowledge Elicitation for Modeling the Winning of “Hearts and Minds”

Inna Shvartsman, Kuldar Taveter

Tallinn University of Technology, Department of Informatics, Raja 15, 12618 Tallinn, Estonia, Emails:  
innashvartsman@hot.ee, kuldar.taveter@ttu.ee

**Abstract**—Agent-oriented modeling is a top-down approach for modeling and simulating the behaviors of complex systems. This research addresses the application of agent-oriented modeling to eliciting and representing knowledge for social simulations. We provide an overview of agent-oriented modeling and describe a case study on conflict resolution that has an impact on winning “hearts and minds” of the occupied territory’s population. After that we propose a method for eliciting and representing knowledge for social simulations by means of agent-oriented modeling. The models created by means of agent-oriented modeling can be implemented on several simulation platforms, such as NetLogo, Jason, and JADE.

## I. INTRODUCTION

THIS article is concerned with composing practical computer-based simulation scenarios for local conflict resolution in the context of counter-insurgency. In military terms, the most important and valuable strategy is the winning-over of the occupied territory’s population. This kind of military strategy is known as winning the “hearts and minds” of the population. The support and trust of local population are necessary for continuous efforts towards the consolidation of peace, social justice, economic stability, and human rights, so that the country’s reconstruction process can be stronger and more prosperous [1], [2].

Compared to conventional wars between nation-states, counter-insurgency is asymmetric in several aspects. Its operational environment is irregular, characterized by high rate and rapid changes, but also considerable constraints. Dimensions, such as material (disparity of arms between the opposing sides), legal (disparate status of the parties of the conflict), and moral (sides are not morally equal), distinguish asymmetric conflicts from traditional warfare [6]. This multi-dimensionality makes the modeling and simulation of asymmetric warfare complicated but for training purposes highly relevant task.

The first step to be taken to achieve adequate agent-based simulations for complex problem domains is to create a balanced set of models that are able to capture the problem domain from different perspectives and at different abstraction layers. In our view, agent-oriented modeling [3] offers such a balanced set of models.

Asymmetric warfare is one of the problem domains where agent-based simulations should be applied because it involves heterogeneous autonomous entities that include humans, physical subsystems, and software components whose behaviors depend on the situation at hand. Because of the complexity of such problem domains, all scenarios to be simulated should be carefully constructed to assure that they are realistic and useful. This is exactly the reason why, as it has been pointed out by [10], “methodological questions are more and more in the focus of research on agent-based simulation”. Different methodological approaches for developing agent-based simulations have been proposed [10], [11], [12], [13]. This article is confined to the first stage of developing agent-based simulations: eliciting and representing knowledge for computer-based simulations. This is a very important stage in developing social simulations because it involves collaboration between social scientists and computer scientists. Our experience in a defense-related project has shown that agent-oriented models considerably facilitate such inter-disciplinary collaboration.

Previously, we have used agent-oriented modeling for developing simulation environments for military operations in urban environment [4]. In the domain addressed by this article, we deal with long-term social processes of winning “hearts and minds” of the population. The rest of this article is structured as follows. Section II provides an overview of agent-oriented modeling. Section III describes the case study of conflict resolution in an Afghan village. Section IV describes how agent-oriented modeling has been applied to the knowledge elicitation and representation for the case study. Finally, Section V draws conclusions.

## II. AGENT-ORIENTED MODELING

*Agent-oriented modeling* [3] is a top-down approach for modeling and simulating the behaviors of complex systems, which include social phenomena. Agent-oriented modeling enables to analyze and model a given problem domain from three balanced and interrelated viewpoint aspects: interaction, information, and behavior. The core of agent-oriented modeling is the viewpoint framework that is represented in Table I, which contains the types of models proposed by agent-oriented modeling. In addition to



representing for each model the vertical viewpoint aspect of interaction, information, or behavior, Table I maps each model to the abstraction layer of analysis, design, or platform-specific design. At the abstraction layer of platform-specific design, agent-oriented models are turned into dynamic models – simulations.

Each cell in Table I represents a specific viewpoint. For example, the viewpoint of interaction design is captured by agent models and interaction models. It is noteworthy that the interaction, information, and behavior viewpoint aspects of agent-oriented modeling straightforwardly correspond to the respective social, information, and individual background factors for agents' behaviors that have been independently coined in [7].

TABLE I. THE MODEL TYPES OF AGENT-ORIENTED MODELING

Abstraction layer	Viewpoint aspect		
	Interaction	Information	Behavior
Analysis	Role models and organization model	Domain model	Goal models and motivational scenarios
Design	Agent models and interaction models	Knowledge models	Behavioral scenarios and behavior models
Platform-specific design	Platform-specific design models		

### III. THE CASE STUDY

The case study is based on the description of conflict resolution presented in [5]. Conflict resolution is not a goal in itself but rather provides a potential entry point for Blue Force (a term used for friendly forces, e.g., International Security Assistance Force [ISAF] in Afghanistan) when, for example, preventing violence or acting upon violence.

According to [5], there is a conflict between Barmack and Ahmed, who are both relatives and dwellers of a village in the Pashtun region of Afghanistan. They are also small landowners. Barmack owns 10 low grade acres of dry land, about 1 km north of the village. On this land graze 30 goats. Ahmed, Barmack's neighbor, also owns 10 low grade acres of dry land contiguous to Barmack's land. Ahmed owns 25 goats. One morning ten of Barmack's goats are missing. Barmack thinks Ahmed has stolen his goats in order to sell them far from the village. Barmack goes to the village and voices his complaint against Ahmed.

Both Barmack and Ahmed do not hesitate to fight to death to save their honor. Each of them starts to contact his close family (brothers, cousins, etc.) for creating a kind of committee – *Lashkar* – of five people to punish the opponent's behavior. The conflict is now potentially a violent conflict.

In parallel, ten elders (called "*Spingari*") are interested in solving the conflict. There are two motivations for the elders' behavior:

- A common social value of the community is to bring peace to the community via the formation of village council known as *Jirga*. This value is also known as

*Jirga value*. Elders who participate in the *Jirga* are followers of this value. By following this value, they are rewarded with additional honor (enhanced reputation).

- There is personal material interest by the elders. Indeed, many elders have relationship with Ahmed or Barmack (some have relationships with both). If a violent fight destroys a property of one party or kills the party, the elders might suffer from the loss. For example, one elder is the uncle of Barmack and the brother-in-law of one of Ahmed's uncles. As the uncle of Barmack, he benefits from petty chores from Barmack's kids [5].

For solving the conflict, the elders go to see Ahmed and Barmack to convince each of them to choose three proxies among the elders who will represent them at the *Jirga*. If Ahmed and Barmack accept the proposal, they know that they must comply with the resolution made by *Jirga* because of the common *Jirga value*.

The decisions by Ahmed and Barmack are now affected by different factors:

- Compliance with the revenge value;
- Compliance with the *Jirga value*;
- Their self-interest.

Blue Force has an interest in the conflict if there is either strong likelihood of violence of the losing tribe or a rumor or evidence that one party has Taliban support. The Blue Force may react to rumor or evidence of Taliban involvement in different ways. For example, the Blue Force may intervene upon a rumor, or it may investigate the rumor or evidence and only then act upon the issue. The Blue Force may also remain neutral in spite of a rumor or evidence. Both remaining neutral and preventing violence has an effect on winning "hearts and minds" that has to be simulated.

When eliciting knowledge for simulations, we have to consider that violent conflict resolution raises the likelihood of violence between conflicting tribes. For example, the case when losing party has Taliban support and Blue Force attempts to prevent violence has different effect on winning "hearts and minds" compared to the situation when conflict is peacefully resolved and Taliban is not involved. In the simulation, the population should be divided into three groups: positive towards Blue Force, positive towards Taliban, and neutral population. Both Blue Force and Taliban can get support from among neutral population.

In case of peaceful conflict resolution, both Ahmed and Barmack decide to comply with the *Jirga*. The *Jirga* meets for five days, three hours a day, in the centre of the village under a tree. Ahmed and Barmack attend the *Jirga*. Each of them exposes his story. The members of *Jirga* ask questions. After the investigation, it appears that the ten goats were stolen by someone else. The *Jirga* decides to let Barmack express his forgiveness to Ahmed for falsely accusing him. Barmack's problem is recognized by the community, and five farmers are told to give one goat each. In such a way, they follow the group solidarity value. Many people come to the elders to congratulate them on their performance on the *Jirga*. Their prestige is enhanced since they solved the problem.

IV. ELICITING AND REPRESENTING KNOWLEDGE

In this section, we outline a knowledge elicitation and representation process that is appropriate for developing social simulation systems. The process consists of a set of questions that facilitate the development of agent-oriented models and simulations. Answering each question produces one or more models of agent-oriented modeling [3]. The questions have been adapted and modified based on [14]. Because of the limited space in this article, we will focus on the models yielded by the questions here, proceeding by viewpoints of agent-oriented modeling.

From the viewpoint of *behavior analysis*, a *goal model* can be considered as a container of three components: goals, quality goals, and roles [3]. A goal is a representation of a functional requirement of a simulation system. A quality goal, as its name implies, is a non-functional or quality requirement of the system. Goals and quality goals can be further decomposed into smaller related sub-goals and sub-quality goals. The hierarchical structure is to show that the subcomponent is an aspect of the top-level component. Goal models also determine roles that are capacities or positions that agents playing the roles need to contribute to achieving the goals.

A starting point for the knowledge elicitation process is the highest-level goal – *purpose* – of the simulation system to be developed. In social simulations, the purpose is typically the process or phenomenon that is being studied. In the case study of conflict resolution, which forms an important part of winning “hearts and minds”, the purpose of the simulation is as simple as “Resolve the conflict”. The highest-level goal model for the simulation system is shown in Fig. 1. In the figure, rectangles stand for functional goals and clouds – for quality goals. Roles are denoted by stick figures. The goal model depicted in Fig. 1 shows that solving the conflict has two aspects: fighting and finding the truth. These aspects obviously exclude each other but to keep the goal model simple, we do not represent this in the model because the problem domain analysis phase of social simulations typically involves intense discussions between non-technical social scientists and computer scientists.

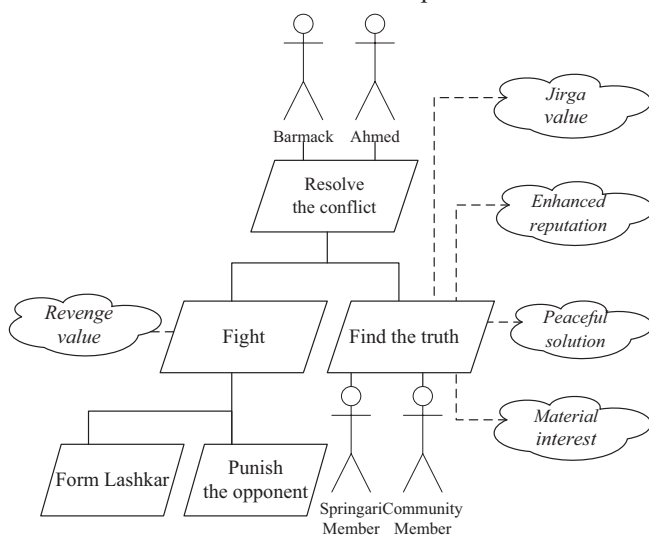


Figure 1. The goal model for solving the conflict

TABLE II. THE ROLE MODEL FOR AHMED

Role name	Ahmed
Description	The role of Ahmed during the conflict
Responsibilities	Form Lashkar; Fight Barmack; Decide compliance to Jirga; Choose 3 representatives for Jirga; Participate in investigation; Prove the innocence
Constraints	Material interest; Revenge value; Jirga value

From the viewpoint of *interaction analysis*, the properties of roles are expressed by *role models* and the relationships between the roles – by *organization model* [3]. An example role model for one of the parties – Ahmed – is represented in Table II. As usual, the role is described in terms of the responsibilities and constraints applying to the agent that will perform the role. Please note that some responsibilities modeled in Table II conflict each other. This is normal because the responsibilities that are eventually fulfilled are determined by the knowledge that agents playing the respective roles hold at the given moment. The knowledge by agents is modeled from the viewpoints of information analysis and information design.

The organization model for conflict resolution is represented in Figure 2. According to the organization model, the Springari Member role relies on the Party role for services and commodities (e.g., foodstuffs). Both the Party and Taliban roles depend on the Community Member role for support; the Jirga Member role controls the Party role because of the social policy of heeding the voice of Jirga, which is modeled as the quality goal “Jirga value” in Fig. 1.

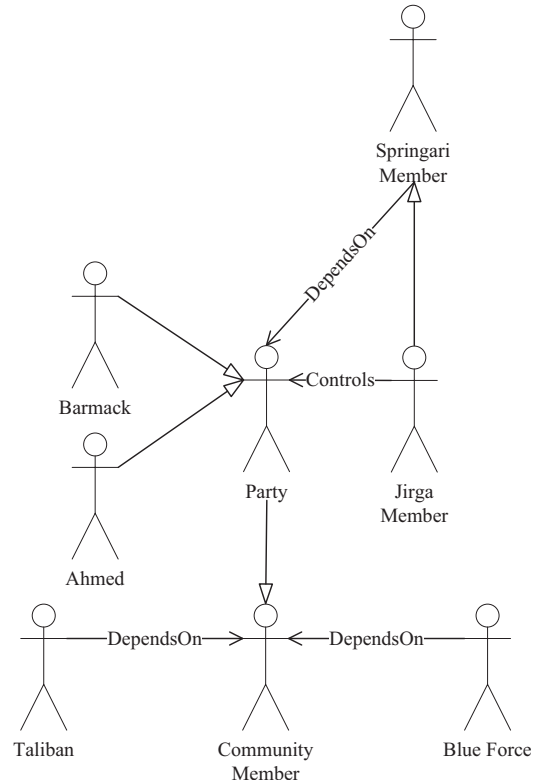


Figure 2. The organization model for conflict resolution

From the viewpoint of *information analysis*, *domain model* represents the knowledge to be handled by the socio-technical system. A domain model consists of domain entities and relationships between them. A domain entity is a modular unit of knowledge handled by a simulation system [3]. For example, to fulfill its responsibilities successfully, an agent playing the role Party in a simulation needs to access the knowledge entities Village and Household, where Village consists of Households.

From the viewpoint of *interaction design*, *agent models* transform the abstract constructs from the analysis stage, roles, to design constructs, agent types, which will be realized in the implementation process. Deciding agent types for simulation systems is simple because usually there is an agent type corresponding to each role. In addition to agent models, *interaction models* represent interaction patterns between agents of the given types. They are based on responsibilities defined for the corresponding roles [3].

From the viewpoint of *information design*, it is essential to represent both private and shared knowledge by agents. An agent's *knowledge model* represents knowledge about the agent itself and about the agents and objects in its environment [3].

Finally, from the viewpoint of *behavior design*, we model how agents make decisions and perform activities. There are two kinds of models under this viewpoint. A *behavioral scenario* describes how agents of the given types contribute to achieving the goals set for the system. *Behavior models* describe the behaviors of individual agents by representing how behaviors depend on the events perceived and knowledge held by agents [3]. Behavior models for the case study embody response functions [8] that determine how agents in simulations make decisions based on knowledge on social values.

At the abstraction layer of *platform-specific design*, agent-oriented models are turned into dynamic models – simulations – that show the effects of the behaviors of individual agents as well as provide information on emergent behavior by the simulation system as a whole. As interactions between the agents involved are highly complex, performing simulations is the only way of predicting their outcome. Appropriate simulations can help to understand the expected behavior of each individual agent and an entire system over time. Therefore, agents can be used for simulating real life situations and exploring the behaviors of humans forming complex simulation systems with “human-in-the-loop” capability.

## V. CONCLUSIONS

We proposed a knowledge elicitation and representation method for developing agent-based simulation systems for social processes. Social processes are studied by a variety of scientific disciplines such as social sciences, psychology, cultural anthropology, etc. The methods used by these disciplines differ from those used by exact sciences in that the underlying mathematics, computational algorithms, and proofs are usually not addressed for social systems. Instead, social processes are described by social relationships, expected outcomes, and theories. For this reason, social

scientists and computer scientists have different theoretical backgrounds and practical experiences. This makes understanding each other and coming to the common vision for models and simulations difficult, especially because social systems are inherently complex and their simulations reflect that complexity [9]. To decrease the complexity, we therefore need to create structured but simple representations of problem domain knowledge. Agent-oriented modeling has proved to be a very suitable approach that facilitates collaboration in this context.

## VI. ACKNOWLEDGEMENT

This research was supported by European Social Fund's Doctoral Studies and Internationalisation Programme DoRa.

## VII. REFERENCES

- [1] Wardak, A. (2003). *Jirga – A Traditional Mechanism of Conflict Resolution in Afghanistan*. The United Nations Online Network in Public Administration and Finance. Retrieved June 24, 2011, from <http://unpan1.un.org/intradoc/groups/public/documents/apcity/unpan017434.pdf>.
- [2] Wardak, A. (2004). The Tribal and Ethnic Composition of Afghan Society. In E. Girardet and J. Walter, *Afghanistan: Crosslines Essential Field Guides to Humanitarian and Conflict Zones*. 2nd Ed. Geneva, Switzerland: Crosslines.
- [3] Sterling, L. & Taveter, K. (2009). *The Art of Agent-Oriented Modeling*. Cambridge, MA, and London, England: MIT Press.
- [4] Shvartsman, I., Taveter, K., Parmak, M., Meriste, M. (2010). Agent-Oriented Modelling for Simulation of Complex Environments. In *International Multiconference on Computer Science and Information Technology - IMCSIT 2010, Wisla, Poland, 18-20 October, Proceedings* (pp. 209-216). Washington, DC: IEEE Computer Society.
- [5] Israel, N., Peugeot, T. (2011). *Social Sciences Modeling in ATHENA*. Working paper.
- [6] Gross, M. (2010). *Moral Dilemmas of Modern War: Torture, Assassination, and Blackmail in an Age of Asymmetric Conflict*. New York, NY: Cambridge University Press.
- [7] Fishbein, M. & Ajzen, I. (2010). *Predicting and Changing Behaviour: The Reasoned Action Approach*. New York, Hove: Psychology Press.
- [8] Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA, and London, England: Harvard University Press.
- [9] United States Marine Corps (2007). *The Agent-Based Simulation Verification, Validation, and Accreditation (VV&A) Framework Study*. Final Report 9 November 2007. Retrieved June 24, 2011, from [https://files.pbworks.com/download/BYAWPV88gF/orsagouge/13331399/VVA%20Phase%20II%20Workshop%203%20Summary%20Report%204-30-08\\_Final.pdf](https://files.pbworks.com/download/BYAWPV88gF/orsagouge/13331399/VVA%20Phase%20II%20Workshop%203%20Summary%20Report%204-30-08_Final.pdf)
- [10] Klügl, F. (2009). Multiagent Simulation Model Design Strategies. In M. Baldon et al. (Eds.), *Proceedings of the Second Multi-Agent Logics, Languages, and Organisations Federated Workshops (MALLOW), Turin, Italy, September 7-10* (CEUR Workshop Proceedings, Vol. 494). CEUR Workshop Proceedings.
- [11] Gilberg, N., Troitzsch, K. G. (2005). *Simulation for the Social Scientist*. 2nd Ed. Open University Press.
- [12] Richiardi, M., Leombruni, R., Saam, N., Sonnessa, M. (2006). A common protocol for agent-based social simulation. *Journal of Artificial Societies and Social Simulation*, 9(1).
- [13] Drogoul, A., Vanbergue, D., Meurisse, T. (2002). Multi-agent based simulation: Where are the agents? In J. S. Sichman, F. Bousquet and P. Davidsson (Eds.), *Multi-Agent-Based Simulation, Third International Workshop, MABS 2002, Bologna, Italy, July 15-16, Revised Papers* (LNCS 2581, 1-15). Berlin, Germany: Springer-Verlag.
- [14] Sterling, L., Miller, T., Taveter, K., Lu, B., and Beydoun, G. (2011). *Requirements engineering using the agent paradigm: a case study of an aircraft turnaround simulator*. Working paper.

# 5<sup>th</sup> International Workshop on Multi-Agent Systems and Simulation

**M**ULTI-AGENT systems (MASs) provide powerful models for representing both real-world systems and applications with an appropriate degree of complexity and dynamics. Several research and industrial experiences have already shown that the use of MASs offers advantages in a wide range of application domains (e.g. financial, economic, social, logistic, chemical, engineering). When MASs represent software applications to be effectively delivered, they need to be validated and evaluated before their deployment and execution, thus methodologies that support validation and evaluation through simulation of the MAS under development are highly required. In other emerging areas (e.g. ACE, ACF), MASs are designed for representing systems at different levels of complexity through the use of autonomous, goal-driven and interacting entities organized into societies which exhibit emergent properties. The agent-based model of a system can then be executed to simulate the behavior of the complete system so that knowledge of the behaviors of the entities (micro-level) produce an understanding of the overall outcome at the system-level (macro-level). In both cases (MASs as software applications and MASs as models for the analysis of complex systems), simulation plays a crucial role that needs to be further investigated.

## TOPICS

MAS&S'11 aims at providing a forum for discussing recent advances in Engineering Complex Systems by exploiting Agent-Based Modeling and Simulation. In particular, the areas of interest are the following (although this list should not be considered as exclusive):

- Agent-based simulation techniques and methodologies
- Discrete-event simulation of Multi-Agent Systems
- Simulation as validation tool for the development process of MAS
- Agent-oriented methodologies incorporating simulation tools
- MAS simulation driven by formal models
- MAS simulation toolkits and frameworks
- Testing vs. simulation of MAS
- Industrial case studies based on MAS and simulation/testing
- Agent-based Modeling and Simulation (ABMS)
- Agent Computational Economics (ACE)
- Agent Computational Finance (ACF)
- Agent-based simulation of networked systems
- Scalability in agent-based simulation
- Agent languages
- Agent learning and planning
- Agent mobility
- Agent modeling, calculi, and logic
- Agent security
- Agents and Service Oriented Computing

- Agents in the Semantic Web
- Applications and Experiences

## STEERING COMMITTEE

**Massimo Cossentino**, ICAR-CNR, Italy  
**Giancarlo Fortino**, Università della Calabria, Italy  
**Marie-Pierre Gleizes**, Université Paul Sabatier, IRIT, France  
**Juan Pavon**, Universidad Complutense de Madrid, Spain  
**Wilma Russo**, Università della Calabria, Italy

## PROGRAM COMMITTEE

**Jean-Paul Arcangeli**, Université Paul Sabatier, France  
**Juan Antonio Botía Blaya**, Universidad de Murcia, Spain  
**Krzysztof Cetnarowicz**, AGH - University of Science and Technology of Krakow, Poland  
**Irenieusz Czarnowski**, Gdynia Maritime University, Poland  
**Paul Davidson**, Blekinge Institute of Technology, Sweden  
**Stefano Galzarano**, Università della Calabria, Italy  
**Alfredo Garro**, Università della Calabria (Italy)  
**Paolo Giorgini**, Università di Trento, Italy  
**Raffaele Gravina**, Università della Calabria (Italy)  
**Samer Hassan**, Universidad Complutense de Madrid, Spain  
**Vincent Hilaire**, Université de Belfort-Montbéliard, France  
**Piotr Jedrzejowicz**, Gdynia Maritime University, Poland  
**Franziska Klügl**, Örebro Universitet, Sweden  
**Adolfo López-Paredes**, University of Valladolid, Spain  
**Ambra Molesini**, Università di Bologna, Italy  
**Ngoc-Thanh Nguyen**, Wroclaw University of Economics, Poland  
**Muaz Niazi**, COMSATS Institute of IT, Islamabad, Pakistan  
**Michael J. North**, Argonne National Laboratory, USA  
**Andrea Omicini**, Università di Bologna, Italy  
**Paolo Petta**, OFAI, Austria  
**Gauthier Picard**, ENSM, Saint-Etienne, France  
**Luca Sabatucci**, ITC-irst, FBK, Italy  
**Valeria Seidita**, Università degli Studi di Palermo, Italy  
**Pietro Terna**, Università di Torino, Italy  
**Erwan Tranvouez**, LSIS, France  
**Giuseppe Vizzari**, Università di Milano Bicocca, Italy

## ORGANIZING COMMITTEE

**Carole Bernon**, Université Paul Sabatier, IRIT (France)  
carole.bernon@irit.fr  
**Giancarlo Fortino**, Università della Calabria (Italy)  
g.fortino@unical.it  
**Jorge J. Gomez-Sanz**, Universidad Complutense de Madrid (Spain) jjgomez@sip.ucm.es





# Multi Agent Simulation for Decision Making in Warehouse Management

Massimo Cossentino, Carmelo Lodato, Salvatore Lopes, Patrizia Ribino  
ICAR Institute  
National Research Council of Italy  
Palermo, Italy  
Email: {cossentino, c.lodato, toty, ribino}@pa.icar.cnr.it

**Abstract**—The paper presents an agent-based simulation as a tool for decision making about automatic warehouses management. The proposed multi-agent system is going to be used in a real environment within a project developed with a company working on logistics. More in details, we have developed a simulation framework in order to study problems, constraints and performance issues of the truck unload operations. We aim to optimize the suitable number of Automated Guided Vehicles (AGVs) used for unloading containers arrived to the warehouse. This is a critical issue since an AGV is a costly resource and an augment in number does not necessarily correspond to an improved unloading speed. The experiment performed with our simulated environment allows us also to evaluate the impact of other elements to the performance.

## I. INTRODUCTION

**M**AKING effective and successful decisions about complex systems is a hard task, especially in business environments. This process usually exceeds human cognitive capabilities because of the huge amount of parameters influencing such systems. The human intuitive judgment and decision making become far from optimal in respect to the growing of the complexity. The quality of decisions is extremely important in many practical situations because a wrong or an ineffective decision could cause a great waste of resources. Overcoming the deficiencies of human judgment is one of the biggest challenges of the scientific community.

Nowadays, simulations are often used in scientific and research contexts in order to evaluate the behavior of several complex systems and especially the behavior of dynamical systems. The simulated system should have the capability of continuously reacting, with a re-organization process, to changes occurring in the environment. Because of their intrinsic nature, agents have been recognized to be a good way for solving complex problems [1][2].

Several studies are being carried out in the field agent-based simulations. Some interesting contributions are given by Franziska Klügl [3][4], Seth Tisue et al. [5], Sean Luke et al. [6] and Nick Collier [7].

In [4] F.Klügl et al. present an integrated framework, named SeSAm (Shell for Simulated Agent Systems), allowing the creation of simulated environments suitable to several kinds of context such as Logistics (coordination, storage layout optimization), Traffic (avoidance of traffic jams, traffic light

control), Passenger Flow (market improvement, evacuation of buildings) etc. . .

Tisue et al. [5] have developed NetLogo, a modeling tool for simulating natural and social phenomena.

MASON [6], proposed by Sean Luke et al., is an extensible, discrete-event multi-agent simulation toolkit in Java. It was designed for a wide range of multi-agent simulation tasks ranging from swarm robotics to social complexity environments.

Finally, RePast [7] is a software framework for agent-based simulation created by Social Science Research Computing at the University of Chicago. It provides an integrated library of classes for creating, running, displaying, and collecting data from an agent-based simulation.

In the field of the logistics, several agent simulations are proposed for different purposes such as modeling and management of supply chains [8][9][10], optimization of production planning [11], traffic [12] etc. . .

The problems addressed in this paper concern the optimization of an automated logistic warehouse. In such kind of warehouse the handling of goods is performed by means of Automated Guided Vehicles (AGVs). Usually these vehicles move along optical guides drawn on the warehouse floor. These optical guides are defined at design time of warehouse and they are used during its entire life cycle imposing constraints about the traffic. A critical issue is the efficient employment of resources in order to avoid overcrowding of the guides.

In addition, another element constraining the performance of a logistic warehouse is the sorter, which task is directing toward a new destination the goods unloaded by AGVs. Generally, a sorter has a given capacity (sorting speed) to be considered in order to balance the elements whose a warehouse is composed of.

In this paper, we propose an agent-based simulation in order to solve a decisional problem about warehouse management generating performance measures. The simulation has been developed using Jason [13], a Java-based interpreter for an extended version of the AgentSpeak [14][15] language based on the BDI (Belief-Desire-Intentions) model [16].

The work presented in this paper was carried out under the IMPULSO <sup>1</sup> (*Integrated Multimodal Platform for Urban and*

<sup>1</sup>Further information available at <http://www.vitrociset.it> - Section Ricerca&Sviluppo

*extra urban Logistic System Optimization*) project funded by the Italian Ministry for Economic Development.

The remainder of the paper is organized as follow. The section II introduces the decision making problem addressed in this paper and defines the simulation objectives. Moreover it provides an overview of the AgentSpeak language and Jason interpreter. In section III, we then proceed to the presentation of the multi agents architecture for the proposed simulation by specifying the features of the agents and the environment in which they will perform their activities. The section IV shows the performance results obtained from the simulation which allow us making considerations. Finally some discussions and conclusions are drawn in sections V and VI respectively.

## II. THEORETICAL BACKGROUND

The decision problem discussed in this paper concerns some aspects of the IMPULSO project. IMPULSO aims to develop new technologies and capabilities in order to improve the management and transport of products, based on cooperation models while ensuring highest levels of security. It offers an integrated system for goods management within the logistic districts, for their storage in special metropolitan distribution centers and finally, for distribution within the cities. One of the issues addressed by the IMPULSO project concerns the evaluation of efficient resources employment to be adopted for the handling of goods inside a logistic district. For clarity, a logistic district is a large area composed of several warehouses where the freight forwarders deliver their container.

In this paper, we see in detail a node of the supply chain. The specific case concerns the management of the automatic container unloading by means of the use of AGVs inside a warehouse of the logistic district.

In the presumed scenario, containers are carried by articulated lorries. Whenever a lorry arrives to the warehouse the container has to be unloaded. The container holds several kinds of goods grouped in boxes called *pallet*. Each pallet must be unloaded from the container and carried to a specific area dedicated to the sorting of goods. In this area each pallet will be opened and its contents (packages of goods) sent to a sorter. The sorter will cluster the packages according to their destination. Finally, smaller vehicles (e.g.: eco-friendly trucks) will take them to their new destination (usually in town).

The transport of pallets toward the sorting area is committed to automatic vehicles with optical guidance (AGVs). This means that each warehouse inside a logistic district must be equipped with appropriate optical signals which defines all permissible paths for an AGV.

Each defined layout of optical paths imposes limits on the use of the resources (AGVs). In accordance with the available paths only some AGVs can work effectively at the same time. Since an AGV is a costly resource, it is crucial to establish how many AGVs can work at the same time without getting in each others way thus delaying the unloading operations.

The choice of the maximum number of AGVs is also constrained by the capacity of the sorter. In other words, a semi-automatic sorter can process a maximum number of

packages per time unit. Thus, within the limits imposed by the available paths, an augment in number of AGVs does not necessarily correspond to an improved unloading speed because of saturation of the sorter. The sorter could actually be a bottleneck of the warehouse and it can cause long waiting queues. A proper allocation of resources, which respects the constraints imposed by the warehouse layout and by the sorting capacity, can significantly reduce management costs.

In addition, it is useful to establish what are the critical paths inside a warehouse, that is those whose unavailability can cause a traffic block. For these reasons, we use a multi-agent simulation as a tool for decision making about warehouse management.

The simulation allows us to explore the variables constraining the problem. In this instance, we want to establish, for a given warehouse configuration, not only what is the maximum number of AGVs usable in order to maintain high performances but also what are the critical elements of the system.

The optimization of the warehouse layout is out of the scope of this paper because it is a task of another component of the IMPULSO project. However, our work provides some useful information for improving the design of automatic logistic warehouses.

The next subsection provides an overview of the tools used for the simulation.

### A. Development environment

We decided to adopt a multi-agent based solution because it adequately fits the real scenario coming from the IMPULSO project. In fact, real AGVs are autonomous robots capable of executing the mission they received by a mission controller. Among the available platforms we decided to use Jason [13] that offers relevant utilities for the implementation of such system.

Jason is a Java-based interpreter for an extended version of AgentSpeak [14][15], a Prolog-like logic programming language. One of the most interesting aspects of AgentSpeak is that it based on a the belief-desire-intention (BDI) model[16].

In the BDI model, agents continually monitor their environments and act to change them, based on the three mental attitudes of belief, desire and intention.

*Beliefs* are information the agent has about the world (i.e. itself, others agents and the environment), which could also be out of date or inaccurate.

*Desires* represent all possible states of affairs that an agent would achieve. A desire is a potential influencer of the agents actions. So it is possible for a rational agent to have desires that are mutually incompatible each other. Desires can be represent possible option for an agent.

*Intentions* are the states of affairs that the agent has decided to work towards. An agent looks at its options and chooses between them. Options selected in this way become intentions.

The behavior of agents in Jason is defined by means of a set of plans created in AgentSpeak.



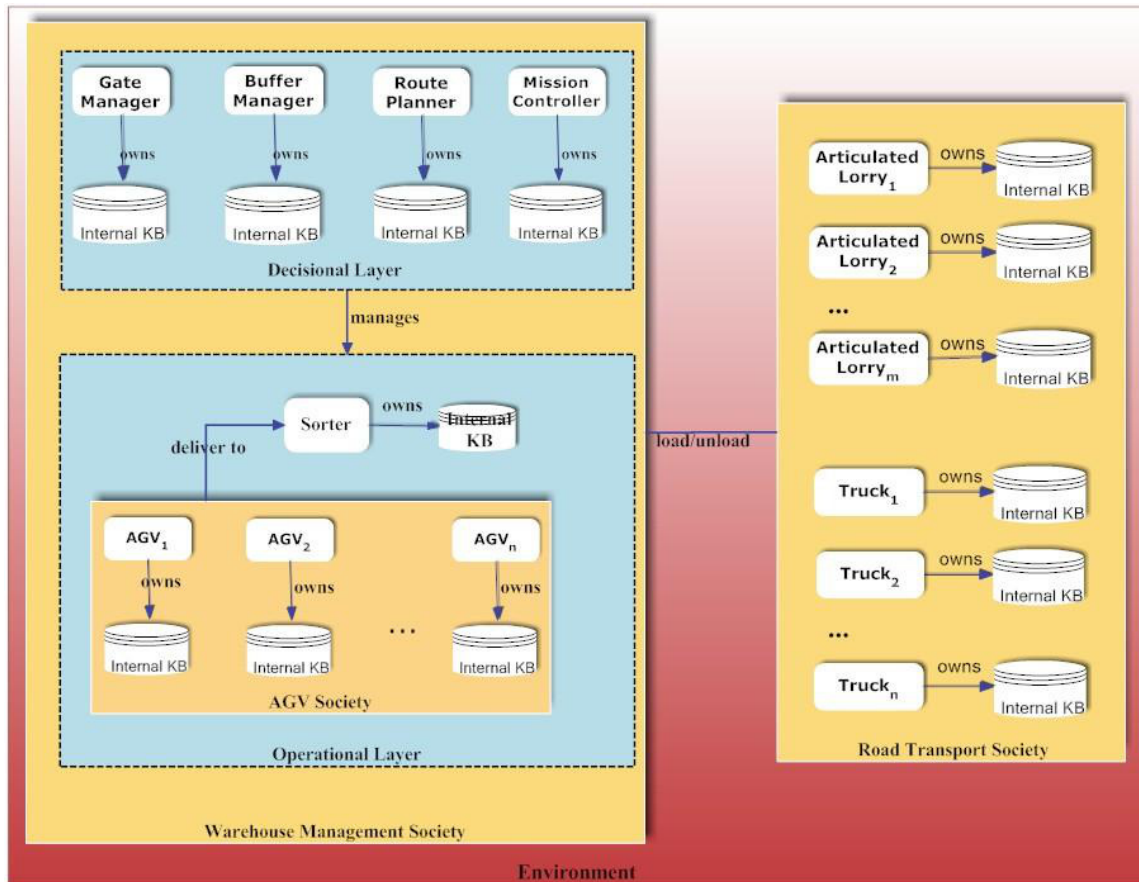


Fig. 1. The Multi-agent system architecture.

Practically, agents respond to the perceptions coming from environment changes. Such perceptions influence beliefs and commitment of agent goals. Agents respond to these changes by selecting plans from the plan repository for each change of beliefs and then by instantiating one of these plans as an intention. These intentions can be composed of actions, goals and plans to be achieved.

A plan in AgentSpeak is composed of three main elements organized in the following form:

$+triggeringEvent : context < -body$

The **triggeringEvent** describes the situations in which a plan may be applicable for the execution. The **context** can be used for specifying the condition to make the plan applicable even if an event has triggered that plan. The **body** can be considered the consequent of the event linked to the context. Within the body commonly are defined the actions that an agent must perform to fulfill its own goals.

In the next section we describe the design and the implementation of a multi-agent organization used for the proposed simulation.

### III. THE PROPOSED SIMULATION FRAMEWORK

The proposed simulation framework is based on an agent organization situated in a specific environment. The multi agent organization is composed of (see fig.1):

- a *Warehouse Management Society* governing the activities inside a warehouse;
- a *Road Transport Society* for goods transportation from/toward a logistic district.

The agents of the Warehouse Management Society belong to different layers in accord with the role played in the society. We distinguish two layers: the *Decisional Layer* and the *Operational Layer*. Agents playing managerial roles belong to the former while the second is defined by the agents which perform operational activities. The decisional layer of the Warehouse Management Society is composed of four agents: the Gate Manager, the Buffer Manager, the Mission Controller and the Route Planner. The operational layer is formed by a Sorter agent and an AGVs Society.

The Road Transport Society consists of different means of transport such as articulated lorries and trucks.

In the subsection III-A we paid attention on constitutive elements of the environment. While the features of each agent will be defined in the subsection III-B.

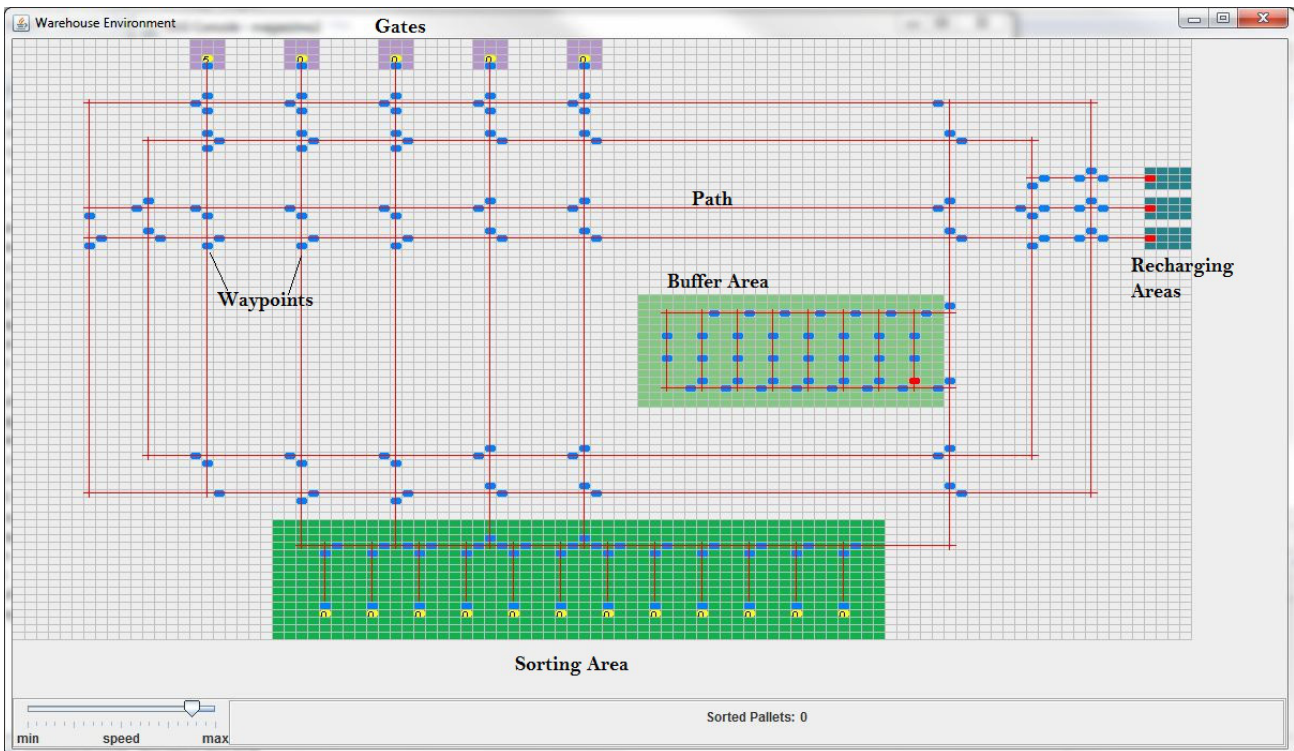


Fig. 2. The simulated environment.

#### A. Environment

As usual, we have defined not only the elements the environment is made of but also how agents can interact with the environment. Specifically we have defined what an agent perceives, when the agent is able to perceive and finally how the actions it performs influence the environment.

The studied environment represents a real warehouse situated inside a logistic district. It is a very dynamic environment because there are several agents performing unsynchronized actions but it is also an open environment due to the exchanges with the outside world.

The elements of such environment are (see fig. 2):

- a set of *Gates* in which articulated lorries can park waiting for unloading;
- a set of *Recharging Areas* where the AGVs can recharge their batteries;
- a *Sorting Area* where the pallets are forwarded toward a new destination through several input points (called *Sorter Places*);
- a *Buffer Area* where it is possible to temporarily store pallets when the sorter is busy. This area is also used for parking AGVs that are waiting for a new mission;
- a set of possible *Paths* representing the optical guidance for AGVs. Each route section (i.e.: path connecting only two waypoints) is usually one-way, but some of them can be two-way (e.g.: the entrance of gates);
- a set of *Waypoints* near to the crossing points of paths.

#### B. Agents

For our purpose we have defined different kinds of agents (see fig.1):

- the *Gate Manager* manages the allocation of gates at the arrival of articulated lorries. It also takes into account the amount of pallets to unload;
- the *Buffer Manager* governs the parking areas and buffering. It can reserve a parking place for the agents that require it;
- the *Route Planner* allocates the paths for AGVs. Each path is computed by means of Dijkstra's shortest path algorithm [17];
- the *Mission Controller* implements the strategy of container unloading. It also assigns to each AGV the mission of carrying pallets from gates to the sorter in accord to a nearest neighbor policy;
- the *Sorter* manages the work inside the sorting area and communicates the free place where it is possible to deliver a pallet for an AGV. Moreover, it interacts with Truck agents for loading the ready boxes for the delivery toward new destinations;
- the *AGV* is the agent that simulates the behavior of real forklift that performs the pallet transport inside a warehouse from arrival gate to sorting area;
- finally the *Articulated Lorry* and the *Truck* are the agents that perform the transport of goods toward and from a logistic district respectively.

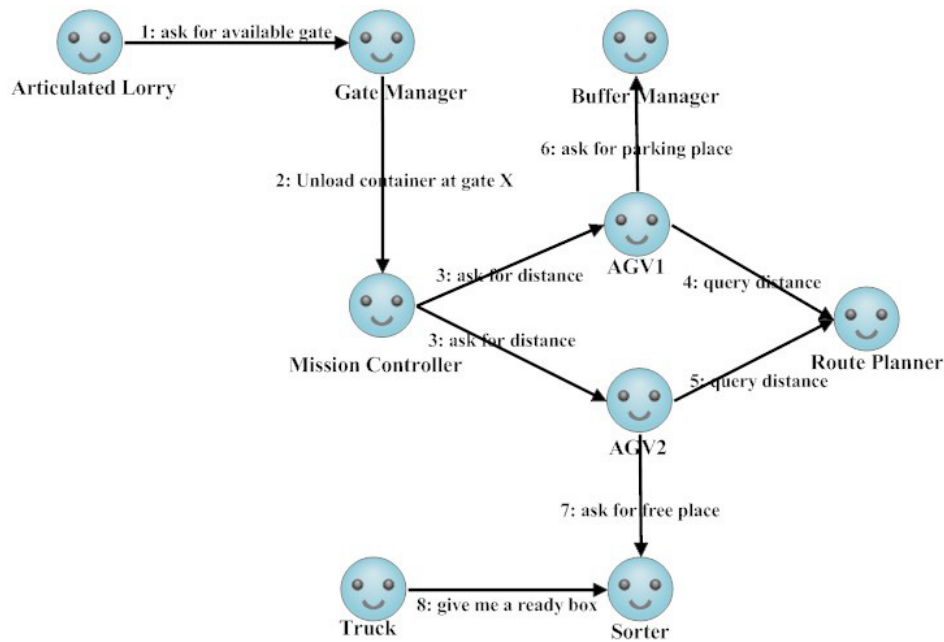


Fig. 3. A typical operative scenario.

Each agent owns an internal knowledge base. The agent knowledge base contains its initial beliefs, the beliefs resulting from perceptions about the environment, the goals and the plans to achieve them.

In the next section we show the experimental results obtained from simulation.

#### IV. EXPERIMENTAL SETUP

The tests have been conducted on a warehouse configuration coming from specifications of the IMPULSO project, but we want to underline that the multi agent model adopted for the simulation is independent of the specific warehouse configuration.

In the proposed instance, the simulated environment is a warehouse consisting of:

- n°5 Gates where the articulated lorries leave their containers waiting to be unloaded;
- n°1 Sorting Area with twelve Sorter Places (where the pallets are left in order to be addressed toward next destination);
- n°3 Recharge Areas where an AGV goes, whenever its battery is low;
- n°1 buffer area with 16 places;
- n°62 crossing with 142 waypoints;
- n°70 oriented route sections.

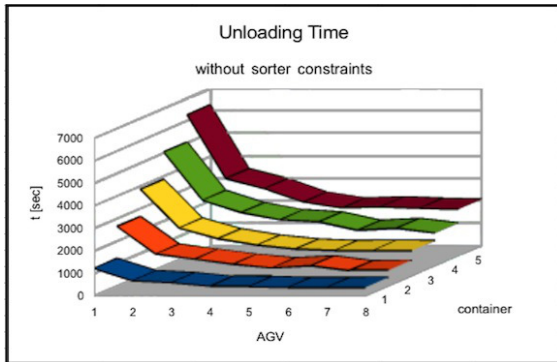
When the simulations start, the warehouse settings are the following:

- the AGVs are located in different places of the warehouse (such as recharge area, buffer area, etc...). We assume they always are on a waypoint;
- all gates are free;

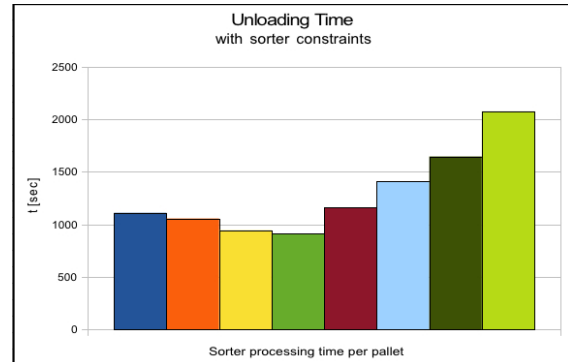
- the sorter is empty;
- all paths are available.

As previously said, we adopted the Dijkstra algorithm for finding the shortest path between gate and sorter and vice versa. The algorithm has been implemented in such a way to provide an alternative path if the shortest one is already busy. In fact, according with the specification of the IMPULSO project, we adopt a very conservative policy in order to avoid collisions between AGVs. This policy consists in reserving the entire path assigned to each single AGV if it is possible. Otherwise we reserve only an alternative intermediate path. We are conscious that the actual reservation strategy may probably cause a waste of time but we want to avoid any chance of a collision between AGVs because too costly and dangerous.

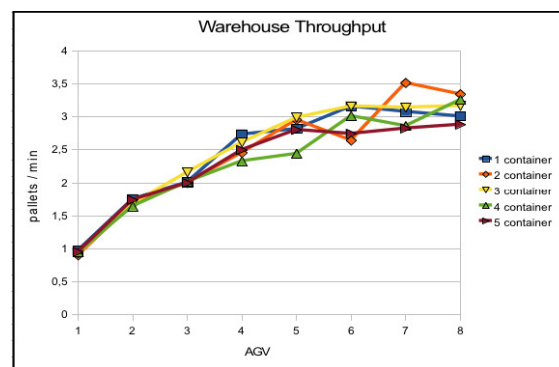
During our simulations, several articulated lorries may arrive. Each of them carries a container with a certain number of pallets. The receiving and unloading of different containers can be performed at any moment. When there is a container ready to be unloaded, some AGVs are assigned to take a pallet from the container and to transport that to the Sorter. The commitment is defined by the Mission Controller agent according to the nearest neighbor policy. The figure 3 shows a simplified diagram of the communications among agents during a typical simulation scenario. We prefer to show only the most meaningful messages exchanged among agents for the sake of clarity. In this scenario only an Articulated Lorry arrives. It asks to the Gate Manager for an available gate where to park. Then, the Gate Manager acquires information about the arrived load, and subsequently, it informs the Mission Controller that there is a container to unload at gate X. The Mission Controller asks to each AGV its distance from the gate



(a) The time spent by AGV (Automated Guided Vehicle) for simultaneously unloading containers using a sorter with infinite capacity.



(b) The time spent by 4 AGVs to unload 100 pallets from 5 containers versus the pallet processing time at the sorter.



(c) The number of pallets unloaded per minute versus the number of AGVs for different numbers of containers.

Fig. 4. Simulation results

where the lorry is parked. The closest AGV is committed to the unload of the first pallet. Then the committed AGV asks to the Sorter for an available place, while other not committed AGVs go to the parking zone. In accordance with our requirements, the Sorter always assigns places starting from the right side of the sorting area (see figure 2). It is worth to note that although in this scenario we suppose to employ only one AGV, usually all available ones (with a proper sequence) are committed to the unloading task.

The Route Planner agent determines, for each AGV, the shortest path among those available, also providing a measure of the distance. Pallets are left in the sorting area for a fixed amount of time simulating the time spent by human operators to transfer goods from the pallet to the boxes. When available, Truck agents release the places occupied by the already filled boxes by loading them.

We conducted several simulations with different parameter values in order to evaluate the behavior of the system. Particularly, we observed the relevance of two variables: the number of AGV agents versus the number of containers that are simultaneously unloaded. The charts shown in figure 4 highlight the results obtained from these simulations. These results are discussed in section V.

## V. DISCUSSION

Diagrams in Figure 4 show the results of several simulations for the warehouse configuration of the case study at issue.

All reported times are scaled according to the real time. We would like to underline that the reported results do not suffer of any random influence.

The diagram shown in the figure 4(a) displays the progress of the unloading time depending on the number of used AGVs and the number of unloaded containers during different simulations (20 pallets for each container). This diagram is based on the assumption that the time spent by the sorter to process one pallet is zero, this corresponds to have an infinite capacity sorter ( $C_{sorter} = N^{\circ}SorterPlace/ProcessingTime$ ).

More in details, in the same figure we can see that the time necessary for unloading one container, decreases with the number of AGVs but the slope of the curve significantly decreases after about 4 or 5 AGVs.

Any decision about the acquisition of the suitable number of AGV should start from the estimation of the average number of containers that are simultaneously unloaded. In the following we will suppose this is 3. In this scenario, our simulations suggest the following decision guidelines: if the preferred criterion is optimizing the cost/benefit ration of the AGV



employment, from figure 4(a) we can see that five AGVs is a reasonable choice. Buying more AGVs does not contribute significantly. For instance, with 5 AGVs we can unload 3 containers in 1205 seconds while with 6 AGVs we need 1139 seconds. The difference (5%) does maybe not justify the increase in cost.

Conversely, if no compromise may be accepted on the unloading time, at all costs, the suggested number of AGVs is 6. Of course we leave this strategical choice to the warehouse manager.

After that we have estimated the number of employed AGVs, we have conducted additional simulations in order to define the impact of the capacity of the sorter on the system performance.

The diagram shown in the figure 4(b) displays the warehouse performance using five AGVs and varying the processing time per pallet of the sorter. We can deduct that in this case the layout of the warehouse influences the performance of the system. In fact increasing the sorter processing time the first places (those on the right side of the sorting area in figure 2) are emptied more slowly forcing AGVs to deliver their pallet in the places positioned in the middle of the sorting area. Since these latter places are closer to gates than the previous ones, the unloading time decreases. This phenomenon may be observed in figure 4(b) for the first 4 experiments (with 0, 30, 60, 90 seconds of processing time per pallet). When the processing time of the sorter exceeds 90 seconds per pallet, the sorter begins to saturate thus causing longer waiting queues and consequently increasing the unloading time.

It is worth to note that 90 seconds is about the time spent by an AGV to carry a pallet from the gate to the sorter. This time is obviously a critical time for the whole system.

These allow us to highlight that the actual policy of allocation of sorter places (that starts always on the right of the sorting area) is far from optimum. As we previously said the optimization of the warehouse layout is out of the scope of the paper, nonetheless we can still use this simulation to suggest a better sorter allocation policy which prefers the middle sorter places when it is possible.

Moreover these simulations have highlighted that there are some critical elements in the given warehouse layout. As a matter of fact, there are some paths that are busier than some others (busier paths are located at the rightmost side in figure 2) and less used paths (those on the leftmost side). This is caused by the actual sorter allocation policy.

Finally, figure 4(c) shows the throughput of the warehouse. This is measured by computing how many pallets are unloaded per minute. This number depends on the number of AGVs and the number of containers to be simultaneously unloaded. In the cited figure, we can observe that the throughput for five AGVs is about 2,8 pallets per minute while using more than five AGVs we can obtain only little improvements. In fact, increasing the number of AGVs the throughput goes towards three (3,1 pallets per minute with 8 AGVs). This diagram highlights once again that adopting five AGVs is a reasonable choice.

## VI. CONCLUSION

Multi-agent simulation is proposed as a tool for making decision about logistic problems. The multi-agent model adopted was tested for the simulation of real warehouse layouts. We have pointed out that the structural constraints of the given warehouse configuration limit the productivity. We have highlighted these limits and consequently we have made some considerations about the employment of resources.

We are currently exploring other resource allocation strategies for paths and sorter places in order to improve the performances of the system. Moreover at the moment we are working on the development of more effective strategies for a better exploitation of AGVs capabilities.

Moreover we are also prefiguring the application of an extension of our system to the study and optimization of the warehouse structural design including the number and position of gates as well as the internal layout.

## ACKNOWLEDGMENT

This work has been partially funded by the IMPULSO *Integrated Multimodal Platform for Urban and extra urban Logistic System Optimization* project funded by the Italian Ministry for Economic Development and coordinated by Vitrociset S.p.A.

A special thanks to Dr. M. Bordin of Vitrociset S.p.A. for the extremely fruitful technical discussions about the details of the IMPULSO project and the related issues of logistics.

## REFERENCES

- [1] M. Wooldridge and N. Jennings, "Intelligent Agents: Theory and Practice," *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.
- [2] M. J. Wooldridge, *Introduction to Multiagent Systems*. John Wiley & Sons, Inc. New York, NY, USA, 2001.
- [3] F. Klügl, "Agent-based simulation engineering," *Habilitation Thesis, Würzburg University, Germany*, July 2009.
- [4] F. Klügl, R. Herrler, and M. Fehler, "Sesam: implementation of agent-based simulation using visual programming," in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. ACM, 2006, pp. 1439–1440.
- [5] S. Tisue and U. Wilensky, "Netlogo: A simple environment for modeling complexity," in *International Conference on Complex Systems*. Citeseer, 2004, pp. 16–21.
- [6] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "Mason: A multiagent simulation environment," *Simulation*, vol. 81, no. 7, p. 517, 2005.
- [7] N. Collier, "Repast: An extensible framework for agent simulation," *The University of Chicagos Social Science Research*, vol. 36, 2003.
- [8] J. Swaminathan, S. Smith, and N. Sadeh, "Modeling supply chain dynamics: A multiagent approach\*," *Decision Sciences*, vol. 29, no. 3, pp. 607–632, 1998.
- [9] T. Lee, N. Park, and D. Lee, "A simulation study for the logistics planning of a container terminal in view of scm," *Maritime Policy & Management*, vol. 30, no. 3, pp. 243–254, 2003.
- [10] M. Fox, M. Barbuceanu, and R. Teigen, "Agent-oriented supply-chain management," *International Journal of Flexible Manufacturing Systems*, vol. 12, no. 2, pp. 165–188, 2000.
- [11] A. Karageorgos, N. Mehandjiev, G. Weichhart, and A. H. "ammerle, "Agent-based optimisation of logistics and production planning," *Engineering Applications of Artificial Intelligence*, vol. 16, no. 4, pp. 335–348, 2003.
- [12] B. Burmeister, A. Haddadi, and G. Matylis, "Application of multi-agent systems in traffic and transportation," in *Software Engineering. IEE Proceedings*, vol. 144, no. 1. IET, 1997, pp. 51–60.

- [13] R. H. Bordini, J. F. Hübner, and M. J. Wooldridge, *Programming multi-agent systems in AgentSpeak using Jason*. Wiley-Interscience, 2007.
- [14] A. Rao, "AgentSpeak (L): BDI agents speak out in a logical computable language," *Agents Breaking Away*, pp. 42–55, 1996.
- [15] M. d'Inverno and M. Luck, "Engineering agentspeak (l): A formal computational model," *Journal of Logic and Computation*, vol. 8, no. 3, p. 233, 1998.
- [16] A. Rao and M. Georgeff, "Bdi agents: From theory to practice," in *Proceedings of the first international conference on multi-agent systems (ICMAS-95)*. San Francisco, 1995, pp. 312–319.
- [17] E. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

# A Multi-Agent Architecture for Simulating and Managing Microgrids

Massimo Cossentino, Carmelo Lodato,  
Salvatore Lopes  
Istituto di Calcolo e Reti  
ad Alte Prestazioni  
National Research Council of Italy  
Viale delle Scienze, ed. 11  
90128, Palermo (Italy)

E-mail: (cossentino, lodato, lopes)@pa.icar.cnr.it

Marcello Pucci, Gianpaolo Vitale  
Istituto di Studi sui  
Sistemi Intelligenti per l'Automazione  
National Research Council of Italy  
Via Dante Alighieri, 12  
90121, Palermo (Italy)

E-mail: (pucci, vitale)@pa.issia.cnr.it

Maurizio Cirrincione  
Université de Technologie  
de Belfort-Montbéliard  
90100, Belfort (France)

E-mail: maurizio.cirrincione@utbm.fr

**Abstract**—With the increasing demand for electric power, new theories have been studied by the scientific community. One of the most promising consists in splitting the electric grid in microgrids, each one composed by renewable and not renewable sources and various loads. These microgrids aim to be as much autonomous as it is possible in producing the energy they need. Energy once produced must be transferred to the loads. This paper proposes a MAS used to simulate the control of the transportation grid. The system is able to react to feeders overloading and failures by redirecting the energy flow and protecting itself.

## I. INTRODUCTION

**N**OWADAYS Microgrids (MG) are expected to contribute to an improved energy efficiency and power supply reliability as well as an increase in the use of renewable energy [1],[2], thanks to the role of Renewable Energy Sources (RES) and power electronic in supplying clean electric energy.

A microgrid encompasses a portion of an electric power distribution system that is located downstream of the distribution substation, it includes a variety of distributed generation (DG) and distributed storage (DS) units, and different types of end users of electricity and/or heat. The microgrid presents an electrical connection point to the utility, known as the point of common coupling, generally it is located at the low-voltage bus of the substation transformer. Several customers can be served by a microgrid as residential buildings, commercial entities, and industrial parks.

Among the new issues there is the optimal generation schedule of DG sources aimed at minimizing the production costs and balancing the demand and supply which comes from RES and distribution feeders.

The management philosophy is crucial for the MG features exploitation [3]. Multi-agent systems have been proposed to provide intelligent energy control and management systems in microgrids. Multi-agent systems offer their inherent benefits of flexibility, extensibility, autonomy, reduced maintenance and more. As a consequence, the design and implementation of a control grid based on multi-agent systems, that is capable of making intelligent decisions on behalf of the user, has become an area of intense research. In a previous work we studied

some different policies that can be adopted to fully exploit the contribution of renewable sources and the accumulation system to the management of a microgrid [4]. The paper is based on the idea of creating an e-market for energy where sources and loads participate in a collaborative way.

The aim we pursue with the simulation proposed in this paper lies in a different scope: we want to study the distribution of electrical power in a MG taking into consideration the topology of the MG (feeders, nodes, protections), the impedance of the single feeder and the corresponding voltage drops and joule losses. As a matter of fact the inadequate management of the power network may cause an unstable behavior of it with consequent blackouts on a large scale. We suppose the grid is composed of totally passive feeders (as they are in real word) and of intelligent connection nodes. Such devices allow for the runtime connection/disconnection of their feeders. This gives to the Electrical System manager the possibility to act on each single branch of the grid thus avoiding system breakdown chain effect.

## II. PROBLEM DESCRIPTION

The goals that are at the basis of the proposed approach are to overcome the limits of a centralized approach to the management of energy flow on a large scale. Each MG will maintain an internal e-market to assign energy produced from sources under its control to the loads it has to take care of. Each MG may include renewable power sources (wind turbines, photovoltaic arrays), non renewable power sources (conventional plants), power storage devices (for instance super-condensers, fuel cells ...), and, finally, loads (industrial, residential and public emergency services). Briefly speaking, the management process is composed by the following steps. Initially, at each discrete simulation time step, a verification of the power balance is performed. This is a crucial step of the work: if the power produced by internal sources (plus what is made available for use from storage devices) is sufficient to provide power to internal loads, the MG is autonomous and it can decide whether to store or to sell the surplus of energy. The policy of energy storage is a sensitive one and we already



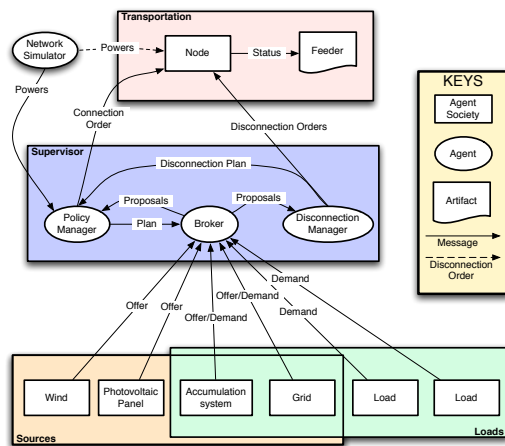


Fig. 1. The agent societies constituting a simple microgrid

made some studies about that [4]. This policy, however, is out of the scope of this paper. If the MG is not autonomous it can buy energy from the grid. The availability of energy is not a sufficient condition for the successful solution of the problem. In fact, energy has to be conveyed from the generator (power sources or storage devices) to the requiring loads. Energy transportation induces a loss of power for thermal loss due to the internal resistance of feeders. Sometimes it may happen that the power produced by a source is sufficient to feed a load but a feeder along the path connecting the two is not able to transport the required amount of power. If this is the case, our system is able to disconnect some feeders (the overheated ones) and if this is not sufficient, it may even disconnect some loads. Disconnection of loads is made according to a priority list that has at the top public emergency services, followed by industrial loads and finally residential ones. Disconnection starts from lower priority loads and it is performed by means of the intelligent nodes. If a feeder (or a load) is disconnected, it is reconnected when the conditions (for instance flow of power) allow that or after a fixed amount of time (for allowing heat dispersion and temperature drop in the feeder). In the next subsection the MAS architecture used to realize this approach will be introduced.

### III. THE PROPOSED AGENT-BASED SOLUTION

From the software architecture point of view, each MG can be regarded as composed of a society of agents. This society is in turn composed of local sub-societies (each sub-society can be a society of agents itself), each one modeling one of the main elements of the cell with the adjunction of a Supervisor society responsible for ensuring a strategic supervision of the energy flow in the cell. The local management of power flow inside each cell element (source/load) is left to single societies responsible for the cell element itself. These societies are self-interested. In the figure 1 we can see the following four agent societies: Loads, Sources, Supervisor, Transportation.

The **Supervisor society** is composed of three agents: Broker, Policy Manager, and Disconnection Manager. The

*Broker* agent is responsible for the brokerage between energy consumers and suppliers. The *Policy Manager* decides: (i) how much power should be provided by each source; (ii) if a battery should recharge or discharge; (iii) if the grid is going to sell or buy energy to/from the cell. In the actual implementation of the decision process, the agent includes a rule system that optimizes energy flow in terms of cost [4]. The *Disconnection Manager* agent is responsible for applying the established disconnection plan of loads or adapting it as a consequence of unforeseen events. It is worth to note that the proposed architecture, spontaneously responds to blackouts propagation since it gives priority to the independence of the cell and it asks for power or provides power to the remaining part of the grid only if necessary/available.

The **Sources Society** is composed of all cell elements that can generate power. Each of them is a society of agents too. Since some elements (e.g. the batteries) can sometime generate and other times consume power, these elements are both members of the Sources and Loads societies. When an element provides power it plays the role of Producer in the Source society. When an element buys power, it plays the role of Consumer in the Loads society. Typical members of a Sources society belong to renewable and non renewable sources, batteries (while providing power), and the grid. A more extended discussion may be found in [4].

The **Loads society** is composed of cell elements that consume power [4]. They can be actual loads as well as power accumulation elements.

The **Transportation society** is composed of the Node sub-society and the feeder artifact. According to the approach we adopted, transportation is managed by way of intelligent nodes that can connect/disconnect each feeder they control, according to a connection plan used to minimize transportation losses of power. Several instances exist at runtime of the Node sub-society, one for each actual node of the MG. Each Node sub-society is composed of the below discussed agents. The *Local Manager* agent is responsible for the node management. It communicates with the Policy Manager in order to receive expected configuration data (expected power levels, connection status) for every feeder connected to the node. According to this data, the Local Manager agent orders the connection/disconnection of each feeder to the Connector agent (see below). The *Connector* agent is responsible for physically controlling the node connections with feeders. The *Monitor* agent is responsible for reading real power data from the MG. In the proposed simulation experiment, it interacts with the Network Simulator agent in order to obtain a good approximation of what the behavior of the real MG would be. If the power read in the connection of some feeder differs from the expected one of more than a specific threshold (set to 5% in the proposed experiment), this agent sends a message to the Local Manager. The *Network Simulator* is responsible for the simulation of the MG that is obtained with the application of the Newton-Raphson algorithm.

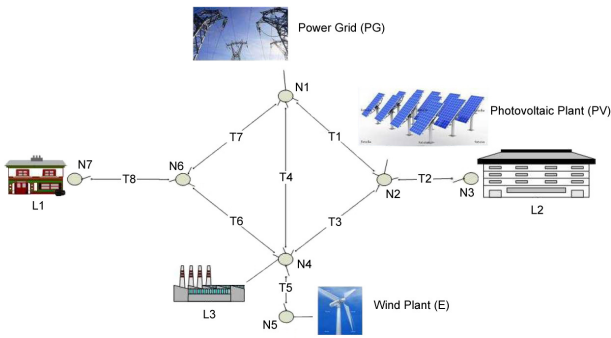


Fig. 2. The microgrid used for the simulation experiment

#### IV. SIMULATION SETUP AND RESULTS

In the simulation the time is discrete and we mainly worked with hours since we wanted to study the steady state of the system, not the transitory state. At the beginning of each simulation, the Broker agent receives the requests of buying/selling power from the MG loads and sources. This information is used by the Policy Manager agent to define a plan used for satisfying the market requests at the specific time slots. This plan is enacted by the Broker agent. That, if it is not possible to satisfy all the requests, delegates the Disconnection Manager to command the disconnection of lower priority loads. The Disconnection Manager defines which loads may be disconnected and it orders to the related nodes to perform this action. If the available power is sufficient, the Policy Manager requests to the Network Simulator a computation of the power flows in each of the nodes of the grid, according to their current connection status. If the flows do not violate the feeders limit powers, this plan is passed to the Nodes of the Transportation society in form of Connection Orders. Each Node sets the status of the feeders connected to it. During the time slot, the Network Simulator agent is queried by the Monitor agent of each node in order to simulate the reading of the power flowing through the feeders connected to the node. If some feeder is supporting a power that is beyond its rated power, the disconnection policies occur.

During the proposed experiment, the system reacts to an overload occurring on some feeder of the MG. This demonstrates the self-protecting feature of the MG that disconnects the feeder and redirects the power through other branches of the grid. Of course if it not possible to find a path, the load requiring such a huge power must be disconnected.

The simulation scenario concerns a microgrid that includes eight feeders which are labeled T1-T8 and seven nodes labeled N1-N7 (Figure 2).

To make the simulation more realistic, some sets of historical data about wind speed and solar irradiation have been used. Similarly, load characteristics used to feed the simulator have been obtained by actual and historical sets of data coming from industrial and residential areas.

The reported experiment consists in a simulation in which

the MG works for about a day and a half, starting at 05:00 and ending at 18:00 of the second day; this interval of time has been chosen to better show the behavior of some components. For instance, in Figure 3 it is possible to note the increase in the electric power supplied by the photovoltaic source (PV), and how it reaches a peak on midday. It is also possible to note as the power required by the domestic installations L1 and L2 appears congruent with the expectations, highlighting a periodical behavior and a power drop during nighttime (from 03:00 to 06:00). Instead, the industrial load never goes below the 30 Kw of power.

Each feeder in the MG is oriented, and the positive direction for the power flow is assumed to be from the node labeled with the smaller index towards that one with the greater one.

Figure 5 shows the substantial congruence between the power transmitted by the border feeders (T2, T8, T5) with the values of L3, L2 and E (Wind Turbines) respectively. Some minimal differences may be noticed and they are due to feeder losses. The negative values on T5 feeder indicate, as said before, a flow from N5 towards N4.

Looking at the other curves in Figures 3, and 4, it can be seen that from 14:00 the demand by the loads increases up until 19:00 of the first day; simultaneously the power supplied by the photovoltaic system decreases. As a consequence, the PG (Power Grid) must increase the supply, except when E has a peak of power. These events cause that the load in the three feeders coming out from N1 (T1, T4, T7) increases too.

When the electric power on a feeder overcomes the nominal value but it remains below the short circuit value, the PolicyManager agent starts a control action. The agent, in order to evaluate the disconnection condition, uses a rule that measures how long the nominal power has been exceeded. This is what happens in the interval time ranging from 08:00 to 16:00. The PolicyManager agent calculates that the threshold of the nominal power, of feeder T4, has been exceeded beyond the time allowed for that and then it sends the disconnection command to the LocalManager agents of the nodes N1 and N4. So, at 16:00 of the first day, the feeder T4 has been disconnected and the power, coming from the PG, has to be redistributed between the remaining feeders. Consequently, the load increases not only on feeders T1 and T7 but also on T3 and T6. The maximum stress on the feeders occurs around 21:00 of the first day simultaneously with the maximum load conditions. The negative values of the electric power on the feeder T6 indicates a flow directed from N6 towards N4.

From the 21:00 of the first day onwards, until about 06:00 of the second day, the overall power requested by loads decreases thus causing a reduced power withdrawal from the PG and thus a lightening of the load on the feeders. In fact, at 02:00 of the second day, The PolicyManager agent authorizes the nodes N1 and N4 to the reconnection of the T4 feeder.

At 3:00 of the second day there is a drastic drop in power in the loads L1 and L2, which reduces from 26 kW to 3 kW; this event affects the supply of energy from the grid and gives a decisive contribution to the maintenance of the connection feeder T4. In fact, the power peak that happens when feeder

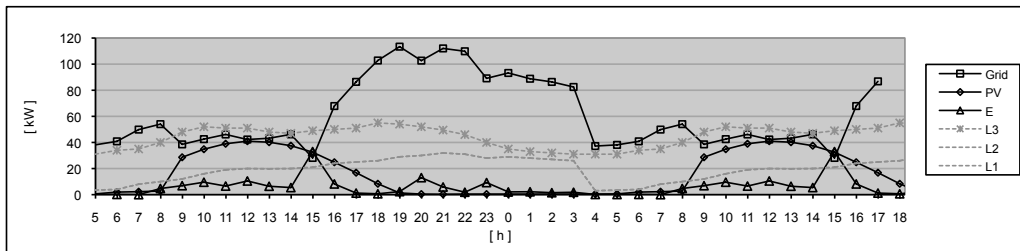


Fig. 3. The power produced by the sources and provided to loads (L1 and L2 curves overlap each other)

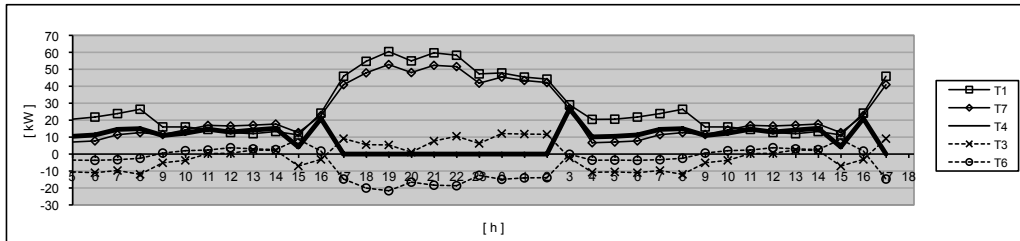


Fig. 4. The power passing through some feeders

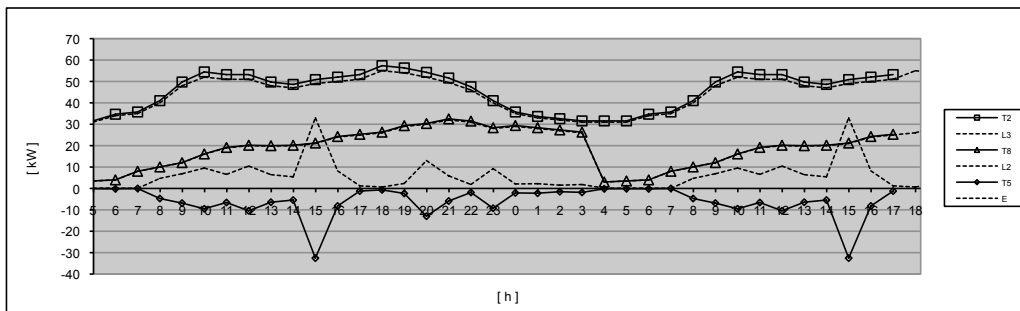


Fig. 5. The power levels in loads and in border feeders

T4 is reconnected wears off quickly to avoid overheating and avoiding a new cable disconnection.

In conclusion, the simulation has shown how the system privileges the absorption of energy from renewable sources compared to the PG, and how it can perform a dynamic reconfiguration of the MG as a result of the overloading of a feeder with corresponding disconnection of some feeders.

## V. CONCLUSIONS

In this paper we have proposed a MAS-based approach for the solution of the energy transportation problem providing a system that is able to react to feeders overloading and failures by redirecting the energy flow and protecting itself.

From the infrastructural point of view, this would imply the adoption of intelligent nodes in the grid. Such nodes enable the dynamic connection/disconnection of feeders thus allowing

the redirection of energy flows as well as the disconnection of loads that may cause overloading problems to the grid.

The system, here presented, assigns a relevant level of decision autonomy to the microgrids thus creating a perfect scenario for the adoption of distributed agent-based solutions.

## REFERENCES

- [1] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, "Microgrids management," *IEEE power energy magazine*, vol. may/june, pp. 54–65, 2008.
- [2] G. Venkataraman and C. Marnay, "A larger role for microgrids," *IEEE power energy magazine*, vol. May/June, pp. 78–82, 2008.
- [3] S. Chakraborty, M. D. Weiss, and M. Simoes, "Distributed intelligent energy management system for a single-phase high-frequency ac micro-grid," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 97–109, 2007.
- [4] M. Cirrincione, M. Cossentino, S. Gaglio, V. Hilaire, A. Koukam, M. Pucci, L. Sabatucci, and G. Vitale, "Intelligent energy management system;" in *Prof. of 2009 IEEE International Conference on Industrial Informatics (INDIN 2009)*, 2009.

## Agent.GUI: A Multi-agent Based Simulation Framework

Christian Derksen  
DAWIS  
University of Duisburg-Essen,  
Schützenbahn 70,  
45127 Essen, Germany  
Email:  
christian.derksen@icb.uni-due.de

Cherif Branki  
School of Computing,  
University of the West of  
Scotland, Scotland  
Email: cherif.branki@uws.ac.uk

Rainer Unland  
DAWIS  
University of Duisburg-Essen,  
Schützenbahn 70,  
45127 Essen, Germany  
Email:  
rainer.unland@icb.uni-due.de

**Abstract**—Multi-agent based simulations (MABS) of real world scenarios are attracting growing interest. Complex real world scenarios require deep knowledge and expertise which can only be provided by specialists in the application area. However, it cannot be expected that such experts understand agent-based technology and simulation. Consequently, tools are required, which deliver a high level, easy usable interface.

In this article we propose a new simulation framework based on the JADE framework. Besides extensions to deal with the time aspect, agent/environment interaction, visualization and load balancing, we also address the usability of the tool for specialists from different domains. For this, our framework, called Agent.GUI, provides an easy to use, customizable graphical user interface. Overall, Agent.GUI is a powerful tool for the development of multi-agent based simulations.

### I. INTRODUCTION

Multi-agent based simulation (MABS) has been receiving increasing interest in the recent years. One reason for this can be seen in the fact that the agent-paradigm allows the mapping of real world entities to autonomous software agents as a first approximation. Exploiting further skills of an agent, like the ability to communicate, learn or reason, can result in further benefits and new solutions [6], [18]. In fact, MABS as a sophisticated alternative to traditional simulation techniques attracts growing interest in a broad range of disciplines. Examples for their practical as well as scientific deployment are traffic simulations [16], crisis management, energy markets or scheduling problems [10], [15], [11].

But complex scenarios from different domains often bring their own complexity with them. It can not be assumed that these domain-specific experts can understand, build or even control the execution of an agent-based simulation - simply due to its inherent complexity.

In this article we propose a new simulation framework that is based on JADE [2]. On the one hand, JADE is extended by specific functionalities for simulation purposes like time and agent synchronization, agent/environment interaction, visualization and load balancing in order to simplify the work for an agent-based developer on such concerns as much as possible.

On the other hand, a main focus is on users, who are not familiar with multi-agent systems or distributed simulations. Here our framework provides a multi-language based GUI that can easily be customized to domain specific requirements.

In this paper we will focus on two important aspects of our framework, the ability for extending the frameworks graphical user interface and the bidirectional interaction of agents with their environment by using our adapted service for simulations.

This article is structured as follows. The next section will give some background information and will motivate our work, while section 3 will present the above mentioned capabilities of our framework. In section 4 we will show an application, which compares the use of our simulation service to the use of ACL for the agent/environment interaction. Section 5 presents the related work. Finally, section 6 concludes the paper.

### II. BACKGROUND AND MOTIVATION

Agents can be regarded as autonomous, problem-solving computational entities with social abilities that are capable of effective proactive behavior in open and dynamic environments. There are a number of definitions for agents (e.g. [17], [12]) and most of them are associating the properties autonomy, social ability, reactivity, proactively and intelligence to agents.

Considering reactive or proactive behaviors of agents, agents can exhibit different levels of sophistication. Literature discusses different types of sophistication (e. g. deliberative, learning or simple reactive one) which we do not want to repeat here.

A Multi-agent system (MAS) is a loosely coupled set of agents which were composed in order to solve problems that monolithic systems (or single agents) can not solve. In order to find the solution of a problem, the agents have to rely on communication, collaboration, negotiation, responsibility delegation and trust Also this subject is discussed in detail in the literature [18].

### A. JADE - Java Agent DEvelopment Framework

JADE<sup>1</sup> is a Java-based framework, which allows developers to implement agent-based systems. JADE is designed as a middleware platform that provides the runtime environment for implemented agents.

The framework provides relevant basic classes that are to be extended to adapt it to the specific need of the application in question. For example the base element agent can hold states and knowledge, which can be internally changed by using pluggable behaviors of different base types. These base behaviors in turn, can be composed to complex parallel or sequential behaviors, which can be hierarchically organized.

For communication purposes JADE provides a FIPA-based message mechanism, which relies on an asynchronously working message-box mechanism at the receiver of a message. Message contents can be composed by using simple textual information, complex but serializable content-objects or parts of individual ontologies. The latter can be built in order to provide a domain specific vocabulary for the involved agents [2], [5].

JADE offers several optional services, which can be configured by explicitly addressing their use in the parameter set for the platform start. Here are for instance services available for agent mobility, fault tolerance or the FIPA<sup>2</sup> compliant DirectoryFacilitator. JADE services can be considered as an intermediate layer between platform and the abode of the agent and are especially useful if information have to be shared over the whole platform. For more specific requirements JADE allows to build individual services.

Agents reside in so called agent-containers on the top of the platform. With the initial start of JADE the Main-Container will be started as well. In order to extend the platform to a distributed system, an administrator can start other JADE instances on remote systems. Defining the necessary set of parameters, the remotely started JADE instance will join the platform during runtime by adding a new container to it, which results to the fact that control over the remote system is required.

With respect to our work, we would like to mention here that JADE does not provide any specific support for MABS like for example the definition of a central and synchronized environment model or the measurements of the current system load. To use JADE for the development of such a system means to start from scratch.

### B. Modeling activities for a Multi-agent based simulation

The modeling activities that developer has to face in order to provide an MABS to end users are manifold. In order to not overload this subject, the most important aspects are discussed here. For a comprehensive discussion it is referred to the literature [3], [7], [8].

Figure 1 below provides a rough overview to the elements on which developers have to work on. It is to recognize, that the development has to focus on several main parts. These are, in order to their importance and their influence on a simulation: the environment model, the scheduling of the simulation and the Multi-Agent system itself.

Since agents, by its definition, acting in their environment, one of the first things to be developed for a MABS is the environment model. According to the suggestions of Russel and Norvig [14] environments can be classified into different types considering aspects like accessibility and determinism. Furthermore they can be static or dynamic, discrete or continuous and episodic or non-episodic.

These classifications imply already the presents (or the absence) of time and shows the close relationship between the scheduling of a simulation in conjunction with the environment. Independently of whether this model has to be visualized or not - which additionally increases the developing effort - the type and the data model for it has to be defined first.

Additional modeling effort is also required for the scheduling of the simulation, which can be in principle event based, time-driven or a mixture of them. Furthermore, time dependent simulations can be either continuous, discrete or hybrid. In case of continuous simulations, each time step is very small, which results de facto in a continuous system behavior. Parts of the simulated system can be for example modeled and described through differential equations, which can be used in order to calculate a time dependent system reaction. Discrete simulations are using the time to look for statistically or randomly sized time intervals to cause certain events. These events will determine the (next) state of the system. Additionally, a simulation can be seen as hybrid, if the model has properties which are either continuous or discrete [6], [10], [13].

After defining the environment model and deciding which scheduling strategy has to be used, the Multi-Agent system has to be built. Every autonomous entity, object and relationship between them has to be modeled and, later on, to be implemented. From the MAS-specific perspective this means that:

- agents have to be identified and their types have to be laid down (e. g. deliberative or learning agent as well as predator or pray agent),
- necessary agent behaviors and their compositions have to be defined,
- communication and all other protocols have to be set as well as a specific ontology,
- negotiation and collaboration have to be considered.

One of the last points to be mentioned here is the interaction between environment and agent. Since an agent can act in its environment, the environment, in turn, may react or respectively act on the agent. If this is an inherent

<sup>1</sup> <http://jade.tilab.com/> and [2]

<sup>2</sup> <http://www.fipa.org/>

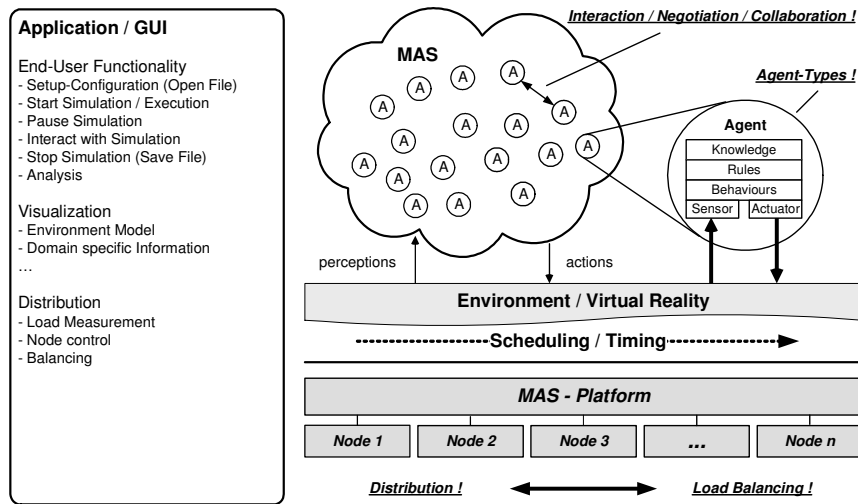


Figure 1: Elements in MABS

point of a simulation, it has to be modeled and implemented as well.

Beside this and considering a simulation, which is usable for experts of different domains, developers have also to focus on graphical user interfaces for the configuration and the interaction with the simulation.

### C. Execution of distributed Simulations

The main intention for distributing a simulation is scalability and reduction of local workloads, in order to speed up the simulation, or to allow the simulation to be bigger in terms of the number of calculating nodes and agents. Thus, on the one hand, a simulator needs to be able to spread its simulation to an arbitrary number of computers while, on the other hand, the load over all nodes needs to be balanced in a meaningful way.

This is basically a statement that indicates a further aspect of our tool. The inherent problem of load balancing is widely discussed and not only a subject in the field of simulations. The number of publications is very high so that its treatment requires a separate discussion, which can not be covered in this presentation of our framework.

### D. Usability for Developers and end users

Building simulation software that enables end users to work with Multi-agent based simulations, results in dealing with a set of further requirements, which exceeds the needs for “simply” providing a development framework. While a framework and its elements are to be well and transparently documented for developers, the end user application additionally has to be extendable, in order to match the user’s demand. Also it should be easy to understand and use. However, neither usability studies [9] nor agent-human interaction will be discussed in this paper. Nevertheless, it needs to be mentioned, that one motivation for the tool is to be seen in finding an optimum between a support for MAS

or MABS developers while, additionally, keeping the predefined user interface as open as possible for all kinds of applications.

Overall we described here the difficulties and the enormous effort, which developer has to face in order to build a MABS from the scratch and providing such simulations to end users.

From this point of view it is certainly not possible, that we present a full description of our framework in this paper. Consequently, in the next section we focus on a general description of our framework, explanations on how the user application can be extended and the use of our simulation service, which is build for a bidirectional agent/environment interaction.

## III. AGENT.GUI - INFRASTRUCTURE AND USAGE

Agent.GUI enables developer to create a JADE-based MAS application, which can be directly used by the end users through the prearranged application window of our framework. This end user application is able to control JADE, which means that end users can manage the JADE platform and their agents as well as the developed MABS.

The Agent.GUI end user application is based on Java-Swing. This multi language tool allows the handling of JADE agencies by considering the developed MAS and its resources as an encapsulated system, called project. Such a project can be configured within the application and can afterwards be executed and distributed.

### A. General Functionalities for agent projects

Agent.GUI handles a JADE MAS and its resources as a project. Starting from an already fully developed multi-agent system with agents that are able to meet demands placed on them, the application requires only some information relevant for the handling of the MAS and its resources. To



get control over these resources, an Agent.GUI project has to be defined and configured initially. In the context of a software lifecycle, we see this as the final step, which has to be done by the developer of the MABS. After this the system should be ready for use by the end user.

Besides assigning a general project name and some textual information the final configurations that a developer has to provide can be done as follows:

- The framework offers the usage of one of two predefined environment model types. This can be, up to now, either a continuous two-dimensional environment or a graph that can act as a central environment model. Both models have already a tailored visual representation which will appear according to its selection. This topic we are planing to discuss in a different presentation.
- External Resources can be picked from the local file system. There it can be chosen between compiled jar-files or complete folders (e.g. bin-Folders of an IDE), which will be added dynamically to the Java-Classpath at runtime.
- Extending and customizing the *PlugIn*-class of our framework, the developer can add her/his own GUI elements to a project (see paragraph *B* of this section).
- From our point of view an ontology can be more than just the central element for the communication of agents. It can also provide comprehensive domain models for a simulation. To use them within the application, the developer can select and add them to the project. Agent.GUI offers a reflective, graphical access to selected subparts of the ontology, so that classes can be initialized and filled with specific values. The ontology is to be created by the BeanGenerator of Protégé<sup>3</sup>.
- JADE-Agents can be run by using start arguments. During the instantiation of an agent they can be passed to it as a simple array of objects. Knowing the required object types and their order for a single agent, the developer can assign this information to the project-definition for later use by the end users. The object types can be selected from the underlying ontologies for the project.
- During the JADE-Configuration the needed base services are to be identified and the definition of the port for the JADE middleware is to be done.

After these above mentioned configurations the MABS should be ready to use for end users and the work for developer is done. Since an end user is meant to use the JADE Multi-agent System for her/his own purpose Agent.GUI enables her/him to configure different start setups for the agency. Furthermore, if selected, one of the predefined simulation environments can be configured in order to apply the project agents to different situations and environments.

### B. Programming interfaces for developer

Developers can use Agent.GUI and their libraries with their IDE simply by adding the core jar-file to their own project as an external library. From here on, the programming interfaces for customizing a MABS and its visualization can be separated into three types: interfaces of the Agent.GUI framework, external interfaces that are coming “naturally” with the application, because they are integral components of the Agent.GUI project (e. g. the JADE libraries) and external resources, which can be individually added to the MABS during the development phase. Such external resources have to be added by selecting them as a part of the IDE and, then, configuring them as jar-resources in the specific Agent.GUI project.

In order to access the object structure of our framework the singleton class *agentgui.core.application.Application* can be used. Starting from this point, developer can access everything the framework provides. Additionally, the framework provides a translation interface in order to allow developers the use of the language of their choice in the source code. The API as well as some tutorials are available with the framework resources.

In case of a needed customization of the visible program, Agent.GUI allows the extension of the main application window and its elements as well as the extension of the project window for the MABS. For this, our *PlugIn*-mechanism and the extendable *PlugIn*-class were designed. The *PlugIn*-class provides access to menus, the main toolbar and allows adding further tabs to the project window. If the customized *PlugIn*-class is configured in the project, the tailored elements will be automatically added to the visual program. The following screenshot in Figure 2 shows the example of an extended application window and some additional tabs which were added to the project window.

Beside this developers can react on application events. For this the project data model and its visual representation was designed by using the common MVC pattern. Individual tabs for the project window as well as the configured *PlugIn* classes will be informed about changes in the project settings or if a simulation is to be started.

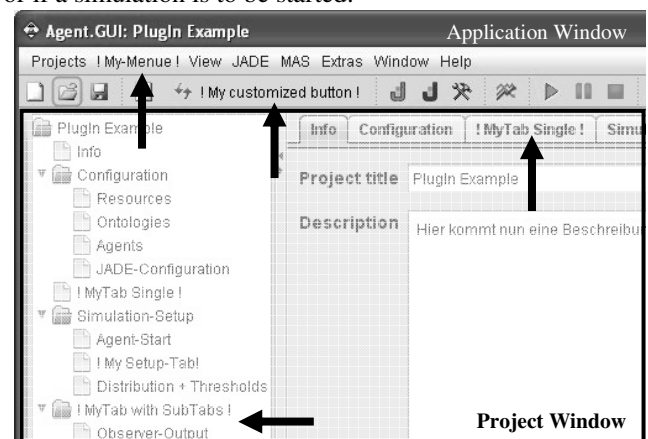


Figure 2: Extended GUI for MABS

<sup>3</sup> <http://protege.stanford.edu/>



### C. Simulation Service

One motivation to build our Simulations Service can be seen in the handling of agencies, with a large number of agents. Such scenarios can be found for example in the current research topic of smart grids and intelligent power supply. Here a big number of participants can be found due household, industrial consumer and power producers. They all rely on an interconnected network, which can be seen as their environment. Analyzing the connection between the participants, seen as agents, and their physical connection to the grid, it is obvious that not only the agents are working in their environment, by consuming electrical power. In turn, the grid delivers the electrical power or, if the supply doesn't meet the user demand, the generation can collapse. For an agent/environment interaction this leads immediately to a situation where an environment acts on agents too, which shows this bidirectional relationship. Testing such interaction with a large number of agents by using ACL messages, we found out that the JADE platform became overloaded, so that we had to find a different solution (see also section IV).

Based on an extended JADE base service we designed our Simulation Service for a bidirectional interaction between agent and environment, taking into account that scheduling strategies can differ, depending on the kind of simulation. Therefore we equipped the simulation service, which is particularly used in order to transport the environment model information to the agents, with a set of methods, which can be individually used depending on the specific simulation schedule.

In general we assume that at least one entity, in our case an agent, has to manage the information about the environment. This agent is to be assumed as the *SimulationMangerAgent*. Other agents, which are acting in this environment, our so called *SimulationAgent*'s, have to be connected to the environment due our *SimulationService* and its inherent sensor/actuator functionalities. Figure 2 shows this relationship and the herein used classes.

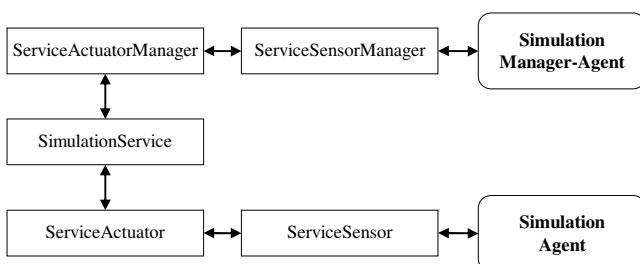


Figure 3: Classes used for the agent/environment interaction

Every agent is equipped with a so called *ServiceSensor*. This applies for the *SimulationMangerAgent* as well as for the other agents. Using the *SimulationService*, this sensor can be connected with the *ServiceActuator* on site; this is in fact the *ServiceSlice* of the JADE agent container, in which the agent is currently located. Colloquially as a metaphor,

one could consider this as the insertion of a plug into an electrical outlet.

In case that a discrete simulation steps forward or an event based notification for an agent occurs, an actuator can transport a new environment model or notifications to the connected sensor of the agent. In order to prevent the loss of the autonomy of the agents, we designed this stimulus in an asynchronous way.

There are several opportunities for the manager agent in order to schedule or organize the simulation process. There are for example methods available that are allowing to run the system in an episodic way (see example in section IV), while other methods allow to address single notifications to agents, which is relevant for event based simulations. Furthermore the *SimulationService* provides synchronized time to all (distributed) containers on the JADE platform; this can be used for timed simulations. The kind of scheduling is therefore still subject of the agent design and is not limited through our framework.

As a generalized environment Agent.GUI provides a class structure that consists basically of three sub parts, which allows describing the state of the environment. They are: (i) a *timeModel*, (ii) a domain specific *abstractEnvironment* and (iii) a *displayEnvironment*. In this a time model can be a simple counter, which steps forward with every state of the environment model (called *TimeModelStroke*) or it can be a concrete time, which can be changed for every state of the environment. The latter two attributes of the model are simple *Object* types, which allow applying a variety of structures to them. Hereby the abstract model should be used for general structures as they can be defined by ontologies, while the latter attribute should contain model information, which can be also displayed. Furthermore, using the so called *TransactionMap*, the simulation service is able to manage different states of the simulation over the simulated time.

### IV. APPLICATION: TESTING THE SIMULATION SERVICE

Since Agent.GUI is build on top of JADE its overall performance depends mainly on the JADE implementation. Nevertheless, a few optimizations were conducted because the simulation service relies on method execution instead of sending ACL messages for the agent/environment interaction. This let to a significant increase in the simulation speed, which was validated through the following small experiment.

Two MAS for a Game of Life (GoL) were implemented that consists of simple cellular automata (Figure 4). In order to evaluate the efficiency of our simulation service, one implementation used ACL messages for the agent/environment interaction; the other one used our *SimulationsService* introduced in the previous section.

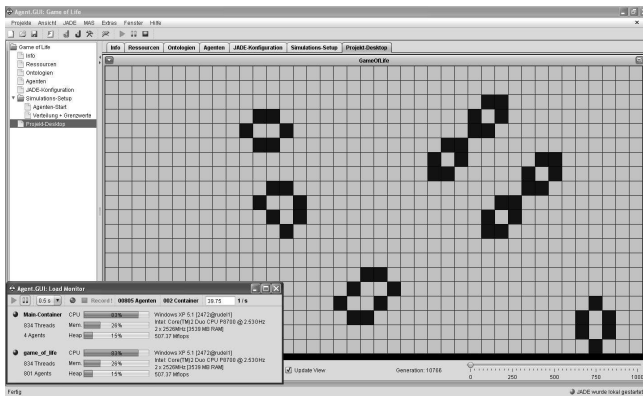


Figure 4: Game of Life in the Agent.GUI application window

In both cases a single agent represented one field while one agent was the manager of the environment, which in turn consisted of all area-agents of the playing field. At initializing of the GoL, the simulation manager created the visual representation first, which allows user to define the initial game situation, before actually starting the game. Executing the 'simulation' by the manager agent then starts the cyclic simulation of the GoL as shown in the next sketch below.

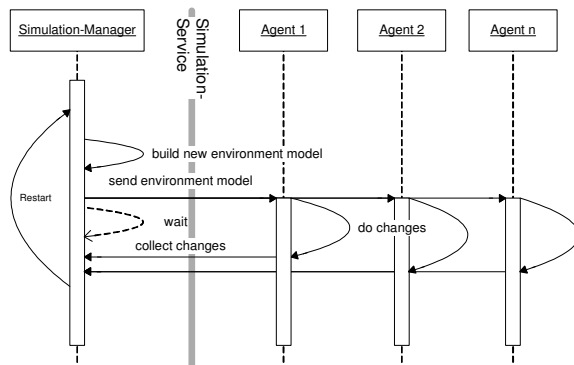


Figure 5: Simulation Cycle for the Game of Life

Collecting all changes from the visual representation and take them over into the private environment model of the 'Simulation-Manager', this model was send as new environment state to every agent. Knowing the name of the agents in the neighborhood, the field agents were able to get these states and calculate their own next state. This new state was send back to the manager agent, who builds up the new environment model and started the next generation of the Game of Life again. The grey line in Figure 5 indicates the use of the *SimulationService* instead of ACL messages in our benchmark study.

For this comparison the following system was used:

CPU: 2 x 2527MHz (Intel® Core™ 2 Duo CPU P8700)  
RAM: 3539 MB RAM  
OS: Windows XP 5.1  
Java: jdk1.6\_014

Executing between 20 up to 3000 field-agents in a single JADE-container, 5000 simulation cycles (generations) were done according to the sequence showed above. The time for the simulation cycles was measured and smoothed within the *SimulationManager*.

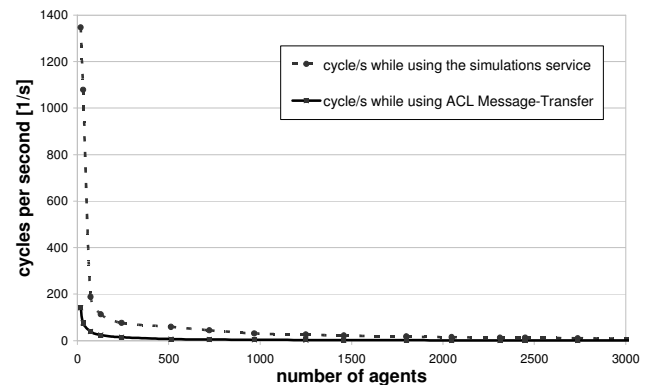


Figure 6: Simulation cycles and number of agents for agent/environment interaction

The measured increase for the agent/environment interaction was 9 times in average. Figure 6 above shows this comparison. Further tests showed that using ACL messages for the agent environment interaction leads more quickly to an overloaded system.

## V. RELATED WORK

To the best of our knowledge there are only very few extensions or tools with similar objectives than Agent.GUI for the JADE platform. We distinguish between those tools that are also based on JADE as agent platform and those that are providing their own agent concepts, but similar core functionality. To the first category belongs SIMJADE [10] or DisSimJADE [4].

SIMJADE is an extended BaseService for the JADE platform. It supports distributed simulations by providing an optimistic Time Warp based synchronization scheme. Furthermore, SIMJADE comes with some basic classes that can be extended to cater for individual needs. This is similar to our time and environment synchronization concepts. However, in contrast to SIMJADE, we have implemented no concrete synchronization process but open interfaces for different schedule mechanisms. Topics like load balancing, automated distribution of simulations, visualization techniques or usability for end users were not addressed with SIMJADE.

DisSimJADE was introduced by Gianni et al in 2009. This simulation framework enables the incorporation of distributed simulation facilities into agent-based systems. It was build on top of JADE. By using the general purpose architecture HLA (High Level Architecture), this framework enables the interaction (distribution of data and the synchronization of actions) between different simulation

systems that follow the IEEE standard<sup>4</sup>. In case of a distributed simulation DisSimJADE does not use the interfaces and infrastructure for distribution which JADE already provides. Instead, the standardized HLA-based communication structure is used for the interaction between distributed agencies. This was basically motivated by the interoperability to other simulation systems with similar interfaces.

The comparison of Agent.GUI to these two approaches for MABS shows that the idea of using JADE for simulation is not new. However, until now it was never implemented consistently in a comprehensive framework.

The category of general framework and end user aspects contains a big number of available tools for agent-based systems. Here we concentrate on programs, which consider an environment as an important factor for a MABS. Especially for such systems, Arunachalam et al [1] pointed out four main criteria for a comparison of such tools. These are in short:

- The design criteria, specified through the definition of an environment, the environment distribution and the coupling of agent and environment.
- For the model specification they rated the ease of specifying the environment, what features a system offers to define the environment and what knowledge an agent can have about this environment.
- With respect to the model execution the quality of visualization and the possibility of property modifications during runtime were considered to be most important.
- Finally, the quality of the available documentation was investigated.

On the basis of these categories *NetLogo*, *MASON*, *Ascape*, *RePast Symphony (RePastS)* and *DIVas* were examined. For a more extensive survey on tools for “Agent Based Modeling and Simulation Tools” (ABMS), we refer to the website of Ron Allan<sup>5</sup>.

The direct comparison to such frameworks and to the above mentioned JADE extensions discloses that there is still a lot of space for improvements for us. But in relation to the above mentioned criteria’s it is our opinion, that Agent.GUI can be seen as a competitive framework with some unique features. Table I, on the right, shows the result of our own assessment in comparison to the mentioned frameworks discussed in [1].

Since Agent.GUI was designed as a general purpose tool for MABS and because those kinds of developments are highly domain specific tasks, some of our ratings can not be clearly applied in the sense of the tool comparison of [1]. This applies for example at the agent and environment

coupling at the design criteria’s for environments. While our framework offers a generalized environment model (A.1.) that can be used in a distributed manner for more or less any kind of environment model, by using the *SimulationService* (A.2.), the coupling given through the bidirectional sensor/actuator relationship implies to be a very high one (A.3.). In the sense of the tool comparison this seems to result to the lowest rating possible there. As a developer, however, is free in the decision to use our service or not, we argue that this is essentially a question of the applied domain, the chosen MAS architecture and last but not least the desired agent/environment interaction. For this, with Agent.GUI, a MABS can be designed with a very low coupling between agent and environment as well, which would finally result to a rating of “Very High”.

TABLE I  
SELF-RATING OF THE AGENT.GUI FRAMEWORK

<u>Criteria</u>	<u>Rating</u>
<b>A. Design Criteria</b>	
1. Environment structured complexity	Very High
2. Environment distribution	Very High
3. Agent and Environment coupling	-
<b>B. Model Specification Criteria</b>	
1. Specification features offered	High / Very High
2. Programming skill of end user	Low / Low
3. Environment knowledge in agents	High / High
<b>C. Model Execution</b>	
1. Quality of visualization	High / Very High
2. Simulation view	Low / Low
3. Model property modification	Medium / High
<b>D. Documentation</b>	
1. Quality of documentation	Medium
2. Effectiveness of the documentation	-

With our framework the responsibility for providing tools for the definition of an environment is basically up to the developer. Nevertheless, since Agent.GUI provides two predefined environment model types (a continuous 2D model and a network model consisting of nodes and arcs), we rated criteria B.2. for both models in all conscience. As it is one of our main intentions that developers provide end-user applications, the aspect of programming skills for end users (not developers) is rated to “Low”. In relation to the above mentioned point of the agent/environment coupling, the aspect of the agent’s knowledge about the environment was rated to “High”. This basically depends also on our predefined environment model types and will differ depending on the concrete application.

Regarding the model execution in C., our framework uses basically the same classes and types for the visualization of a simulation as they were used for the definition of an

<sup>4</sup> IEEE Standard 1516: Standard for Modeling and Simulation High Level Architecture

<sup>5</sup> <http://www.grids.ac.uk/Complex/ABMS/ABMS.html>

<sup>6</sup> P2D: Physical 2D environment / NM: Network Model

environment; for this standardized interfaces are used. This is why we have here the same evaluation as in B.1 which relies on the connection with our predefined environment types. A 3D or toroidal simulation view, as it was asked for in the comparison of [1] (C.2.), can not be provided by our framework until now. Our rating in relation to the model modification during runtime has also to be seen in connection to our two environment models. Additional it should be mentioned here, that for the Game of Life agency for example this aspect must be rated with “Very High”, as a modification at runtime was even desirable, which shows again the strong dependencies to the domain.

The aspect of documentation under D. is one of the next important tasks for the development of our framework, as indicated by the relatively low rating (D.1.). Since outside of our group so far only a very few developers have worked with Agent.GUI, it is not possible to evaluate the effectiveness of the documentation at this time.

Due to the fact that Agent.GUI is based on JADE, our framework is compliant to FIPA standards too. The above mentioned design criteria is extensively addressed; especially in an active and bidirectional relationship between agent and environment. Regardless of our two predefined environment model types, we see a big advantage in the fact that Agent.GUI provides open interfaces for any kind of environment model and simulation scheduling, which allows a nearly unrestricted development (e.g. for 3D-models etc). Another benefit comes with the functionalities regarding the customizable load balancing approach, which we could not find in any other tool or framework for MABS.

## VI. CONCLUSION

In this paper we introduced a framework for Multi-agent based simulations called Agent.GUI that is built on top of the JADE platform. Based on our frameworks base-GUI it allows the programmer to realize a domain specific end user application for Multi-Agent based simulations. For this purpose Agent.GUI provides open and adaptive interfaces.

An example scenario that illustrates the efficiency of our bidirectional simulation service in comparison to the use of ACL messages for the agent/environment interaction was shown and discussed.

Beside an in-depth comparison study of our framework to other agent frameworks, it is already planned to present further aspects of our framework in the near future. Here we would like to discuss the load balancing abilities of Agent.GUI for distributed large scale simulations. Furthermore the usage of predefined environments for smart energy networks will be shown soon. Currently, work is in progress to use Agent.GUI for simulations of intelligent and self configuring high pressure gas grids.

## REFERENCES

- [1] S. Arunachalam, Rym Zalila-Wenkstern, and Renee Steiner. Environment mediated multi agent simulation tools. In *SASO Workshops*, pages 43–48. IEEE Computer Society, 2008.
- [2] Fabio L. Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing Multi-Agent Systems with JADE*. Wiley, April 2007.
- [3] Jacques Ferber Fabien Michel and Alexis Drogoul. Multi-agent systems and simulation: A survey from the agent community’s perspective. 2009.
- [4] Daniele Gianni, Andrea D’Ambrogio, and Giuseppe Iazeolla. Dissimjade: a framework for the development of agent-based distributed simulation systems. In Olivier Dalle, Gabriel A. Wainer, L. Felipe Perrone, and Giovanni Stea, editors, *SimuTools*, page 21. ICST, 2009.
- [5] Thomas R. Gruber. A translation approach to portable ontology specifications. *KNOWLEDGE ACQUISITION*, 5:199–220, 1993.
- [6] Alexander Helleboogh, Tom Holvoet, Danny Weyns, and Yolande Berbers. Extending time management support for multi-agent systems. In Paul Davidsson, Brian Logan, and Keiki Takadama, editors, *MABS*, volume 3415 of *Lecture Notes in Computer Science*, pages 37–48. Springer, 2004.
- [7] B. Logan and G. Theodoropoulos. The distributed simulation of multiagent systems. *Proceedings of the IEEE*, 89(2):174 –185, February 2001.
- [8] Charles M. Macal and Michael J. North. Tutorial on agent-based modelling and simulation. *J. Simulation*, 4(3):151–162, 2010.
- [9] Jakob Nielsen. Usability engineering. In Allen B. Tucker, editor, *The Computer Science and Engineering Handbook*, pages 1440–1460. CRC Press, 1997.
- [10] Dirk Pawlaszczyk and Ingo Timm. A hybrid time management approach to agent-based simulation. In Christian Freksa, Michael Kohlhase, and Kerstin Schill, editors, *KI 2006: Advances in Artificial Intelligence*, volume 4314 of *Lecture Notes in Computer Science*, pages 374–388. Springer Berlin / Heidelberg, 2007. 10.1007/978-3-540-69912-5\_28.
- [11] Evangelos Pournaras, Martijn Warnier, and Frances M. T. Brazier. A distributed agent-based approach to stabilization of global resource utilization. In Leonard Barolli, Fatos Xhafa, and Hui-Huang Hsu, editors, *CISIS*, pages 185–192. IEEE Computer Society, 2009.
- [12] Anand S. Rao and Michael P. Georgeff. Bdi agents: From theory to practice. In Victor R. Lesser and Les Gasser, editors, *ICMAS*, pages 312–319. The MIT Press, 1995.
- [13] Reuven Y. Rubinstein and Benjamin Melamed. *Modern Simulation and Modeling*. Wiley & Son, 1998.
- [14] Stuart J. Russell and Peter Norvig. *Artificial Intelligence - A Modern Approach (3. internat. ed.)*. Pearson Education, 2010.
- [15] J. Vázquez-Salceda, W. W. Vasconcelos, J. Padget, F. Dignum, S. Clarke, and M. Palau Roig. Alive: an agent-based framework for dynamic and robust service-oriented applications. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1 - Volume 1*, AAMAS ’10, pages 1637–1638, Richland, SC, 2010. International Foundation for Autonomous Agents and Multiagent Systems.
- [16] Junwei Wu and Xiaojun Cao. Intelligent traffic simulation grid based on the hla and jade. In Wu Zhang, Zhangxin Chen, Craig C. Douglas, and Weiqin Tong, editors, *HPCA (China)*, volume 5938 of *Lecture Notes in Computer Science*, pages 456–464. Springer, 2009.
- [17] Michael Wooldridge and Nicholas R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 1994. Submitted to Revised.
- [18] Michael Wooldridge. *An Introduction to MultiAgent Systems*. Wiley & Sons, 2nd edition, July 2009.

# Minority Game: the Battle of Adaptation, Intelligence, Cooperation and Power

Akihiro Eguchi, Hung Nguyen  
 University of Arkansas, Fayetteville, AR, U.S.A.  
 Email: {aeguchi, hpnguyen}@uark.edu

*Abstract—Minority game is a simulation of a zero-sum game, which has a similar structure to that of a real world market like a currency exchange market. We discuss a way to implement the game and provide a simulation environment with agents that can use various types of strategies to make decisions including genetic algorithms, statistics, and cooperative strategies. The goal of this simulation study is to find the effective strategies for winning the zero-sum game. Results show that both honesty and dishonesty can lead to a player's success depending on the characteristics of the majority of players.*

## I. INTRODUCTION

THE El Farol Bar problem was proposed in 1994 by W. Brian Arthur [1]. This problem involves inductive reasoning using previous histories of other agents to make their decisions. In this problem, there are  $N$  people who independently decide every week to go to a bar or not.

The El Farol Bar problem can also be used in market contexts with agents buying or selling an asset at each step. After each step, the price of the asset is calculated by a simple supply and demand rule: if there are more buyers, the market price will be high, and conversely, if there are more sellers than buyers, the market price will be low. With the price high, the sellers do well, and with the price low, the buyers do well. In either way, the minority group always wins.

A variant of the El Farol Bar problem is the minority game, which was proposed by Challet, Marsili, and Zhang [2]. In the minority game, we have an odd total number of players to always produce majority / minority decisions. If the majority of people stay home, the bar becomes “enjoyable” while if the majority of people go to the bar, it becomes “crowded”. Therefore, the payoff of the game is to declare the agents who take minority action as winners, the majority as losers.

Another important aspect of the minority game is that this is a zero-sum game, in which the decisions of each player influence those of others, and the total sum of the profit and loss of all players becomes zero. In this paper, we assume the minority game to be a simple economic model and discuss the characteristics of the zero-sum game by implementing the minority game [3].

## II. SCORING METHOD

Agents are limited in their computational abilities, and they can only retain the last  $M$  game outcomes in their memory. This memory acts like a shift register with the new bit pushing out the oldest bit. Every agent makes his/her next decision based only on these  $M$  bits of historical data.

Each agent accumulates “capital” reflecting his/her overall score. The agent gets a real point only if the strategy used wins in the next turn. To make the game closer to a real-life stock market, we chose to use our own system of scoring:

If the agent wins a round:  $score = score + nMajority$

If the agent loses a round:  $score = score - nMinority$  with  $nMajority$  and  $nMinority$  denoting the number of players in the majority and minority groups, respectively. Because initially all agents receive 0's as their scores, this scoring system ensures that at any time, the sum of all agents' scores stay the same, or that the total score (capital in the real world) is conserved, but its distribution changes after each turn.

## III. IMPLEMENTATION

Our simulation is an agent-based model [4] with four different types of agents. Normal agents make decisions using genetic algorithms [5]. Given a sequence of the last  $M$  outcomes, there are  $2^M$  possible inputs for an agent's decision making. A strategy specifies what the next action is for every sequence of last  $M$  outcomes, so there is a total number of  $2^{2^M}$  possible strategies. After every turn, a certain number of strategies with poor performances, calculated based on a virtual score the agent could have been with the strategy, is discarded, and new strategies are randomly generated.

Team agent is a Normal agent who belongs to a team to share their memories to make a team decision without accessing previous decisions of others as done in a previous work [6]. The agents with the higher scores have more weight for their votes by the following formula:

$$R = \sum_{i=1}^{nTeamMembers} \frac{s_i - min}{max - min} \times r_i$$

where  $s_i$  is the score of agent  $i$ ; and  $r_i$  is the response of the agent (-1 or 1). If  $R > 0$ , the team's decision is to stay, and vice versa.

Additionally, we assign each Team agent a percentage of loyalty towards their team. We have Team agents with two types of loyalty: the first is to tell a lie about his intention when voting for the team decision (type 1), and the second is to do the opposite of the team decision (type 2).

Super agent is another agent who makes decisions based on a certain number of previous market results, which weigh more on the most recent history. It calculates the probability of the Super agent to go to the bar in the next turn:

$$P = 100 - \left( \sum_{i=1}^{nHist} i^2 \times HistPercentStay[i] \right) / \sum_{i=1}^{nHist} i^2$$

This is a simple way to predict the market, and it is based on the assumption that the number of agents that go and the number that stay will converge in the long run as they learn.

#### IV. DATA STRUCTURE

To conserve space, instead of using Java's boolean to store the histories of agents, we use short instead. Each boolean costs 1 byte, so for an agent with 10 history entries, it costs 10 bytes to save his history, which seems like a small number. However, if each agent has 12 strategies, the total number of bytes used to store all possible histories for a normal agent's strategies is over 200 MB for 1501 agents. Therefore, instead of using 10 boolean values to indicate whether the agent has won or lost the past latest 10 turns, we chose to use one bit to store the outcome of one turn, so in total we only need 1 bit instead of 1 byte to store the history. Because the maximum number of history entries an agent can store is 10, we use one Java's short value to store all the history entries. Thus, instead of 10 bytes, we only need 2 bytes for each agent. In total, we only need 3 KB instead of 15 KB. Now using the same approach for histories stored within each strategy (not within each agent), the total number of bytes we need to allocate for the strategies is just over 55 MB. This is a good upper bound for memory consumption, compared to 200 MB with Java's boolean.

Since each agent's history is a short, we can just store each strategy's responses as a boolean array and use the agent's history as the index for that array. In the end, we are able to reduce the number of bytes allocated for all strategies further to about 18.5 MB for the same configurations.

In addition, instead of using arrays of booleans to store the responses of Normal strategies, we chose arrays of integers, so the boolean value in the previous implementation is now represented by only one bit. Since one boolean value consumes 8 bits (1 byte), this method of representing one boolean value using 1 bit thus reduces the memory consumption 8 times. As a result, we increase the maximum number of Normal agents from 1,600 to 10,000 on our laptops. Therefore, our final design consumes a total of about 2.3 MB, or almost 1/100 of the initial 200 MB.

#### V. SIMULATION

##### A. Normal agents

Suppose there are 1001 Normal agents with the memory size of 6 turns in the simulation, Figure 1 shows the graph produced when it runs for 3650 turns. In the graph on the left side, the red curve represents the best score of all agents in the simulation while the blue curve represents the absolute value of the worst. The bar graph on the right shows the distribution of the scores of each agent. The average score of all agents, which is zero, is denoted as a red vertical line.

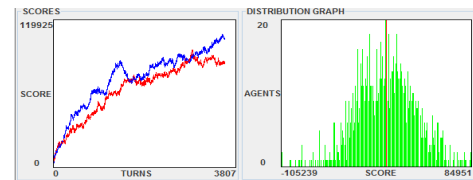


Fig 1. Graph of Normal agents

The distribution graph shows a well-balanced bell curve that shows a normal distribution of the agents' scores. The genetic algorithm, which the Normal agents use, optimizes the winning probability of each agent, so supposedly each agent fine tunes their decision-making strategy and increases their score as time goes on. However, because of the characteristics of the zero-sum game, all agents cannot win at the same time. This configuration shows the characteristics of a zero-sum game very well.

##### B. Normal agents and Super agents

We ran the simulation again with 1001 agents for 3650 turns, but with 501 Normal agents and 500 Super agents who use statistics based on market history rather than their past decisions.

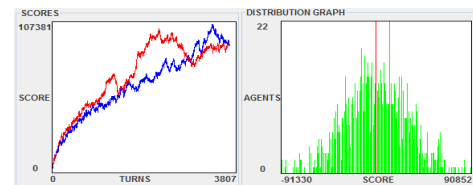


Fig 2. Graph of Normal agents and Super agents

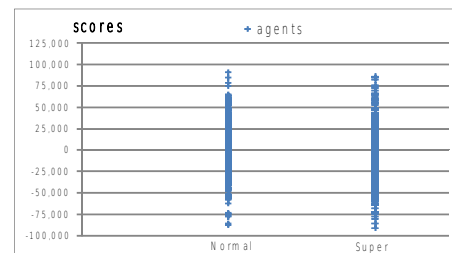


Fig 3. Comparison of Normal and Super scores decomposed

Interestingly, similar to the previous configuration, we observed a well-balanced normal distribution (Figure 2). In addition, the distribution of each type of agent is very similar to each other (Figure 3). This suggests that the bell curve in Figure 2 is the sum of two smaller and almost identical bell curves. Thus, it shows that simple statistics based on a market history can make winning decisions as good as genetic algorithms can.

##### C. Normal agent and loyal Team agents

Another interesting configuration is when Team agents are 100% loyal to their teams. The participants in this configuration consist of 501 Normal agents and 5 teams, each of which consists of 100 members.



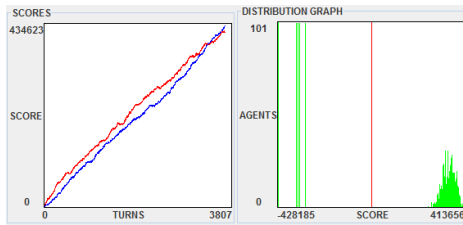


Fig 4. Graph of Normal agents and team agents



Fig 5. Comparison of Normal and Team scores

Team agents perform far more poorly than Normal agents since they all do exactly the same thing. As this is a minority game, this type of Team strategy is the worst of all strategies. Figures 4 and 5 also show how the scores of these two types of agents are highly independent of each other.

*D. All agents in one simulation*

In the next two configurations, we ran the experiments with 501 Normal agents, 5 teams of 100 agents each, and 500 Super agents for 3650 turns. Team agents now have their loyalties randomly generated for both types.

*1) With disloyal Team agents: liars (Type 1)*

Type 1 disloyal Team agents lie about their intentions.

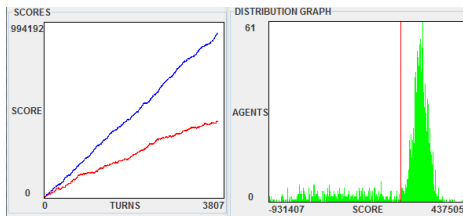


Fig 6. All agents - Team agents use team strategy 1

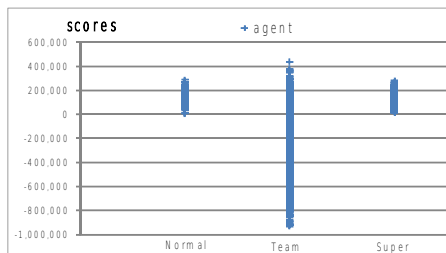


Fig 7. Comparison of agent types

Using Figure 7 and simulation B, we can conclude that the bell curve to the right of the red average line in Figure 6 is the sum of Normal and Super agents. Their averages are much better than those of Team agents (Figure 7). Team agents' scores are scattered, and there is a very strong negative correlation between score and loyalty. About 20% of the

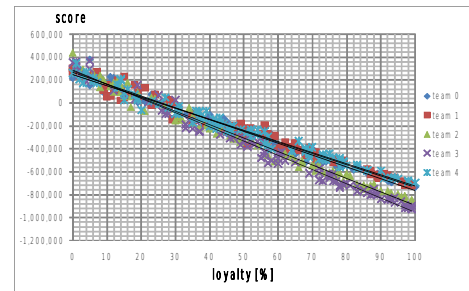


Fig 8. Plotting of Team agents who use team strategy 1

Team agents who are the most disloyal perform above the zero mark.

*2) With disloyal Team agents: perverse fellows (Type 2)*

In this case, type 2 disloyal agents do the opposite of their team's resolution. We predicted that the team decisions affect the majority of the members, so the ones who do not follow the team's agreement are more likely to win.

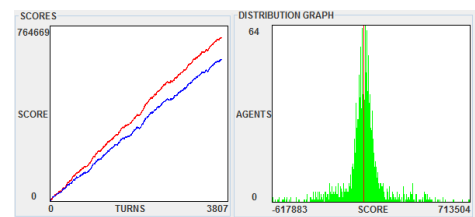


Fig 9. All agents - Team agents use team strategy 2

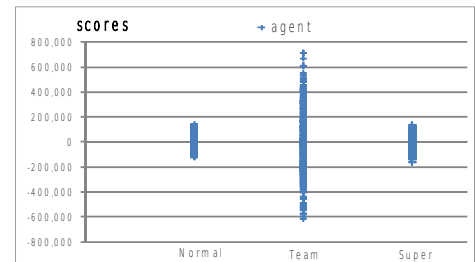


Fig 10. Comparison of agent types

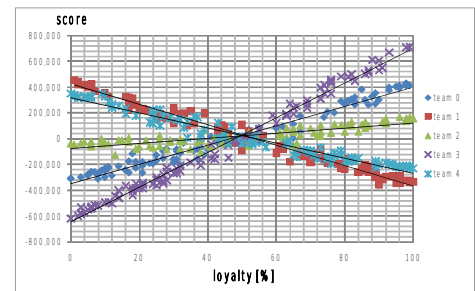


Fig 11. Plotting of Team agents who use team strategy 2

To our surprise, the negative correlation between loyalty and score as in type 1 disloyalty is not true anymore. There is either a positive or negative correspondence between loyalty and score for members of the same team. The ones who betray their team 50% of the time do not lose or gain anything in the game (Figure 11). In contrast to Figure 7, Figure 10 shows how the Team agents, taken as a whole, perform just as well as the Normal and Super agents. However, Team agents' scores are more spread out. When viewing both Fig-



ures 7 and 10, we can recognize how the best Team agents win over the Normal and Super agents, and even make their scores shift down dramatically. This helps “even out” the field, so the score distribution goes back to the bell curve shape (Figure 9).

#### E. The final showdown of Team agents

To see how the two types of Team agents would compete with each other, in this final simulation, we have eight teams, whose loyalties are randomly generated, with 100 agents each: the first four are type 1 and the other four are type 2.

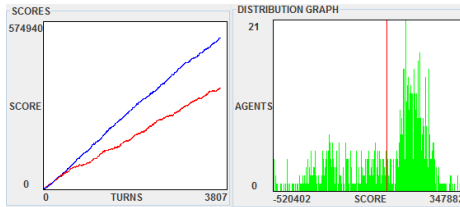


Fig 12. Two types of Team agents

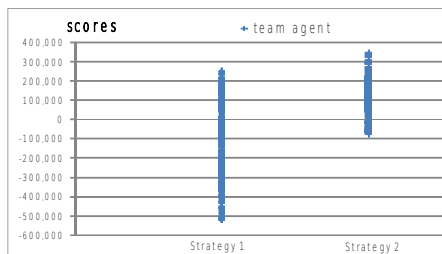


Fig 13. Comparison of team strategies

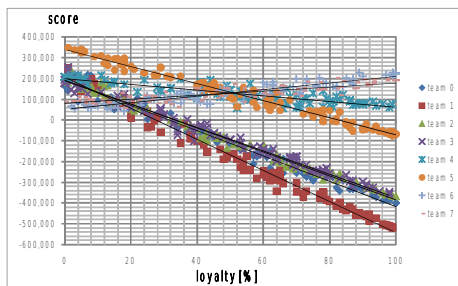


Fig 14. Plotting of two types of Team agents

Type 2 outperforms type 1 both absolutely and on average. Furthermore, the score distribution of teams following type 2 is more balanced (Figure 14), and the difference between the highest and lowest scores is less than those of teams following type 1 (Figure 8). However, because teams with type 2 “steal” from teams following type 1, the total score distribution is skewed to the right.

#### VI. OBSERVATIONS

If we take a look at the wealth distribution in the real world, we can see a similar trend to our simulation. According to a report by Dr. Zhu Xiao Di, in the United States in 2004, the top 25 percent of households owned 87 percent of the wealth in the country whereas the bottom 25 percent of households owned nothing [7]. This is illustrated in Figures 7 and 8, which show the scores of loyal team agents who are “bullied” by both their disloyal teammates and by Normal

and Super agents. From the results, if you are a Normal agent, then you will likely live an average life. However, if you are a Team agent, i.e., more prone to be positively or negatively influenced by other people, your life could be extreme in either way. At first, we assumed that being disloyal to the society that one belongs to would be the only way to win the game, but the simulation showed that this is not always the case. When the honest and loyal are the minority, they win against the tricksters and treacherous majority; and vice versa. Therefore, we successfully showed that similar to the real world, any organization could be extremely successful either by being honest or dishonest depending on its environment.

#### VII. CONCLUSION

In this paper, we discussed a way to implement a variant of the minority game. In order to deal with a large number of participants in the simulation, we introduced several ways to optimize the architecture. Then, we ran the simulator with several strategies to observe the results. Because of the characteristics of the game, the genetic algorithm produced a normal distribution. We also showed how a genetic algorithm is comparable to statistical analysis of the game. Since this is a minority game, team decisions play an important part. If all Team agents are honest, they are likely to perform worse than Normal agents. Then, we showed how the three types of agents would perform together in the same game with disloyal Team agents who lie to their teammates (type 1), or who do the opposite of their team's decision (type 2). The observation is that type 2 helps Team agents as whole: it makes the best Team agents better than the best of Normal and Super agents and narrows the gap between the worst of team agents and the worst of the other two types. There also exist both positive and negative correlations between loyalty and score with type 2 while only negative correspondence is observed with type 1. The last simulation illustrates how type 2 is more effective than type 1. This also shows that to win this game, both loyal and disloyal Team agents can win. The losers are the ones with opposite characters of their own teams.

#### REFERENCES

- [1] W. B. Arthur, “Inductive Reasoning and Bounded Rationality,” in *American Economic Review*, vol. 84, pp. 406 - 411, 1994.
- [2] D. Challet, M. Marsili, and Y.-C. Zhang, *Minority Games*, Illustrated edition. Oxford Univ Press, 2005.
- [3] G. W. Greenwood and R. Tymerski, “A game-theoretical approach for designing market trading strategies,” in *Proc. 2008 IEEE Conf. Computational Intelligence and Games*, pp. 316-322, 2008.
- [4] A. Chakraborti, I. M. Toke, M. Patriarca, and F. Abergel, “Econophysics: Empirical facts and agent-based models,” 2009.
- [5] Marko Sysi-Aho, Anirban Chakraborti, and Kimmo Kaski. “Biology helps you to win a game,” in *Proc. Unconventional Applications of Statistical Physics Conf.*, vol. T106 of *Physica Scripta T-series*, pp. 32-35, 2003.
- [6] T. Kalinowski, H.-J. Schulz, and M. Brieze, “Cooperation in the Minority Game with local information,” *Physica A: Statistical Mechanics and its Applications*, vol. 277, issues 3-4, pp. 502-508, Mar. 2000.
- [7] Z. X. Di, “Growing Wealth, Inequality, and Housing in the United States,” Joint Center for Housing Studies, Harvard Univ., Cambridge, MA, W07-1, 2007.

# Towards a Generic Testing Framework for Agent-Based Simulation Models

Önder Gürçan\*<sup>†</sup>, Oğuz Dikenelli\*

\*Ege University

Computer Engineering Department  
Universite cad. 35100  
Bornova, Izmir, Turkey  
E-mail: name.surname@ege.edu.tr

Carole Bernon<sup>†</sup>

<sup>†</sup>Toulouse III University

Institut de Recherche en Informatique de Toulouse (IRIT)  
118 route de Narbonne  
31062 Toulouse Cedex 9, France  
Email: name.surname@irit.fr

**Abstract**—Agent-based modeling and simulation (ABMS) had an increasing attention during the last decade. However, the weak validation and verification of agent-based simulation models makes ABMS hard to trust. There is no comprehensive tool set for verification and validation of agent-based simulation models which demonstrates that inaccuracies exist and/or which reveals the existing errors in the model. Moreover, on the practical side, there are many ABMS frameworks in use. In this sense, we designed and developed a generic testing framework for agent-based simulation models to conduct validation and verification of models. This paper presents our testing framework in detail and demonstrates its effectiveness by showing its applicability on a realistic agent-based simulation case study.

## I. INTRODUCTION

VERIFICATION and validation of simulation models is one of the main dimensions of simulation research. Model validation deals with building the *right model*, on the other hand, model verification deals with building the *model right* as stated in Balci [1]. Model testing is a general technique which can be conducted to perform validation and/or verification of models. Model testing demonstrates that inaccuracies exist in the model or reveals the existing errors in the model. In model testing, test data or test cases are subjected to the model to see if it functions properly [2].

There are many works about verification and validation of agent-based simulations [3]–[6]. However, these studies do not directly deal with model testing process and there is no proposed model testing framework to conduct validation and verification through the model testing process. Based on this observation, our main motivation is to build a testing framework for agent-based simulation models to facilitate the application of the model testing.

Naturally, one has to define the whole model testing requirements of ABMS to be able to develop a model testing framework. To define these requirements, we first define basic elements of ABMS that can be subject of model testing process. Then, we use a generic model testing process [1] and elaborate on the requirements of the model testing framework when it is used throughout this process. Finally, we categorize ABMS's model testing requirements in micro- and macro-levels by an inspiration from ABMS applications in sociology domain [7]. In this categorization, the micro-level takes basic

elements individually and defines the framework requirements from the basic element's perspective. On the other hand, the macro-level considers a group of basic elements and assumes that such a group has a well defined model that needs to be validated. Hence, the macro-level defines model testing requirements of such groups.

After having defined the requirements of the framework, a conceptual architecture which includes some generic conceptual elements to satisfy these requirements of the framework is proposed. These elements are specified and brought together to conduct the model testing of any ABMS application. Then, a software architecture is introduced which realizes the conceptual elements. This architecture is extensible in a sense that new functionalities based on domain requirements might be easily included. Also, on the practical side, since there are many agent-based simulation frameworks in use [8], the proposed architecture is generic enough to be customized for different frameworks.

This paper is organized as follows. The next section defines the testing requirements for ABMS. Section III then describes the generic agent-based simulation testing framework we propose. A case study that shows the effectiveness of the proposed framework is studied in Section IV. After discussing the proposal in Section V, Sections VI and VII conclude the paper with an insight to some future work.

## II. TESTING REQUIREMENTS FOR AGENT-BASED SIMULATION MODELS

The basic elements of agent-based simulations are agents, the simulated environment and the simulation environment [9]. *Agents* are active entities that try to fulfill their goals by interacting with other agents and/or simulated environments in which they are situated. They behave autonomously depending on their knowledge base. Moreover, during an agent-based simulation, new agents may enter the system or some agents may also disappear.

A *simulated environment* contains non-agent entities of the simulation model and agents of that environment. This environment can also carry some global state variables that affect all the agents situated in it and can have its own

dynamics like the creation of a new agent. In an agent-based simulation model, there must be at least one simulated environment. However, there may also be various simulated environments with various properties depending on the requirements and the complexity of the model. As well as explicitly specified behaviours of these elements (agent and simulated environments), higher level behaviours can emerge from autonomous agent behaviours and model element interactions (agent to agent interactions and agent to simulated environment interactions).

The *simulation environment* (or infrastructure), on the other hand, is an environment for executing agent-based simulation models. Independent from a particular model, it controls the specific simulation time advance and provides message passing facilities or directory services. Unlike the other basic elements, the simulation environment is unique for every simulation model and does not affect the higher level behaviours.

The basic elements are developed and brought together following a development process to produce a simulation model [10]. The overall simulation model is also verified and validated in parallel with the development process. Our aim is to develop a generic testing framework to conduct model testing in agent-based simulations. In general, testing requires the execution of the model under test and evaluating this model based on its observed execution behaviour. Similarly, in the simulation domain this approach is defined as *dynamic validation, verification and testing (VV&T) technique* according to Balci's classification [2]. According to Balci, dynamic VV&T techniques are conducted in three steps. Below, we interpret those three steps in terms of model testing of agent-based simulations to be able to capture the requirements for the intended testing framework:

- 1) *Observation points for the programmed or experimental model are defined (model instrumentation)*. An observation point is a probe to the executable model for the purpose of collecting information about model behaviour. In this sense, a model element is said to be *testable* if it is possible to define observation points on that element. From the perspective of ABMS, *agents* and *simulated environments* might be testable when it is possible to define observation points for them. The *simulation environment*, on the other hand, is not a testable element. However, it can be used to facilitate the testing process.
- 2) *The model is executed*. As stated above, in agent-based simulations, model execution is handled by the simulation environment. During model execution, a model testing framework can use the features of the simulation environment (if any) to collect information through the observation points.
- 3) *The model output(s) obtained from the observation point(s) are evaluated*. Thus, for evaluating the model outputs, a model testing framework should provide the required evaluation mechanisms. Observed outputs are evaluated by using reference data. Reference data could be either empirical (data collected by observing the real

world), a statistical mean of several empirical data, or they can be defined by the developer according to the specification of the model.

Apart from the above requirements, to be able to design a well structured testing framework, we also need to identify the testing requirements of testable elements in detail. As we stated in the first step, the testable elements of agent-based simulation models are *agents* and *simulated environments*. In this sense, both of these elements need to be verified and validated. However, what we also require is a means of verifying and validating hypotheses about how interactions and behaviours of these elements at different abstraction levels are related to each other. As Uhrmacher et al. [11] stated, agent-based simulation models describe systems at two levels of organization: *micro-level* and *macro-level*. In sociology, the distinction between these levels is comparatively well established [7]. The *micro-level* considers the model elements individually and their interactions from their perspectives. However, the *macro-level* considers the model elements as one element, and focuses on the properties of this element resulting from the activities at the *micro-level*.

In the following subsections, depending on the characteristics of agent-based simulations, we derive both the *micro-* and *macro-level* testing requirements of agent-based simulation models.

#### A. Micro-level Testing Requirements

In this level, the testing requirements of the basic elements alone and interactions from their perspective are considered.

In this sense, a *micro-level* test may require the following:

- Testing building blocks of agents like behaviours, knowledge base and so forth and their integration inside agents.
- Testing building blocks of simulated environments like non-agent entities, services and so forth and their integration inside simulated environments.
- Testing the outputs of agents during their lifetime. An output can be a log entry, a message to another agent or to the simulated environment.
- Testing if an agent achieves something (reaching a state or adapting something) in a considerable amount of time or before and/or after the occurrence of some specific events with different initial conditions.
- Testing the interactions between basic elements, communication protocols and semantics.
- Testing the quality properties of agents, such as workload for agents (number of behaviours scheduled at a specific time).

#### B. Macro-level Testing Requirements

In this level, the testing requirements of the elements of agent-based simulations as groups or sub-societies is considered. The aim is to test the expected collective properties as a whole:

- Testing the organization of the agents (how they are situated in a simulation environment or who is interacting with who) during their lifetime.

- Testing if a group of basic elements exhibit the same *macro-level* behaviour with different initial conditions. This macro-level behaviour could be either an *emergent* behaviour or a *non-emergent* behaviour.
- Testing if a group of basic elements is capable of producing some known output data for a given set of input data.
- Testing the timing requirements of *macro-level* behaviours of a group of basic elements.
- Testing the workload for the system as a whole (number of agents, number of behaviours scheduled, number of interactions etc.).

### III. THE GENERIC AGENT-BASED SIMULATION TESTING FRAMEWORK

#### A. The Conceptual Model

To be able to satisfy the testing requirements for agent-based simulation models, developers first need a tool that supports model instrumentation. In other words, the tool should allow defining observation points for each testable element individually and as a group. Moreover, this tool has to support collecting information from these observation points while the model is executed. And apparently, it has to provide evaluation mechanisms for the assessment of the collected information.

In this sense, we designed a generic testing framework that provides special mechanisms for model testing of ABMS. As we mentioned before, testing requires the execution of the model under test. In this context, we call each specific model designed for testing a *Test Scenario*. A *Test Scenario* contains at least one model element under test (depending on the level and the need), one special agent to conduct the testing process (*Test agent*), the other required model elements, the data sources in which these elements make use of and a special simulated environment (*Test environment*) that contains all these elements (see Figure 1). It can also include one or more fake elements to facilitate the testing process. Each *Test Scenario* is defined for specific requirement(s) which includes the required test cases, activities and their sequences, and observation requirements. For executing *Test Scenarios*, we designed another mechanism called *Scenario Executer*. *Scenario Executer* is able to execute each *Test Scenario* with different initial conditions for pre-defined durations.

*Test agent* is responsible for instrumenting the testable elements, collecting information from them and evaluating these information in order to check if they behave as expected. *Test agent* can access every model element during the execution of a *Test Scenario*. However, none of the model elements are aware of it. So it does not affect the way the other elements of the scenario behave. To be able to supply this feature, we designed a special simulated environment called *Test environment*. All the model elements of the scenario, including *Test agent*, are situated in this environment. However, apart from *Test agent*, none of the other elements are aware of *Test environment*.

Another special mechanism introduced is the usage of special elements called *Fake agents* and *Fake environments*

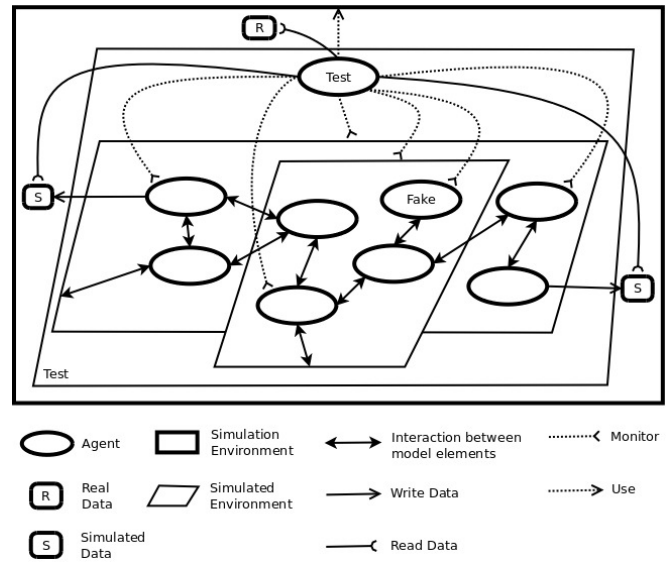


Fig. 1. An illustrative example for a test scenario. As represented in the figure the basic ingredients for test scenarios are: the Test agent, fake agents, the basic elements of agent-based simulation models (agents, simulated environment and simulation environments) and the data they use/produce. The Test agent is able to collect information from all these elements.

to facilitate the testing process. They are especially useful when a real element is impractical or impossible to incorporate into a scenario execution. They allow developers to discover whether the element(s) being tested respond(s) appropriately to the wide variety of states such elements may be in. For example, for a *micro-level* test aiming at testing the interaction protocol of a model element, there is no need to use the real implementation of the other model elements, since the aim is to focus on the interaction protocol. In this sense, *Fake agents* mimic the behaviour of real agents in controlled ways and they simply send pre-arranged messages and return pre-arranged responses. Likewise, *Fake environments* mimic the behaviour of real simulated environments in controlled ways and they are used for testing agents independently from their simulated environments. Although the term “mock” can also be used in testing in multi-agent systems literature [12], we preferred using the term “fake” rather than “mock” for describing the non-real elements, since there is also a distinction between “fake” and “mock” objects in object-oriented programming. Fakes are the simpler of the two, simply implementing the same interface as the object that they represent and returning pre-arranged responses [13]. Thus a fake object merely provides a set of method stubs. Mocks, on the other hand, do a little more: their method implementations contain assertions of their own.

The next subsection explains the internal architecture of our generic testing framework.

#### B. The Internal Architecture

The internal architecture of the generic testing framework is given in Figure 2. This framework is based on the JUnit<sup>1</sup>

<sup>1</sup>JUnit, <http://www.junit.org/>

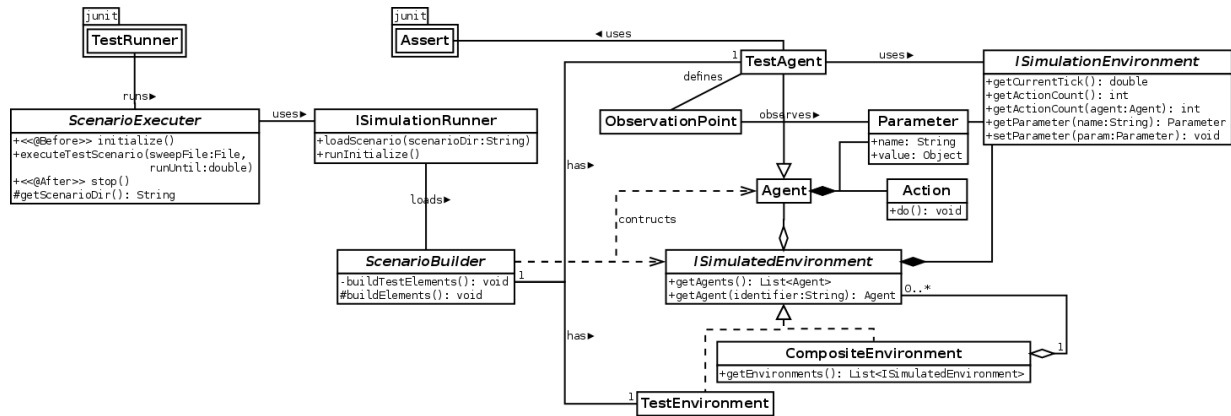


Fig. 2. The conceptual UML model for the generic testing framework.

testing framework, which is a simple framework to write repeatable tests for Java applications. Basically, the test runner of JUnit (`TestRunner`) runs test cases and prints a trace as the tests are executed followed by a summary at the end. Using JUnit infrastructure, we defined our scenario executer (`ScenarioExecutor`) as a test case of JUnit. Consequently, by using the existing mechanisms and graphical user interfaces of JUnit, test scenarios can easily be executed.

When loaded by the JUnit test runner, `ScenarioExecutor` first initializes the given test scenario by using the generic simulation runner interface (`ISimulationRunner`) that builds the scenario by using a builder (`ScenarioBuilder`). `ScenarioExecutor` uses its `getScenarioDir()` method to retrieve the name of the directory in which the required files of the scenario are located. After, the `ScenarioExecutor` executes the test scenario with different parameters by sweeping the provided file until the defined limit for the test scenario. To do so, `ScenarioExecutor` class provides an `executeTestScenario()` method that enables executing the same test scenario with different initial conditions for different predefined durations. The runner of the agent-based simulation framework is responsible for loading `ScenarioBuilder`. `ScenarioBuilder` builds the scenario by constructing required model elements. It builds the `TestEnvironment` and the `TestAgent` internally by using `buildTestElements()` method. Other model test elements (the simulated environments (`ISimulatedEnvironment`) and the agents (`Agent`))<sup>2</sup>, on the other hand, are built externally by using the provided stub method `buildElements()`.

`TestAgent` is able to access all basic elements in order to make model instrumentation. For accessing the simulated environments and the agents, it uses the `TestEnvironment` and for accessing the simulation infrastructure it uses a special interface (`ISimulationEnvironment`) that provides utility

methods to gather information about the ongoing scenario execution. For example, it can get the current value of the simulation clock (`getCurrentTick()`), get the number of actions scheduled at specific time points (`getActionCount()`)<sup>3</sup> and so on. `TestAgent` is responsible for managing the testing process in a temporal manner. Basically, it monitors the agents and the simulated environments through the observation points (`ObservationPoint`) and performs assertions (using `Assert`) depending on the expected behaviour of the agent-based model under test. However, if the ABMS framework provides predefined features for defining observation points, it is not necessary to use the `ObservationPoint` concept in the concrete model testing framework. Since `TestAgent` itself is also an agent, all these aforementioned mechanisms can be defined as agent actions (`Action`) that can be executed at specific time points during the testing process. It can monitor and keep track of the states of all the elements of the test scenario, or the messages exchanged between them during the scenario execution. As a result, `TestAgent` is able to test the model at specific time points by using instant or collected data, and when there is a specific change in the model (when an event occurs). If all the assertions pass until the specified time limit for the test, the test is said to be *successful*, otherwise the test is said to be *failed*.

Fake agents can be defined by using the same interface (`Agent`) as the real agents they mimic, allowing a real agent to remain unaware of whether it is interacting with a real agent or a fake agent. Similarly, fake environments can also be defined by using the same interface (`ISimulatedEnvironment`) as the real interfaces they mimic.

### C. Implementation

The generic testing framework has been successfully implemented for Repast (Figure 3). Repast is an agent-based simulation framework written in Java [14]. It provides predefined classes for building agent-based simulation models

<sup>2</sup>We do not address implementation issues on how to apply these concepts in practice, as this is highly dependent upon the simulation framework used and the objective of the simulation study.

<sup>3</sup>Since many agent-based simulators use a global scheduler, such information can be retrieved from the scheduler of the simulation infrastructure.

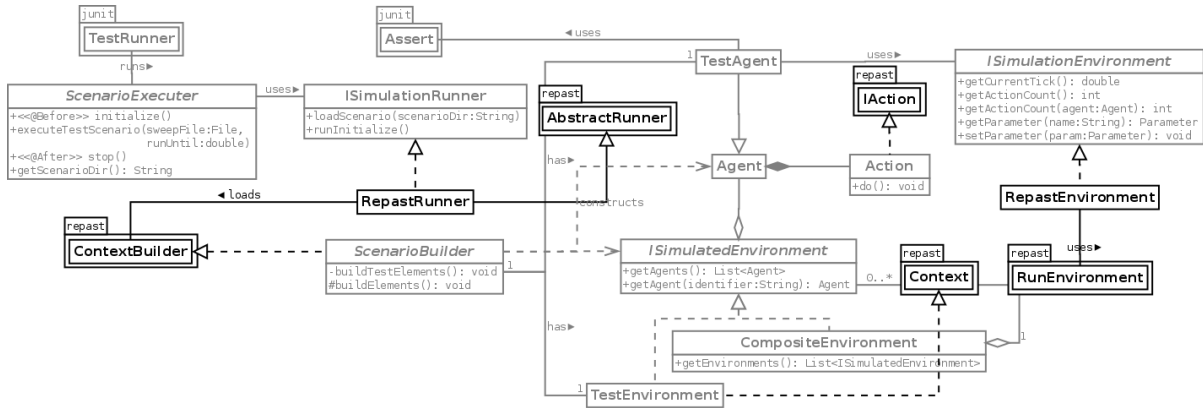


Fig. 3. The UML class model for the Repast implementation of the generic testing framework. The ObservationPoint concept is removed here, since Repast provides a special mechanism for defining observation points.

as well as for accessing the Repast simulation infrastructure during run time. For the implementation, first a simulation runner (RepastRunner) is defined by extending the AbstractRunner class provided by Repast. Since Repast uses the ContextBuilder interface for building simulations, our ScenarioBuilder implements this interface. Then, a class for representing the Repast simulation infrastructure (RepastEnvironment) is defined. This class uses the methods provided by RunEnvironment class of Repast for accessing the Repast simulation infrastructure as defined in ISimulationEnvironment interface. And after, TestEnvironment is realized by implementing Context interface provided by Repast, since it is the core concept and object in Repast that provides a data structure to organize model elements. Finally, the actions of agents are implemented as a subclass of IAction provided by Repast.

D. Usage

The developer first needs to extend ScenarioBuilder to define the elements of the test scenario and the initial parameters. Then the TestAgent needs to be designed together with its monitoring and testing actions for the testing process. Finally, ScenarioExecuter should be extended for defining the different initial conditions and time limits for each scenario execution.

IV. CASE STUDY: TONIC FIRING OF A MOTONEURON

To demonstrate the effectiveness of our testing framework, we show its applicability on a micro-level testing example<sup>4</sup>. For the case study, we have chosen a test scenario from one of our ongoing projects. In this project, we are developing a self-organized agent-based simulation model for exploration of synaptic connectivity of human nervous system [15].

The nervous system is a network of specialized cells that communicate information about organism’s surroundings and itself. It is composed of neurons and other specialized cells that aid in the function of the neurons. A neuron is an excitable cell

<sup>4</sup>We are unable to give a macro-level testing example because of space limitations.

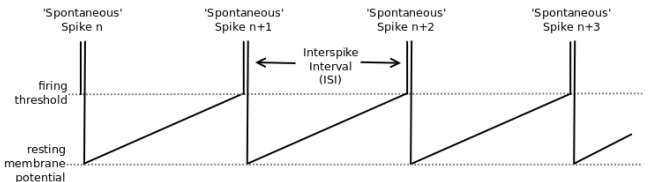


Fig. 4. Tonic firing of a neuron. During tonic firing, a neuron’s membrane potential continuously rises to the firing threshold and makes the neuron fire spontaneous spikes. The time interval between consecutive spikes are called inter-spike intervals (ISI).

in the nervous system that processes and transmits information by electrochemical signalling. Neurons emit spikes when their membrane potential crosses a certain threshold (firing threshold). After emitting the spike, the neuron membrane potential is reset to a certain lower value (resting membrane potential). According to their activation, neurons are of two types: (1) if a neuron is a resting one, it emits a spike when the total synaptic input is sufficient to exceed the firing threshold, or (2) if a neuron is a firing one (e.g., motoneurons, propriospinal neurons), it emits a spike when the membrane potential constantly rises to the firing threshold (Figure 4). A spike is delivered to the other neurons through its axons. When a spike transmitted by a pre-synaptic neuron through one of its axons reaches a synapse, it transmits the spike to the post-synaptic neuron after a certain amount of time (depending on the length of the axon) which is called an axonal delay. To study synaptic connectivity in human subjects, it has been customary to use stimulus evoked changes in the activity of one or more motor units<sup>5</sup> in response to stimulation of a set of peripheral afferents or cortico-spinal fibers. Besides, the ability to record motor activity in human subjects has provided a wealth of information about the neural control of motoneurons [16]. Thus, in our project we are focused on simulation of motor units. We developed and brought together the basic elements of our agent-based simulation model. Meanwhile, to

<sup>5</sup>Motor units are composed of one or more motoneurons and all of the muscle fibers they innervate.

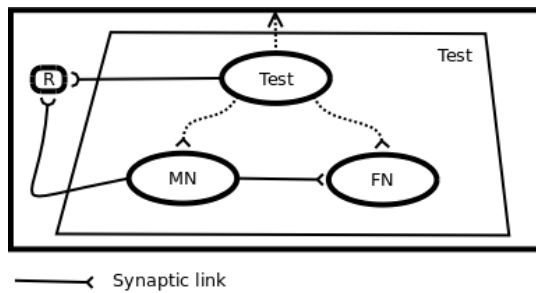


Fig. 5. An illustrative diagram for the “tonic firing of a motoneuron” case study.

verify and validate the model, we designed various test cases both for micro- and macro-levels.

In order to demonstrate how the testing framework can be used, one of our test scenarios was chosen. In this scenario, one *micro-level* behaviour of motoneurons is considered: constant emission of spikes (since they are tonically active). Figure 5 is an illustrative diagram for the selected test scenario. The basic element under test is *Motoneuron* agent (MN). For tonic firing, MN uses the realistic data recorded from a single motor unit of a human subject (R) in Ege University labs<sup>6</sup>. In order to be able to test this *micro-level* behaviour, MN is connected to a *FakeNeuron* agent (FN) with a synaptic link. FN imitates a resting neuron and it is just responsible for receiving the incoming spikes. The synaptic link is responsible for conducting a given spike to FN after a certain amount of time (axonal delay). During the scenario execution, MN will emit spontaneous spikes constantly. These spikes will be delivered to FN through the synaptic link. Each time FN receives a spike, its membrane potential will rise a little for a while and then go back to resting membrane potential. In order to test tonic firing behaviour of MN, *Test agent* observes the activity of both MN and FN for the given amount of time (for each scenario execution, this amount may differ). At the end of this time, it conducts tests using the information it collected during the scenario execution.

For implementing this scenario, first a test builder (*TonicFiringScenario*) needs to be created by extending *ScenarioBuilder* class (Algorithm 1). Within this class, the construction of the basic elements of the test scenario (*Motoneuron* agent and *FakeNeuron* agent) is done. Then, *Test agent* (*TonicFiringTester*) is implemented together with its behaviours by extending *TestAgent* class for the testing process (Algorithm 2 and Algorithm 3). As shown in Algorithm 2, *TonicFiringTester* monitors the activities of *Motoneuron* and *FakeNeuron* agents by observing their membrane potentials. For defining the observation points, the watch mechanism provided by the Repast infrastructure is used (by using `@Watch` annotation). The resting membrane potential is -55 and the firing threshold is -45 in our simulation model. Thus, when the membrane potential of *Motoneuron* becomes more than -45, *Test agent* records the time of occur-

Algorithm 1 Source code for *TonicFiringScenario* class.

```
package motorunit.test03;
import motorunit.*;
public class TonicFiringScenario
    extends ScenarioBuilder{
    private RunningNeuron motorNeuron;
    private FakeNeuron fakeNeuron;

    public static String EXP_MOTOR_ACTIVITY =
        "../data/real_motoneuron_activity.txt";

    @Override
    protected void createAgents() {
        motorNeuron = new RunningNeuron("Motoneuron",
            EXP_MOTOR_ACTIVITY);
        fakeNeuron = new FakeNeuron("FakeNeuron");
        double axonalDelay = 10.0;
        motorNeuron.makeSynapseWith(fakeNeuron,
            axonalDelay);
    }
}
```

Algorithm 2 Source code for *TonicFiringTester* class. Summarized for the better representation of the model instrumentation.

```
package motorunit.test03;
...
import java.util.*;
import repast.simphony.*;
...
public class TonicFiringTester extends TestAgent {
    ...
    private Vector<Double> mnSpikes;
    private Vector<Double> fnActivities;

    public TonicFiringTester() {
        ...
        mnSpikes = new Vector<Double>();
        fnActivities = new Vector<Double>();
    }

    @Watch(watcheeClassName="motorunit.RunningNeuron",
        watcheeFieldNames = "membranePotential",
        query = "colocated",
        triggerCondition = "$watchee.getPotential()>=-45",
        whenToTrigger = WatcherTriggerSchedule.IMMEDIATE)
    public void monitorMotoneuronActivity() {
        double tick=RepastEnvironment.getCurrentTick();
        mnSpikes.add(tick);
    }

    @Watch(watcheeClassName="motorunit.FakeNeuron",
        watcheeFieldNames = "membranePotential",
        query = "colocated",
        triggerCondition = "$watchee.getPotential()>=-55",
        whenToTrigger = WatcherTriggerSchedule.IMMEDIATE)
    public void monitorFakeNeuronActivity() {
        double tick=RepastEnvironment.getCurrentTick();
        fnActivities.add(tick);
    }
    ...
}
```

rence in a list to keep track of the activities of *Motoneuron* during the simulation. Likewise, when the membrane potential of *FakeNeuron* becomes more than -55, *Test agent* records the time of occurrence to keep track of the activity of *FakeNeuron*.

<sup>6</sup>Ege University Center for Brain Research, <http://www.eubam.ege.edu.tr/>.



**Algorithm 3** Source code for TonicFiringTester class. Summarized for the better representation of the test cases.

```

package motorunit.test03;
import static org.junit.Assert.*;
import java.util.*;
import repast.simphony.*;
import motorunit.*;
import umontreal.iro.lecuyer.randvar.
        RandomVariateGen;
public class TonicFiringTester extends TestAgent {
    private RunningNeuron motorNeuron;
    private FakeNeuron fakeNeuron;
    ...
    public TonicFiringTester() {
        motorNeuron = getTestEnvironment().
            getAgent("Motoneuron");
        fakeNeuron = getTestEnvironment().
            getAgent("FakeNeuron");
        ...
    }
    ...
    @ScheduledMethod(start = ScheduleParameters.END)
    public void testTonicFiringOfMotorNeuron() {
        // motoneuron has to generate some spikes.
        assertTrue(motorNeuronSpikes.size() > 0);

        Vector<Double> isiV = ISIDistribution.
            getInstance().calculateISI(motorNeuronSpikes);
        RandomVariateGen rvgSim =
            RandomVariateGenFactory.getGenerator(isiV);
        assertNotNull(rvgSim);
        RandomVariateGen rvgExp =
            motorNeuron.getRandomGen();
        assertNotNull(rvgExp);
        // distribution should be the same
        assertEquals(rvgExp.getClass().getName(),
            rvgSim.getClass().getName());

        Distribution dExp = rvgExp.getDistribution();
        Distribution dSim = rvgSim.getDistribution();
        // alpha parameters should be close
        double alphaExp = dExp.getParams()[0];
        double alphaSim = dSim.getParams()[0];
        assertEquals(alphaExp, alphaSim, 0.1);
        // gamma parameters should be close
        double gammaExp = dExp.getParams()[1];
        double gammaSim = dSim.getParams()[1];
        assertEquals(gammaExp, gammaSim, 0.1);
    }
    @ScheduledMethod(start = ScheduleParameters.END)
    public void testConductionOfSpikes() {
        for (int i = 0; i < fnActivities.size(); i++) {
            double axonalDelay = fnActivities.get(i)
                - mnSpikes.get(i);
            assertEquals(10.0, axonalDelay, 1.0);
        }
    }
}

```

As shown in Algorithm 3, TonicFiringTester executes two actions for testing the *micro-level* behaviour of Motoneuron at the end of each scenario execution (ScheduleParameters.END)<sup>7</sup>. For defining the test cases, the schedule mechanism provided by Repast infrastruc-

<sup>7</sup>The time for the end of the scenario execution may change at each execution, according to the values given by the developer in ScenarioExecutor. See Algorithm 4.

**Algorithm 4** Source code for TonicFiringExecuter class.

```

package motorunit.test03;
import org.junit.Test;
public class TonicFiringExecuter
        extends ScenarioExecuter {
    @Test
    public void tonicFiringTestScenario()
        throws Exception {
        executeTestScenario(null, 2000001);
        executeTestScenario(null, 4000001);
    }
}

```

ture is used (by using @ScheduleMethod annotation). One of the test cases (testTonicFiringOfMotorNeuron()) checks whether the generated spikes of Motoneuron agent have similar characteristics with the real data or not (testTonicFiringOfMotorNeuron()). This test case first tests if Motoneuron agent generated some spikes. And after, it tests if the running (simulated) data generated by Motoneuron agent have similar statistical characteristics: they should represent the same statistical distribution whose parameters are nearly the same. The second test case (testConductionOfSpikes()) is designed to test if the spikes generated by Motoneuron agent are received by FakeNeuron agent properly. To do so, it examines if all the consecutive activities of Motoneuron agent and FakeNeuron agent have the same time difference since there is a 10.0 ms axonal delay.

Finally, in order to execute the test scenario (with various criteria) the base class that the JUnit runner will use (TonicFiringExecuter) is implemented by extending the ScenarioExecuter class (Algorithm 4).

## V. RELATED WORK

Although there is a considerable amount of work about testing in multi-agent systems in the literature (for a review see [17]), there are not much work on testing in ABMS. Niazi et al. [5] present a technique which allows for flexible validation of agent-based simulation models. They use an overlay on the top of agent-based simulation models that contains various types of agents that report the generation of any extraordinary values or violations of invariants and/or reports the activities of agents during simulation. In the sense of using special agents in which the agents under test are not aware of, their approach is similar to ours. But unlike our approach, they define various types of special agents. However, we use a single agent for testing, since at every test our aim is to test one single use case of the system [18]. Besides, they define an architecture but since they begin without defining the requirements it is not quite possible to understand what they are testing. Pengfei et al. [6] proposes validation of agent-based simulation through human computation as a means of collecting large amounts of contextual behavioural data. De Wolf et al. [19] propose an empirical analysis approach combining agent-based simulations and numerical algorithms

for analyzing the global behaviour of a self-organising system.

None of the above approaches is well structured and their authors do not give internal details. These approaches also do not take into account neither simulated nor simulation environments. However, we think that both environments need to be involved in the model testing process since they are two of the main ingredients of agent-based simulation models.

## VI. CONCLUSION & FUTURE WORK

This body of work presents the initial design of a novel generic framework for the automated model testing of agent-based simulation models. The basic elements for testing are identified as agents and simulated environments. For testing each use case for these elements, a test scenario needs to be designed. In our active testing approach, for each test scenario, there is a special agent that observes the model elements under test, and executes tests that check whether these elements perform the desired behaviours or not. The framework also provides generic interfaces for accessing both the simulation environment and the simulated environments. However, these interfaces are not mature and need to be improved in order to be able to design more comprehensive test scenarios.

To demonstrate the framework's applicability, it is implemented for a well-known agent-based simulation framework called Repast. For model instrumentation, the "watch" mechanism provided by Repast is used. However, if an agent-based simulation framework does not provide such a mechanism, the developers may need to implement it. To solve this design problem, the `Observer` design pattern [20] can be used. In this sense, to show the suitability of the proposed generic framework in case of adoption of frameworks different from Repast, we are planning to implement it for other frameworks.

Since testing is meaningful when it is involved in a development methodology, as another future work, we are planning to define a test-driven process based on these testable elements and the generic framework defined in this paper. Moreover, we are also planning to show how our generic testing tool can be used for testing self-organising multi-agent systems. The metrics for self-organization and emergence mechanisms for achieving self-\* properties are given in recent works [21] and [22]. We believe that the capabilities of our framework will be able to test and validate all the metrics given in these studies.

## ACKNOWLEDGMENT

The authors would like to thank Kemal S. Türker and Ş. Utku Yavuz from Ege University Center for Brain Research for supplying scientific data about the activity of motoneurons. The work described here was partially supported by Ege University under the BAP 10-MUH-004 project. Önder Gürcan is supported by the Turkish Scientific and Technical Research Council (TUBITAK) through a domestic PhD scholarship program (BAYG-2211) and by French Government through the co-tutelle scholarship program.

## REFERENCES

- [1] O. Balci, "Validation, verification, and testing techniques throughout the life cycle of a simulation study," in *Proc. of the 26th Conf. on Winter simulation*, ser. WSC'94. San Diego, CA, USA: Society for Computer Simulation International, 1994, pp. 215–220.
- [2] O. Balci, "Principles and techniques of simulation validation, verification, and testing," in *Proc. of the 27th Conf. on Winter simulation*, ser. WSC'95. Arlington, VA, USA: IEEE Comp. Soc., 1995, pp. 147–154.
- [3] T. Terano, "Exploring the vast parameter space of multi-agent based simulation," 2007, pp. 1–14.
- [4] F. Klügl, "A validation methodology for agent-based simulations," in *Proceedings of the 2008 ACM symposium on Applied computing*, ser. SAC '08. New York, NY, USA: ACM, 2008, pp. 39–43.
- [5] M. A. Niazi, A. Hussain, and M. Kolberg, "Verification and Validation of Agent-Based Simulation using the VOMAS approach," in *MAS&S @ MALLOW'09, Turin*, vol. 494. CEUR Workshop Proceedings, September 2009, p. (on line).
- [6] X. Pengfei, M. Lees, H. Nan, and V. V. T., "Validation of agent-based simulation through human computation : An example of crowd simulation," in *Multi-Agent-Based Simulation XI*, 2011, pp. 1–13.
- [7] K. G. Troitzsch, "Multilevel simulation," in *Social Science Microsimulation*, 1995, pp. 107–122.
- [8] C. Nikolai and G. Madey, "Tools of the trade: A survey of various agent based modeling platforms," *Journal of Artificial Societies and Social Simulation*, vol. 12, no. 2, 2009.
- [9] F. Klügl, M. Fehler, and R. Herrler, "About the role of the environment in multi-agent simulations," in *Environments for Multi-Agent Systems*, ser. LNCS, D. Weyns, H. Van Dyke Parunak, and F. Michel, Eds. Springer Berlin / Heidelberg, 2005, vol. 3374, pp. 127–149.
- [10] F. Klügl, "Multiagent simulation model design strategies," in *MAS&S @ MALLOW'09, Turin*, vol. 494. CEUR Workshop Proceedings, September 2009, p. (on line).
- [11] A. Uhrmacher and W. Swartout, *Agent-oriented simulation*. Norwell, MA, USA: Kluwer Academic Publishers, 2003, pp. 215–239.
- [12] R. Coelho, U. Kulesza, A. von Staa, and C. Lucena, "Unit testing in multi-agent systems using mock agents and aspects," in *Proc. of the 2006 Int. Workshop on Software eng. for large-scale multi-agent systems*, ser. SELMAS'06. New York, NY, USA: ACM, 2006, pp. 83–90.
- [13] M. Feathers, *Working effectively with legacy code*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2004.
- [14] M. North, N. Collier, and J. Vos, "Experiences creating three implementations of the Repast agent modeling toolkit," *ACM Trans. Model. Comput. Simul.*, vol. 16, no. 1, pp. 1–25, January 2006.
- [15] O. Gürcan, O. Dikenelli, and K. S. Türker, "Agent-based exploration of wiring of biological neural networks: Position paper," in *20th European Meeting on Cybernetics and Systems Research (EMCSR 2010)*, R. Trumph, Ed., Vienna, Austria, EU, 2010, pp. 509–514.
- [16] K. Türker and T. Miles, "Threshold depolarization measurements in resting human motoneurons," *Journal of Neuroscience Methods*, vol. 39, no. 1, pp. 103 – 107, 1991.
- [17] N. Nguyen, A. Perini, C. Bernon, J. Pavon, and J. Thangarajah, "Testing in multi-agent systems," in *Agent-Oriented Software Engineering X*, ser. Lecture Notes in Computer Science, M.-P. Gleizes and J. Gomez-Sanz, Eds. Springer Berlin / Heidelberg, 2011, vol. 6038, pp. 180–190.
- [18] K. Beck, *Test-driven development : by example*. Boston: Addison-Wesley, 2003.
- [19] T. D. Wolf, T. Holvoet, and G. Samaey, "Engineering self-organising emergent systems with simulation-based scientific analysis," in *In: Proceedings of the Fourth International Workshop on Engineering Self-Organising Applications, Universiteit Utrecht*, 2005, pp. 146–160.
- [20] C. Larman, *Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development (3rd edition)*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2004.
- [21] E. Kaddoum, M.-P. Gleizes, J.-P. Georgé, and G. Picard, "Characterizing and Evaluating Problem Solving Self-\* Systems (regular paper)," in *The First Inter. Conf. on Adaptive and Self-adaptive Systems and Applications (ADAPTIVE 2009)*, Athens, Greece, 15/11/2009-20/11/2009. CPS Production - IEEE Computer Society, 2009, p. (electronic medium).
- [22] C. Raibulet and L. Masciadri, "Towards evaluation mechanisms for runtime adaptivity: From case studies to metrics," in *Proceedings of the 2009 Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns*, ser. COMPUTATIONWORLD '09. Washington, DC, USA: IEEE Comp. Soc., 2009, pp. 146–152.

# Modeling Agent Behavior Through Online Evolutionary and Reinforcement Learning

Robert Junges and Franziska Klügl  
Modeling and Simulation Research Center  
Örebro University, Sweden  
Email: {robert.junges,franziska.klugl}@oru.se

**Abstract**—The process of creation and validation of an agent-based simulation model requires the modeler to undergo a number of prototyping, testing, analyzing and re-designing rounds. The aim is to specify and calibrate the proper low-level agent behavior that truly produces the intended macro-level phenomena. We assume that this development can be supported by agent learning techniques, specially by generating inspiration about behaviors as starting points for the modeler. In this contribution we address this learning-driven modeling task and compare two methods that are producing decision trees: reinforcement learning with a post-processing step for generalization and Genetic Programming.

## I. MOTIVATION

AGENT-BASED simulation as an innovative paradigm is particularly apt for the modeling and analysis of complex systems. Based on (mostly) local, low-level interactions, the agents together produce some higher level phenomenon. This bottom-up approach (see [1] coining the notion of social science from the bottom-up) supports the understanding of why and when a phenomenon emerges. It goes beyond only describing macro-level behavior or pattern and requires a highly expertise-based development process for the model. It is basically exploratory as – specially with emergent phenomena – the explicit link between the agent and the system level is missing. Thus, the success of a modeling process is highly depending on the experience of the modeler about what low level behavior might generate the desired macro-level phenomenon.

However, if we want to make agent-based simulation accessible to more people – specially to people without experience in modeling and simulating complex systems –, new ways of systematically developing agent-based simulation models have to be tackled. Our idea is to use adaptive agent architectures for enabling the modeler to develop the model on a higher abstraction level, assuming that this approach will make modeling easier. That means the modeler focuses on the characterization of the phenomenon he/she is interested in, and based on given functionality of effectors and sensors, the behavior model of the agents is developed in a self-adaptive way. Finally, we hope to establish a learning-driven analysis and design approach using self-adaptive agents.

Our main objective for this contribution is to explore the suitability of different learning techniques for a particular modeling problem. Thus our idea is not to evolve or learn a perfect behavior control, but a behavior model for which the source

code can be understood by a human modeler. Decision trees form an obvious behavior representation candidate for this task: they are an intuitive representation for decision-making processes and there are a number of learning techniques that can operate on them. However, supervised decision tree learners such as C4.5[2] or other classification techniques[3] have requirements for a sufficient number of appropriate cases – in our case good situation-action assignments – that cannot be applied directly.

In the following, we first give a short survey on related work concerning modeling for simulated agents and learning technology. Then we will shortly describe the two techniques that we want to analyze here: a combination of Q-Learning with a Decision Tree learning for generalization, and Genetic Programming. After a short introduction of our simple test scenario, we will provide a set of experiments and results. The paper ends with a conclusion and next steps.

## II. SELF-ADAPTIVE AGENTS MODELING

We propose a learning-driven analysis and design approach for using self-adaptive agents in the behavior modeling task. This approach is based on the following core idea: an appropriate conceptual model of the overall system can be developed by setting up a simulation model of the environment allowing to evaluate agent performance and integrating agents that may learn their decision making behavior.

The design strategy starts with the definition of an environmental model together with a function that evaluates agent performance in this environment. After that, the modeler determines what an individual agent shall be able to perceive and to manipulate. In the next step, the designer selects an appropriate combination of agent learning procedures, used by the agents to determine a behavior program that generates the intended overall outcome in the given environment. At the end of this process, ideally a decision tree representing the agent behavior is available in a way that it fits the environmental model and the reward given and thus produces the aggregate behavior intended.

The basic assumption is that the learned decision tree then is sufficiently elaborated that it can serve as a starting point for further steps in a development methodology, such as technical design or implementation. Thus, we transfer the initial agent behavior design from the human modeler to a simulation system. This strategy could be also described as a variant

of an environment-driven strategy for developing multiagent simulation models[4].

Specially in complex systems, a higher number of degrees of freedom have to be handled. This could make a manual modeling process cumbersome, particularly when knowledge about the requirements for the overall system or experience for bridging the micro-macro gap are missing. We assume that using agents that learn at least parts or initial versions of their behavior is a good idea for supporting the modeler in finding an appropriate low level behavior model.

### III. MODELING AND LEARNING

Adaptive agents and multiagent learning have been one of the major focuses within Distributed Artificial Intelligence since its very beginning [5]. The following paragraphs shall give a few general pointers and then a short glance on directly related work on agent learning for behavior design, not for optimizing. It is important to keep in mind that the objective of our work is not addressing mere learning performance but suitability for the usage in a modeling and analysis support context.

Reinforcement learning [6] and evolutionary computing [7] are recurrent examples for categories of learning techniques applied in multiagent scenarios.

A reinforcement learning approach for automatically programming a behavior-based robot is described in [8]. Using Q-Learning algorithm, new behaviors are learned by trial and error, based on a performance feedback function as reinforcement. In [9], also using reinforcement learning, agents share their experiences and most frequently simulated behaviors are adopted as a group behavior strategy. The authors conclude that both learning techniques are able to learn the individual behaviors, sometimes outperforming a hand coded program, and behavior-based architectures speed up reinforcement learning. However, these approaches are for learning “controllers”. The actual behavior program is secondary as long as the given tasks are fulfilled.

Evolutionary Computing (EC) has also been applied for behavior generation in multiagent systems. In [10], Genetic Programming (GP) is used to evolve agent behavior in a Predator and Prey scenario. Agents derive their decision-making process from a decision tree model, that is built throughout the execution. Agent performance is the focus.

Additionally, [11] and [12] discuss the performance of evolutionary generated behavior in multiagent systems. The former presents optimization problems solved by single agent and multiagent approaches. The latter presents a robotic soccer scenario and the concept of layered learning, as a form of problem decomposition.

In [13], Denzinger and Kordt propose a technique for generating cooperative agent behavior using evolutionary online learning. An experiment is developed for this multiagent scenario applied to a pursuit game, where agents are guided by situation-action pairs, or SApairs. The authors compare the proposed online approach with an offline approach for the same problem. In addition to their own SApairs, the agents

have in their memory SApairs that model the other agents’ behaviors. These pairs can be added by observing other agents or by communicating from one agent to another. The results show that incorporating this online learning phase improves the agents’ performance in more complex variants of the scenario and when randomness is introduced, when compared to the offline approach. In the proposed online learning strategy the agents are not required to learn a complete strategy, but only how to perform well in the next steps, after the learning.

Some differences can be pointed out between the work of this paper and the work presented in the last paragraph: in their case, although the results of the learning phase are used in an online way, the learning itself is offline, it does not take place in the “real” scenario where all agent are adapting at the same time; they use Genetic Algorithms (GA), where the individuals are composed of a number of SApairs and genetic operators act on SApairs – not changing its content, but exchanging them among individuals. This way, the GA only operates on creating new programs (individuals with their rules) and not on creating new rules, as it happens in our GP approach, presented in section IV-B; they focus on learning parts of the problem as the execution evolves, and not to learn a complete model of the agent for the problem, as in our case; they assume the agents have knowledge about all the scenario, which can be unrealistic in certain scenarios. Since their focus is on cooperative behavior, a model of other agents is crucial for the success of the learning.

To extend the work presented in [13] and improve the performance of the GA: in [14], the authors include a mechanism to collect data about the usage of the SApairs – such as number of times used and how it changed the fitness – and use this information for applying the genetic operators during the evolutionary learn phase; in [15], the authors present several case studies encoding application specific features into the fitness function. The conclusion is that there is not much difference in terms of performance by refining knowledge already available in the fitness function, however adding new application knowledge improves the performance significantly; in [16], the authors address the problem of modeling other agents’ behavior. The authors point that a model of other agent generated out of few observations often results in inaccurate predictions, while a model comprised of many observations decreases the efficiency of the modeling process. To address the first issue it is proposed to use a method with stereotypes, where the agent, based on current observations can classify the other agent in one of these stereotypes – which in their turn are composed of a set of SApairs – and therefore have a model of the other agent behavior. The second problem is addressed by building trees that branch at each level according to a different feature. The idea is to create a tree-like compact representation, reducing the model to only the relevant observations. The conclusions indicate that when a correct matching of stereotypes is made, there is a significantly increase of performance, and the compactness of representation through trees is a promising approach, but deserves more analysis, specially to minimize the risk of building a model that ignores important observations.

Although there is a wealth of publications dealing with the performance of particular learning techniques, there are not many works focusing on the resulting behavioral model aiming at understandability by a human interpreter. In our learning-driven design approach we transfer the initial agent behavior design from the human modeler to a simulation system. We assume that using agents that learn at least parts or initial versions of their behavior is a good idea for supporting the modeler in finding an appropriate agent-level behavior model.

Nevertheless, the basic question on a way to such a learning-driven analysis methodology is about the availability of appropriate learning techniques, for this form of application, for a particular domain, or maybe just for a particular system. In a previous work we evaluated the applicability of reinforcement learning techniques for this purpose [17]. One of the main problems is the interpretability of the resulting behavior program of the agents. To overcome this problem, in [18] we proposed to use a decision tree learner for post-processing the situation-action pairs with the highest fitness values. However, we reported issues with convergence in RL affecting the generated decision tree, when evaluating its size and quality. In the present work we start our investigation with Genetic Programming. The aim is to overcome the search-space exploration problem – present in reinforcement learning techniques – and include compactness and generalization in the behavior representation by directly working with a decision tree model.

#### IV. APPROACHING AGENT-LEARNING

In this section we describe the learning techniques chosen for evaluation in this contribution. Their implementation, as well as the experiments conducted – presented in section V – used the multiagent simulation tool SeSAM ([www.simsesam.org](http://www.simsesam.org)).

##### A. Q-Learning plus C4.5: RL+

Reinforcement Learning is a well-known machine learning technique. It works by developing an action-value function that gives the expected utility of taking a specific action in a specific state. We selected Q-Learning [19] for our investigation. In this technique, the agents keep track of the experienced situation-action pairs by managing the so-called Q-Table, that consists of situation descriptions, the actions taken and the corresponding expected prediction, called Q-Value. Nevertheless, the use of the Q-Learning algorithm is constrained to a finite number of possible states and actions. As a reinforcement learning algorithm, it is also based on modeling the overall problem as Markov Decision Processes (MDP). Thus, it needs sufficient information about the current state of the agent for being able to assign discriminating reward. The Q-Learning algorithm could be implemented by means of the standard high-level behavior language in SeSAM.

However, Q-Learning only gives us rules, mapping the agents perceptions to possible actions and their expected utility, and we need to generate a decision tree representation of the implicit behavior of these rules. Since the number

of generated rules can be large, we suggest to use a post-processing step for improving the analysis of the rule set on a detailed level. In this contribution we focus on using the C4.5 algorithm [2] to generate decision trees, using the rules generated by Q-Learning as the input. For a better description of the requirements and implementation of Q-Learning and C4.5 please refer to [18].

The decision tree, returned at the end of the simulation process for a given agent, is generated by selecting the best rules from the Q-Table (the ones with higher Q-Value) and applying them in the C4.5 algorithm. To do that, we map the individual components of the situation description as the features of the algorithm, and the action corresponding to that situation as the correct classification.

##### B. Genetic Programming

Genetic Programming (GP) is an Evolutionary Computation (EC) technique that aims at solving problems by evolving a population of computer programs. New populations of, hopefully better, programs are created in each generation using the previous generation as the input for transformation operators [20]. Evolution is processed on the basis of the Darwinian principle of natural selection (survival of the fittest) and variations of natural processes, such as sexual recombination (crossover), mutation and duplication.

In our approach, we integrate a standard implementation of the genetic programming functionalities into SeSAM, as described in [21]. Agent behavior programs are represented as decision trees, coded as strings. This way of representing behavior requires the mapping of the agents' perceptions and actions, to the functions (nodes) and terminals (leaves) sets of decision tree. To cope with this, the simulations needs components for: using a decision tree encoded in the string format to determine the actions of an agent; managing the population of behavior programs and evolving the strings.

The GP component provides the set of primitives necessary to handle the decision trees. The central primitive is the **Evaluation** primitive. It receives a decision tree string and the description of a situation from the simulation, and outputs a terminal, that the agent may map to a sequence of action calls. Other primitives implement the creation of new individuals and the genetic operators for crossover and mutation.

We define a unique pool of strings, that are used and evaluated collectively by all agents in the simulation. The finally returned decision tree is the tree with the highest fitness overall generations.

#### V. SCENARIO AND PERFORMANCE

For this contribution we selected a pedestrian evacuation scenario. Although it is a simple problem, it provides a way of evaluating the requirements of the learning techniques and point out the challenges.

##### A. Scenario Description

The scenario, depicted in Figure 1(a) consists of a room (20 x 30m) surrounded by walls with one exit and 10 column-type

obstacles (with a diameter of 2m). In this room a number of pedestrians are placed randomly at the upper half part and shall leave as fast as possible without hurting during collisions. We assume that each pedestrian agent is represented by a circle with 50cm diameter and moves with a speed of 1.5m/sec. One time-step corresponds to 0.5sec. Space is continuous, yet actions allow only discrete directions. We tested this scenario using 2 and 4 agents.

Agents can perceive obstacles and the exit when within a field range of 2m. These perceptions are divided in five areas, as depicted in Figure 1(b). Additionally, the agents hold a binary perception telling them whether the exit is to their left or right side.

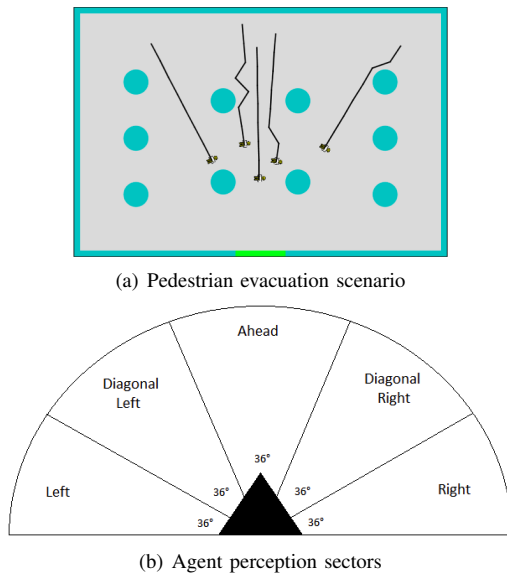


Fig. 1. Scenario and perception sectors

In the reinforcement learning case, all possible perceptions are converted into a string that represents the situation, in the situation-action pairs, or rules, developed by Q-Learning. For the genetic programming case, the perceptions are used as functions, representing the nodes of the decision tree, and are used in combination with the possible actions of the agents, that correspond to the terminals or leaves of the decision tree.

The action set consists of:  $A = \{MoveLeft, MoveDiagonalLeft, MoveStraight, MoveDiagonalRight, MoveRight, Noop, StepBack\}$ . We assume that the agents are per default oriented towards the exit.

For any of these actions, the agent turns by the given direction (e.g., +36 degrees for *MoveDiagonalRight*), makes an atomic step and orients itself towards the exit again.

### B. Learning Evaluation

We evaluate the learning using experiments composed by a series of trials, with 100 iteration steps each, representing the number of steps that the agents have to evacuate the room before a new trial begins. With reinforcement learning, these trials are sequenced as explore and exploit trials. During

explore, agents execute random actions and build their Q-Table with the situations experienced and the actions executed. During exploit the agents select only the best action for each situation, again based on the Q-Table. We used a total of 3000 explore-exploit trial pairs. At the end of the simulation, the Q-Values are used to select the best rules from the Q-Table that will form the training set used to build the decision tree with the C4.5 algorithm.

For the genetic programming case, each trial represents the use of one of the decision trees in the population. When all decision trees in the population are tested, a new generation is created. We experimented with 3000 generations.

The fitness value is assigned as the result of the agents acting in the environment, for both learning techniques. In the reinforcement learning case, this value is given on an action level, feeding the evaluation of the rule, and as a consequence developing the Q-Table. In the genetic programming case, the fitness of the decision tree is given at the end of the trial, as the sum of the fitness collected from all actions performed in that trial.

The rewards are given to the agent  $a$  for executing an action at time-step  $t$ :

$$reward(a, t) = reward_{exit}(a, t) + reward_{dist}(a, t) + penalty_{coll}(a, t)$$

Where:

- $reward_{exit}(a, t) = 1000$ , if agent  $a$  has reached the exit in time  $t$ , and 0 otherwise;
- $reward_{dist}(a, t) = \beta \times [d_{t-1}(exit, a) - d_t(exit, a)]$  where  $\beta = 10$  and  $d_t(exit, a)$  represents the distance between the exit and agent  $a$  at time  $t$ ;
- $penalty_{coll}(a, t)$  was set to 100 if a collision free actual movement had been made, to 0 if no movement happened, and to -1000 if a collision occurred.

Together, the different components of the feedback function stress goal-directed collision-free movements.

All agents perceive the environment in parallel, so their perception is based on the same overall situation at the time-step  $t$ .

### C. Learning Configuration

Q-Learning was set with a linear learning rate function (from 1 to 0.2) and a discount factor of 0, which means the agents consider only the immediate best action. We also used a balanced exploration feature for selecting random action in the explore trials, in a way that all possible actions are tested equally and therefore we have a better confidence on the Q-Value. In the post-processing case, the decision tree must be tested for correspondence to the original situation-action pairs that were used to produce it. This is basically a matter of convergence which is not trivial in our scenario [18]. At the end of the simulation, the best rules – the ones with higher Q-Values – are selected as input for generating the decision tree in C4.5. Additionally, we exclude the rules that have not been tested sufficiently, according to an experience threshold.



For the Genetic Programming case, we defined the population with a size of 30 individuals. That means each generation consisted of 30 trials of 100 steps each. One trial for each individual. The initial population was generated using the *ramped half and half* method. Each individual was represented by a decision tree with a maximum depth of 5 levels. We set the probability for reproduction (copy the individual in the next generation) to 0.5, the probability of crossover to 0.4 and the probability of mutation to 0.1. Tournament selection was used to select the best fitted from 5 random individuals in the population for any of the before mentioned genetic operations.

#### D. A Glance on Results

The first result to be analyzed from the simulations concerns the performance of the learning technique: the number of collisions throughout the simulations. Table I shows the average number of collision: over all the 3000 explore-exploit trials for reinforcement learning (considering only data collected from exploit trials); over all the 3000 generations for the genetic programming approach (considering only the best individual in each generation). We average the number of collisions recorded by all the agents in the simulation. Reinforcement learning requires more time to develop a good set of rules to accomplish the task, while the high level representation in genetic programming is able to cope well with this simple scenario.

TABLE I  
AVERAGE COLLISIONS

	RL	GP
2 agents	$0.62 \pm 1.04$	$0.002 \pm 0.05$
4 agents	$1.28 \pm 1.18$	$0.18 \pm 0.30$

Additionally, it is important to see how the agents behave in the scenario. To do that, we consider example trajectories of an exemplary agent from the 2 agents scenario, at a) 500, b) 1000, c) 2000 and d) 3000 explore-exploit trials in the reinforcement learning case, and generations in the genetic programming case. In the genetic programming case we consider the best decision tree, according to the fitness evaluation, in those specific generations. This is depicted in Figures 2 and 3. One can see how the decision trees perform and how they are **converging** to a good solution.

The evolution of the fitness value is also considered. The Figure 4 shows the fitness distribution – represented by the Q-Value – over all the rules for one exemplary agent in the simulation, in the RL+ case. A low number of rules with high Q-Value reflect the situations where the agent perceives the exit, and by reaching that, the reward gets increased. There is also a number of rules with a Q-Value of 0, meaning the rules have not been tested during the simulation, mainly because the agents did not experienced those situations.

The Figure 5 show the evolution of fitness for the decision trees population in GP, considering the best individuals in each generation. After a number of generations the value stabilizes, however a big variance can be verified, meaning that there is a

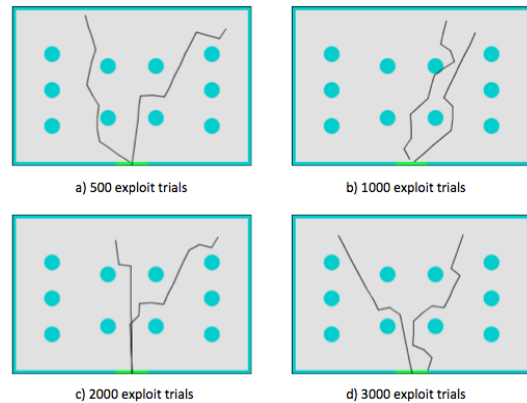


Fig. 2. Agent example trajectories for RL+ with 2 agents, over exploit trials

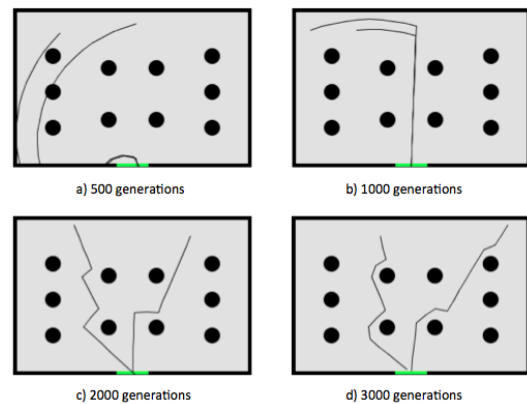


Fig. 3. Agent example trajectories for GP with 2 agents, over generations

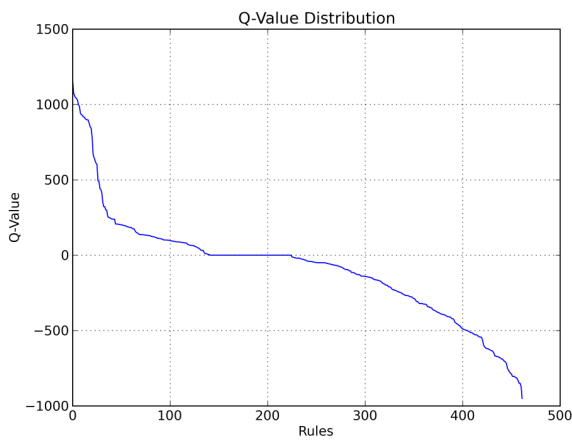
need for improvement on the GP settings, mainly the genetic operators probabilities.

#### E. Evaluation of the Decision Tree

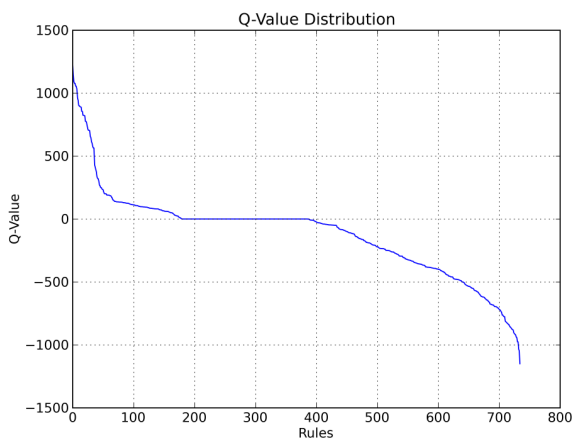
As our objective is not to develop a black box behavior controller for this simple scenario, but to generate advice for the modeler about potential behavior models for the individual agents, it is central to have a look onto the decision trees themselves. That means, we have to analyze the resulting decision trees not just according to their performance in the given scenario, but also their value as a source of inspiration for the modeler.

Clearly, the size and compactness of a decision tree is a relevant descriptor for how good a modeler can analyze its contents. In the RL+ case the size and compactness of the tree is correlated to the number and diversity of situation-action pairs that are used for its generation. This is influenced by the experience threshold, stating how often a situation action pair has to be tested. Actually, this filtered all rules with observations of the exit and just left over situation that the agents more often perceive resulting in trees that are not appropriate for all situations. The internal nodes of the trees refer to single perceptions, thus the compactness in scenarios with only a few agents is reduced as the most frequent





(a) RL 2 agents



(b) RL 4 agents

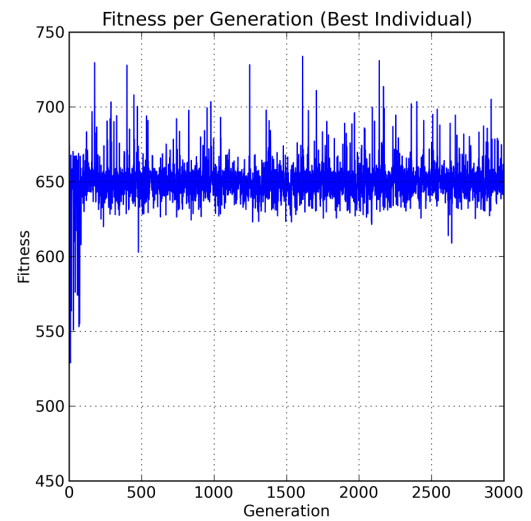
Fig. 4. Q-Value evolution

perceptions only contain at most one other obstacle. This is even the case in the scenarios with 4 agents.

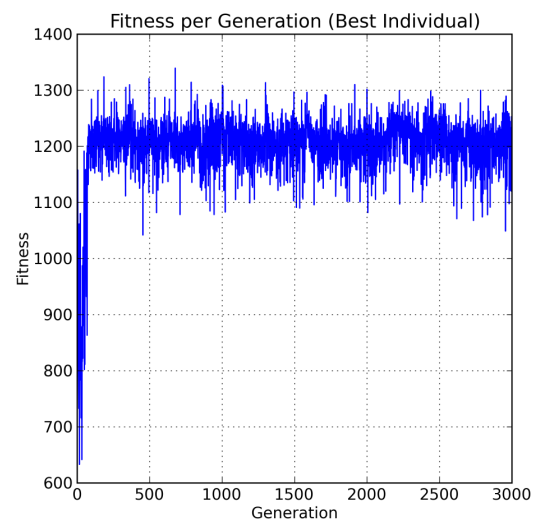
For being readable, the trees learned by the GP approach have to be pruned as well. In all trees that we analyzed, there are many branches that will never be used as they represent conditions that have been excluded before. Thus, the resulting effective size of the tree is much smaller than initially observed. Again, not all situations that an agent can perceive are handled appropriately.

Figures 6, 7 and 8 allow a direct comparison between a hand-made decision tree and examples generated from RL+ and GP.

However, size and compactness are only weak indicators of how well a learned behavior model can serve as a basis for modeling. Only the modeler can finally state whether the learned decision tree contains something "interesting" for the particular modeling problem. In Figure 6 a decision tree is depicted that was created by a human modeler who supposed that this is a good behavior model for the scenario.



(a) GP 2 agents



(b) GP 4 agents

Fig. 5. Fitness over generations for best individuals

It does not just avoid collisions but even considers the coarse direction toward the exit when deciding about the avoidance direction. However, potential problems with other simulated pedestrian, moving to positions where the agent following this tree decided to go in the next step are not regarded. An analysis of the learned trees immediately shows that they are not of the same complexity than the manual tree, but can indeed point to alternative, better solutions. Specially the RL+ tree shows that it is not sufficient to just avoid the obstacle, but the turning behavior must be larger for avoiding immediate collisions coming from movement to colliding sectors. That means an analysis of the learned behavior actually has the potential to help improving the manually developed tree.

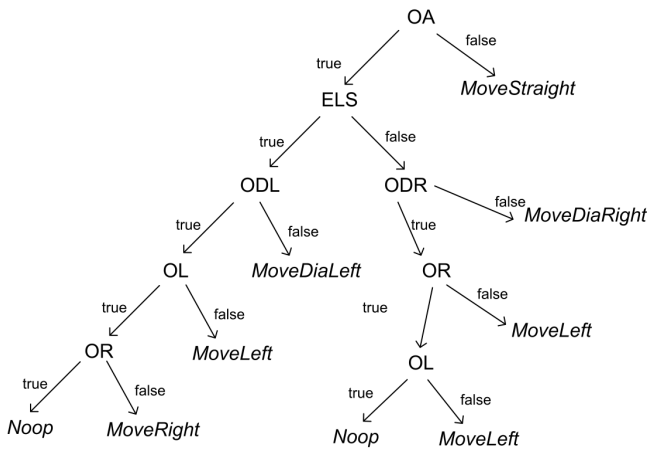
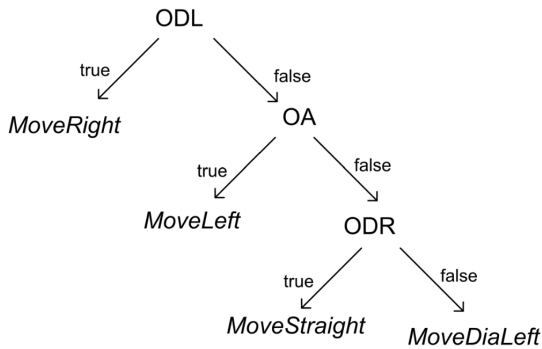
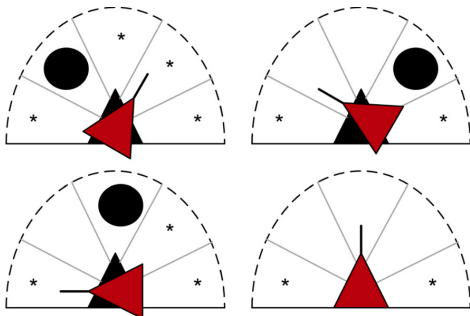


Fig. 6. Hand-made decision tree



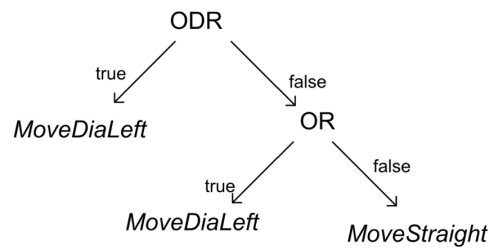
(a) RL+ tree, 4 agents scenario, experience threshold = 150



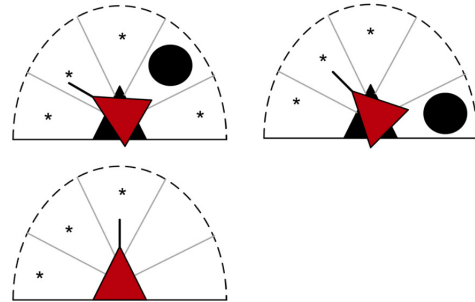
(b) Visualization of the content of the RL+ tree

Fig. 7. RL+ decision tree

Analyzing their contents, both learned trees are far from being optimal and are suspect to assumptions that cannot hold in any case. They both contain movements into sectors that are not tested whether there is an obstacle or not. This shows that learned behavior models may play an important role for detecting bugs in the environmental model, scenario or the functions describing the validity, i.e. fitness of an agent. The example GP tree only considers obstacles on the right side of the agent – a next step must be to test whether there is an inherent bias in the scenario that is responsible for that. The basis for this depicted tree was the best tree found during



(a) GP tree, 4 agents scenario



(b) Visualization of the content of the GP tree

Fig. 8. GP decision tree

3000 generations. Maybe, more generations are needed to evolve a more complete tree. Moreover, we intend to include a check after each genetic recombination operation, preventing the creation of trees with redundant checks in the same branch. This way, evolution can focus on finding relevant relations among the perceptions, considering that redundancies in the original representation are only partially helpful for that.

## VI. CONCLUSION AND FUTURE WORK

In this contribution we compared decision tree based behavior models learned from the results of a Reinforcement Learning approach or directly evolved using a Genetic Programming approach. We not just considered performance as a controller, but also had a look onto the resulting behavior models - going a step further to our original goal of modeling support by generating suggestions for a modeler when he is getting stuck with developing a multiagent simulation model. Clearly, we are just at the beginning of our endeavor trying to find out the appropriate learning techniques for our goal in general and for simulation problems with particular structures.

The next steps are related to further improvements of the learning algorithms. Whereas for RL+ we already did extensive tests concerning effects of different configurations and alternative setups, this has still to be done for the GP approach. A systematic analysis of the influence of the many different parameter configurations and scenario setups should be conducted, deepening the comparison of the two learning techniques. We want to avoid integrating components into the objective function that intentionally influence the shape of the tree. This would involve meta-level considerations, making the development of the fitness function even more complex and would confuse a modeler

as performance and modeling concerns would not be separated.

An important question in our setup concerns the robustness of the learning approach with respect to small changes in the objective function that contains the characterization of valid behavior. It is clear that developing this function is the most difficult task when using a learning-based modeling approach. Therefore it is essential to know how sensitive the learning algorithm is to slight changes in this objective function.

Additionally, we want to test variations of the learning techniques that focus on the modeling support goal. Having in mind the design strategy, starting from the definition of sensors and actuators, and going to a decision tree behavior representation, the learning techniques differ on how they evolve such a model. RL essentially works on developing a set of rules, evaluated individually, that need to be translated to a tree representation. In this translation process we lose information about which were the original rules and what was their assessment. In case the modeler wants to further develop this model by modifying branches of the tree, it becomes difficult to integrate this knowledge back in the learning process. On the other hand, GP evolves directly a tree representation. There is no need for converting information. The human modeler can alter parts of the program and use it back in the simulation for validation. However, the fitness assessment is done in a program level. There is no information about the influence of a particular branch of the tree on the final value. We intend to modify our GP approach to include individual action fitness evaluation – on a similar level as it's done with RL+ – in order to develop an editable tree program that can be interpreted not only on the global performance level. This would represent an important step towards the development of a systematic way to analyze the learned tree and identify elements that should be integrated into the final model. How to present a tree that the modeler is able to evaluate, which are the problems in the manual behavior model and which elements of the learned tree are responsible that the learned tree does not face these problems. Up to now we were mostly focusing on finding the most appropriate learning techniques, supporting the modeler in using the learned results cannot be neglected.

Finally, we will peruse further experiments in more complex scenarios. Complexity, at first, can be increased by: having more agents in the simulation; broadening the perception range of the agents, to include more perception variables; adding more elements to the objective/fitness evaluation; and also by including heterogeneous agents, that are required to perform different roles and are subject to different objective/fitness functions.

GP seems to be more appropriate to the space exploration problem, yet additional processing may be required depending on how the complexity increase will impact the size of the decision tree. A possible solution would be to split the search space, providing different program trees for different sub-

problems. However, this division depends on the problem domain and additional steps would have to be included in the design strategy.

## REFERENCES

- [1] J. M. Epstein and R. L. Axtrell, *Growing Artificial Societies: Social Science from the Bottom Up*. MIT Press, 1996.
- [2] R. S. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
- [3] S. B. Kotsiantis, "Supervised machine learning : A review of classification techniques," *Informatica*, vol. 31, pp. 249–268, 2007.
- [4] F. Klügl, "Multiagent simulation model design strategies," in *MAS&S Workshop at MALLOW 2009, Turin, Italy, Sept. 2009*, ser. CEUR Workshop Proceedings, vol. 494. CEUR-WS.org, 2009.
- [5] G. Weiß, "Adaptation and learning in multi-agent systems: Some remarks and a bibliography," in *IJCAI '95: Proceedings of the Workshop on Adaption and Learning in Multi-Agent Systems*. London, UK: Springer-Verlag, 1996, pp. 1–21.
- [6] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [7] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, ser. Natural Computing Series. Springer, 2003.
- [8] S. Mahadevan and J. Connell, "Automatic programming of behavior-based robots using reinforcement learning," *Artificial Intelligence*, vol. 55, no. 2-3, pp. 311 – 365, 1992.
- [9] M. R. Lee and E.-K. Kang, "Learning enabled cooperative agent behavior in an evolutionary and competitive environment," *Neural Computing & Applications*, vol. 15, pp. 124–135, 2006.
- [10] T. Francisco and G. M. Jorge dos Reis, "Evolving predator and prey behaviours with co-evolution using genetic programming and decision trees," in *Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation*, ser. GECCO '08. New York, NY, USA: ACM, 2008, pp. 1893–1900.
- [11] L. Hanna and J. Cagan, "Evolutionary multi-agent systems: An adaptive and dynamic approach to optimization," *Journal of Mechanical Design*, vol. 131, no. 1, p. 011010, 2009.
- [12] W. H. Hsu and S. M. Gustafson, "Genetic programming and multi-agent layered learning by reinforcements," in *In Genetic and Evolutionary Computation Conference*. Morgan Kaufmann, 2002, pp. 764–771.
- [13] J. Denzinger and M. Kordt, "Evolutionary online learning of cooperative behavior with situation-action pairs," in *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*, 2000, pp. 103 –110.
- [14] J. Denzinger and S. Ennis, "Improving evolutionary learning of cooperative behavior by including accountability of strategy components," in *Multiagent System Technologies*, ser. Lecture Notes in Computer Science, M. Schillo, M. Klusch, J. Müller, and H. Tianfield, Eds. Springer Berlin / Heidelberg, 2003, vol. 2831, pp. 205–216.
- [15] J. Denzinger and A. Schur, "On customizing evolutionary learning of agent behavior," in *Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, A. Tawfik and S. Goodwin, Eds. Springer Berlin / Heidelberg, 2004, vol. 3060, pp. 146–160.
- [16] J. Denzinger and J. Hamdan, "Improving modeling of other agents using tentative stereotypes and compactification of observations," in *Intelligent Agent Technology, 2004. (IAT 2004). Proceedings. IEEE/WIC/ACM International Conference on*, sept. 2004, pp. 106 – 112.
- [17] R. Junges and F. Klügl, "Evaluation of techniques for a learning-driven modeling methodology in multiagent simulation," in *MATES*, 2010, pp. 185–196.
- [18] —, "Learning convergence and agent behavior interpretation for designing agent-based simulations," in *Proceedings of the Eighth European Workshop on Multi-Agent Systems EUMAS 2010*, 2010.
- [19] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, 1992.
- [20] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems)*, 1st ed. The MIT Press, Dec. 1992.
- [21] R. Junges and F. Klügl, "Evolution for modeling - a genetic programming framework for sesam," in *Proceedings of ECoMASS@GECCO 2011. Evolutionary computation and multi-agent systems and simulation (ECoMASS)*, 2011.

# Visualizing Agent-Based Simulation Dynamics in a CAVE - Issues and Architectures

Athanasia Louloudi  
 Modelling and Simulation Research Center  
 Örebro University, Sweden  
 Email: athanasia.louloudi@oru.se

Franziska Klügl  
 Modelling and Simulation Research Center  
 Örebro University, Sweden  
 Email: franziska.klugl@oru.se

**Abstract**—Displaying an agent-based simulation on an immersive virtual environment called CAVE (Cave Automatic Virtual Environment), a human expert is enabled to evaluate the simulation's dynamics from the same point of view as in real life - from a within perspective instead of a birds eye view. As this form of face validation is useful for many multiagent simulations, it should be possible to setup such a system with as little effort as possible. In this context, we systematically analyse the critical issues that a realization of such a system raises. Addressing these problems, we finally discuss design aspects of basic framework architectures.

## I. INTRODUCTION

**P**ERFORMANCE evaluation is important for agent-based simulations [1]. In order to ensure that the model used is able to produce reliable and plausible results, significant oversight from the human expert is required. However, this could easily become a source of great expense, especially in cases involving agent behaviour in explicit metric space such as in pedestrian simulations. To reduce this cost, an immersive visualization, based on multiple interlinked views with different levels of detail, could be considered as a useful tool for evaluating the plausibility of simulation models at the agent level. Due to the high degree of immersion, face validation [2] can be very much facilitated. In this form of validation, one or more human experts assess the model based on animations, simulation output or from a within perspective (e.g., agent's view). Debugging and model evaluation is then clearly benefited by zooming into the system, even into the agents for observing the ongoing dynamics or monitoring the interaction between agents.

While immersive visualization has advanced significantly [3], the creation of a complex and dynamic virtual environment in the form of a multiagent system is not a trivial task as it combines technologies that are not equivalent. It is important for the modeller of a multiagent simulation to be able to focus on the model development and avoid the complexity of how to deal with setting up an immersive visualization. The ideal case would be to establish a connection between the simulation and the visualization interface using minor configurations. The connected systems would then automatically generate the 3D representation of the virtual world from the simulation output, while they would enable an immersive movement of the human observer in the simulation without the need for further adaptation. However, such a combination of systems

introduce technical and conceptual issues which have to be tackled in a profound way, starting from a theoretical analysis. In this contribution we discuss challenges and their solutions for visualizing the dynamics of a multiagent simulation in a CAVE-based virtual environment [4].

The remainder of this work is organised as follows. Initially, in Section 2, we describe the system and its basic requirements. Then Section 3 analyses the particular challenges which may occur when realizing such a system which enables the immersive visualization of an agent-based simulation in a CAVE. Section 4 sketches alternative architectures for the coupled system while the involved challenges are elaborated. In section 5, we discuss our first attempt to create a prototype. In the remainder of this work we discuss the work related to our approach and finally we give a short conclusion.

## II. SYSTEM DESCRIPTION AND REQUIREMENTS

The overall goal of this work concerns the creation of a framework that will be able to visualize the dynamics of a multiagent simulation within the CAVE. Two are the main concerns towards the achievement of this goal:

- 1) The representation alignment of a 2D multiagent simulation with a fine grain 3D visualization platform.
- 2) Human immersion in one representation.

The idea behind this approach is depicted in Fig. 1. It is clear that in order to achieve these tasks it is important to successfully align non identical representations before enabling a human to join the simulation.

Our initial task deals with the coupling of two dynamic representations of the same multiagent system, on different levels of temporal and representational granularity. We consider that *simulation* and *visualization* are two different systems-platforms representing the same multiagent model. The simulation is not embedded into the visualization system, but rather represented separately. This coupled system should communicate explicitly the information related to any given state of the simulated situation, while consistency between the two representations has to be assured.

Simulation is a more qualified representation, with refined object structures. Contrarily, the visualization refers to the 3D representation of the multiagent system using detailed object models and complex animations. It is of great interest to note that the two systems have different characteristics and a clear

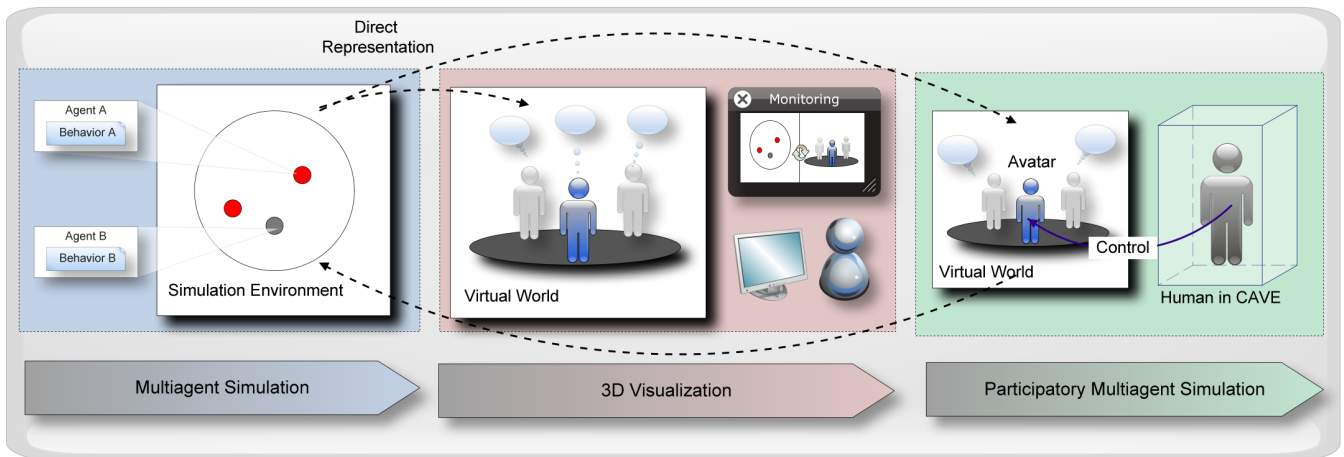


Figure 1. Graphical illustration of the components involved in the system and their relation. Multiagent simulation representation (left), 3D visualization of the same representation; enables better insight for the human modeller (middle), human inside CAVE and participatory multiagent simulation (right). The human has a direct representative in the simulation which he controls (avatar).

Table I  
COMPARISON OF VISUAL ASPECTS BETWEEN MULTIAGENT SIMULATION AND VIRTUAL WORLD

	Simulation	Virtual World
<b>Role of visualization</b>	Add on for illustration	Raison d'etre
<b>Spatial Dimensions</b>	2D/3D, discrete/continuous	3D continuous
<b>Temporal Resolution</b>	Arbitrary	Real-time
<b>Viewpoint</b>	Bird-eye view	Camera entities
<b>Representation Detail</b>	Coarse	Observer-camera distance
<b>Object Models</b>	Points, polygons	Detailed 3d models
<b>Behaviour Models</b>	Complex behaviour	Complex animations
<b>Interaction</b>	Parameter adaptation	Directly with characters

separation of concerns. To indicate an example, one could think of a large scale pedestrian simulation [5] against the virtual world crowd simulation [6]. The simulation aims to represent the behavioural aspects of the modelled system in the simplest but still valid way, whereas in visualization the focus lies on the realistic representation of the 3D object models and their animated behaviour. Table I, gives a brief comparison of the two types of representations based on visualization features.

It is also worth mentioning that both systems should run in a both coupled and decoupled way, on the same or even on different computers. Consequently modularization and online data generation should be placed among the tasks that the final system should accomplish. Based on these functional requirements it is clear that the solution must be more than just connecting agents and environments to a number of specialized default object models and animations, using default infrastructure feeding a preconfigured virtual environment. A generic solution should be developed, that means it should work for different models in several domains.

In the next phase, the focus lies on the immersion of a human in the system. The modeller can join the simulation for instance as an observer, looking through the eyes of a particular agent. In this way the human can see what the agent perceives and how it reacts as a unit. In addition, if the

simulation enables participation (i.e., participatory simulation [7]), then the human can interact with the individual agents in the environment and access information related to their behaviour. The variety of possible interactions is much higher than in macro simulations where only one observation point is possible.

It is evident that developing a framework capable of incorporating such functionalities would offer better insight and it could lead to concrete assessments over the plausibility of the simulation model. However, despite the fact that the change of perspective is a central idea for face validation, there are still several technical as well as conceptual challenges beyond the mere motivation.

### III. CHALLENGES

#### A. Representation Alignment Issues

Focusing on the technical level of this task that is to map abstract concepts in simulation to graphical assets in visualization, one can clearly identify the difference in the level of abstraction. The exact representation of the simulated multiagent model in both systems is clearly inefficient. It is necessary to define a formal way to communicate specific information that is related to the entities and their actions between the two platforms. Communication in that sense is not only the generation and transmission of data but also

the interpretation of it (eg. Agent generation: Female/ Male, Activity: Walk/Run/Idle etc.).

An additional complexity comes from the different temporal resolutions of the two representations. Synchronisation and consistency during runtime are important in the efficiency of the overall result. We organize the issues involved according to the following two dimensions: Modelling phase and Runtime phase.

#### 1) Modelling Phase:

a) *Mapping agents and entities to object models:* Multi-agent simulations with a spatial environmental model, with a few exceptions, are based on 2D representations. In contrary, characters in the virtual world are presented at a higher level of detail, involving realistic object models. Fig. 2 illustrates such an example. This gap has to be bridged, thus it is important to establish a method to efficiently associate agents to their corresponding object models. This entails an amount of information about the object model such as shape, size, scale in order to plausibly visualize the simulated objects/entities. The measures must be corresponding. For example, if a 2D shape is used in collision avoidance, the bounding box of the 3D object model in the virtual world should not be much larger in order to maintain the effects of the collision avoidance behaviour. Hereby, the visualization of entities (i.e., environmental resources) in the virtual world may be more problematic and time consuming for the modeller than expected. That is because the virtual world may involve a rich environment with a large number of heterogeneous passive entities such as obstacles with different shapes, materials and textures, which in simulation may be visualized by the exact same 2D shape.

b) *Mapping agent behaviour to animation:* The behaviour of the agents should also be depicted. In order to have a concrete visual representation, not only the geometries of the object models are necessary but appropriate animations as well. Detailed animations have to be configured (e.g., movement speed, frame rate of display, etc.) and to be connected to the relevant information in the simulation model which may correspond to abstract notions. Additionally, if the internal state of the agent is changing during the simulation run, the animations and geometries should also follow. For instance, assume that we want to visualize the life cycle of a human agent. Morphing operators should be used in order to enable changes in the shape of the object model. A problem rises when internal states that are present in the simulation, do not have any corresponding visual representation in the virtual world. To indicate an example, consider the case in which an agent turns red when a value is below a specific threshold (eg., when the agent is hungry).

Apart from the dynamics driven by the individual behaviour of the characters, the interactions between the agents have to be considered as well. This means that dynamics led by the full process of agents' interaction become a critical issue for visualization (e.g., characters that talk, wave to each other etc.). Additionally, on the simulation side, only aspects of an agent that affect the other agents' behaviour need to be visible

to the modeller, whereas in the virtual world only what is within the observer's field of view is supposed to be visualized. In this case, the level of detail (LOD)[8] plays an important role in the representation of dynamics.

In the same context, the global environmental properties have also to be dealt. A multiagent simulation may involve dynamics that are triggered by special environmental entities which effect globally the overall visualization (e.g., evacuation signal). Then the question of what should be visualized, where and how seems critical.

c) *Rendering and configuration:* The virtual world needs additional configuration in order to provide a rendered scene that preserves realism. Infrastructure such as cameras and lighting has to be adapted according to the demands of each scenario.

2) *Runtime Phase:* Visualization in the visualization system is real-time, measured in frames per second (fps) whereas in simulation the update is arbitrary. This is a problem that has to be tackled. In many applications, simulated time is intentionally different from real time so as to facilitate the testing procedure. Essentially, the simulation time is as fast as it can be (i.e., faster or slower when compared to real-time).

Assuming that the visualization is following the more qualified simulation, we can identify two problematic cases:

- The visualization system is slower than the simulation
- The simulation is slower than the visualization system

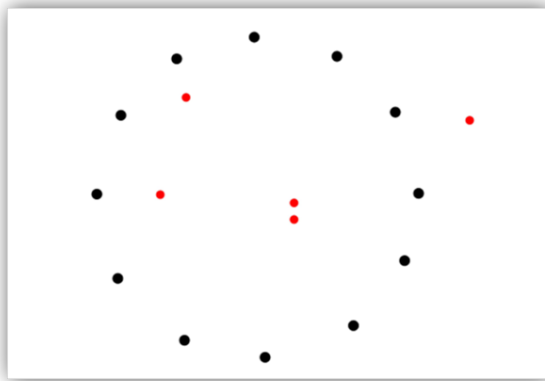
a) *The visualization is slower than the simulation:* This means that the visualization cannot afford the information flow. If we consider that the update difference is not that large, then a possible reason for this problematic situation to occur is that the characters have more elaborated object models whereas their corresponding agents have simple reasoning structures.

b) *The simulation is slower than the visualization:* In this case, the simulation is not able to feed the visualization with data in a timely manner. Depending on the update difference, it is possible that the visualization may have to stop and wait for new incoming data and eventually no real-time visualization is possible. A reason behind this problem could be that the agents hold costly reasoning structures while the characters in the visualization have simple object models. Additionally, in the simulation, there is not stable need for coordination or planning in every reasoning cycle. This may lead to quadratic or even exponential agent update whereas in pure rendering is linear in the number of characters.

#### B. Human Immersion Issues

The idea of involving a human in one representation can influence in a significant way the overall system's operation as this immersion will have an effect on the behaviour of the rest of the agents. In a simple case, the immersed human-avatar should be perceived as an obstacle for the other agents-characters walking around its position. However, if from being simply an observer, the avatar is interacting with other agents, then both its actions and their results have to be transferred back to the qualified representation and to be integrated to the running simulation so that the corresponding agent actually





(a) 2D simulation model



(b) 3D virtual model

Figure 2. Illustration of a simulation model in contrast to its visualization in virtual space

executes the actions induced by the human. As a consequence the system appears with mixed qualifications resulting to a significant increase of complexity. This type of coupling that is bidirectional, can cause an evident problem for the synchronisation; simulation and visualization time must be in line. One of the two platforms has to take over the control with respect to the time advance so as to assure consistency during the simulation run.

After analysing a number of key issues, the remaining question is how such a system could be efficiently realised so as to ensure that the focus will not be shifted from the multiagent simulation and without adding any significant effort to the modeller. These challenges need to receive full attention in our work. In the following section, we are elaborating the problems and discuss potential solutions for coupling the two representations through conceptual architectures of the overall system.

#### IV. ARCHITECTURE ALTERNATIVES

##### A. Architecture A

Simply sending information from the simulation about (dynamic) positioning of the agents to the visualization platform is not sufficient. Due to the different resolution and granularity, such a direct one way connection wouldn't solve the overall task, as information about the current state of the simulated situation (agents, entities, global properties etc.) has to be transferred and processed on the visualization side as well. Moreover, if we assume that the visualization engine is powerful enough to render the scene, an information overflow or lack of data (depending on the simulation time) is very probable to occur and there is no automatic way to avoid it. The most critical problem lies on the fact that this approach does not provide full functionality. The human can only be an observer when connected to the visualization platform. Therefore, we propose an architecture that is capable to handle the transfer of all information between the two systems. Fig. 3 depicts schematically the framework architecture. The basic elements in this conceptual view are:

- Simulation layer
- Visualization layer
- Networking component
- Simulation Visualizer component
- Control layer
- Human in CAVE

1) *Simulation Layer*: This is the layer in which the multi-agent simulation model runs. Apart from the responsibility to drive the system's dynamics, in this side the generation of the scene takes place too. The starting situation of the simulation is stored and managed here. The situation scene should be exported in a format that can be (automatically) imported to the visualization platform.

2) *Visualization Layer*: In the visualization side, the scene has to be configured. Every agent in the simulation and every entity, should correspond to the relevant object model in the virtual world. Additionally the rendering features have to be configured as well. Lights, cameras have to be set up in a way that the scene can be rendered properly. The questions of how many lights are necessary, or where should they be positioned so that the scene is appropriately illuminated, rise. There are a number of algorithms from graph theory that can be used for automatically solve this problem [9]. The camera's configuration can be set to a standard parameter while the speed of the camera movement can be adapted to a reasonable value considering the size of the overall scene.

3) *Networking Component*: The simulation and visualization platforms, are coupled using a client/server communication bus in order to send messages containing respective information from one system to the other.

4) *Simulation Visualizer Component*: Assuming that the basic visualization information shall only be handled within the visualization system keeping the actual multiagent simulation model clean from such information is important. Consequently, the question how to import the information to the visualization platform where no ontological information about the original model might be available arises. To cope with



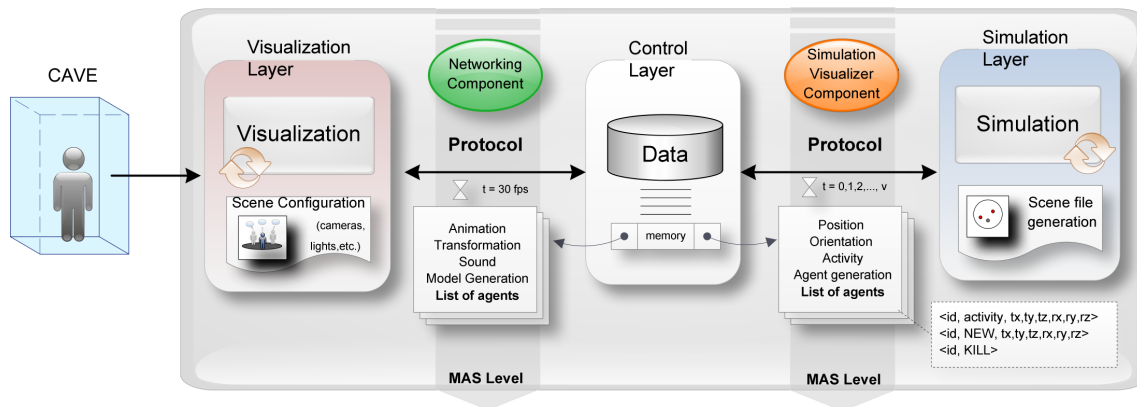


Figure 3. Schematic illustration of the framework architecture A

this problem, we consider the implementation of a component that keeps track of the simulation dynamics in an abstract, yet sufficiently detailed way.

This component is responsible for establishing a connection on the (abstracted) context between simulation and visualization. To be more specific, in every update cycle of the simulation, for every agent, information about its current state (e.g., position, orientation, activity etc.) is sent to the visualization platform. This information can contain the agents id, position, rotation, activity etc.

Each activity token corresponds to an animation that is displayed until a different token for the agent with the given id is received. The character connected to the given id is moving to the new position and orients itself according to the given rotation information. Accordingly, a reserved token could generate or delete agents from the visualization.

Finally, regarding the global properties, a specific message from the simulation may be connected to a particular sequence of changes in the visualization, as for example a signal "fire" might be interpreted as the trigger coming from the global simulation entity world for visualizing smoke in the virtual world.

5) *Control Layer*: Until now we have dealt with the visual representation problems but the synchronization problems still remain. In the case where the visualization system is slower than the simulation, there are several possibilities for dealing with this situation. Clearly, the easiest solution is to reduce the simulation time advance and slow down the simulation. As we assume no inherent connection to the real-time, the simulation time can be reduced. Another alternative could be that the visualization system saves all the incoming information for each character and then renders the events in the correct sequence. Yet, if the buffer of activities is restricted then problems might occur. Information has to be skipped causing gaps in the visualized information. Of course such a solution should be avoided as it wouldn't support the validation of the multiagent simulation model. In the second situation where the simulation is slower than the visualization, the problems are more critical. As the simulation is unable to feed the

visualization in a timely way, one solution could be that the simulation data are being recorded and visualized offline otherwise the visualization has to stop and wait for new data to arrive before proceeding to any action, which is clearly confusing for a human observer.

Another issue to be taken into account, is the granularity of control. We have a qualified representation and a following one. However, the more realistic the visualization shall be, the more the visualization engine will need to take over the control of the details of the interacting characters. Detailed animations with different configurations, morphing, sound, etc. are concepts that are not available in a simulation engine, the question is whether the information for planning and configuring their usage is handled on the simulation side or the underlying reasoning is done in the intermediate components on the visualization side. The information from the simulation has priority, but if there is not sufficient detail, then the visualization engine has to compensate. Therefore, which of the two platforms takes care of which details?

Due to these problems, we introduce in this layer a module that is capable of managing the flow of information between the two systems with a certain degree of reasoning while taking into account the problem of having different temporal resolution. When information arrives arbitrarily from the simulation, the particular data is stored so as to be used when necessary in the visualization engine. Then, if there is a conflict identified, it is the control layer responsible to find an intelligent way to resolve it. For instance simplifying the path of the characters when a number of positions has been received from the simulation that could not yet be visualized, deciding which agents are displayed on which level of detail. When the simulation is slower than the visualization, the representation component may extrapolate the behaviour of the agents based on an estimation where the next steps of the agents may be oriented towards.

6) *Human in CAVE*: Taking into consideration the architecture proposed, an asynchronous operation of the two systems would imply the need to have a roll-back function in the simulation side. Imagine a case in which the simulation is

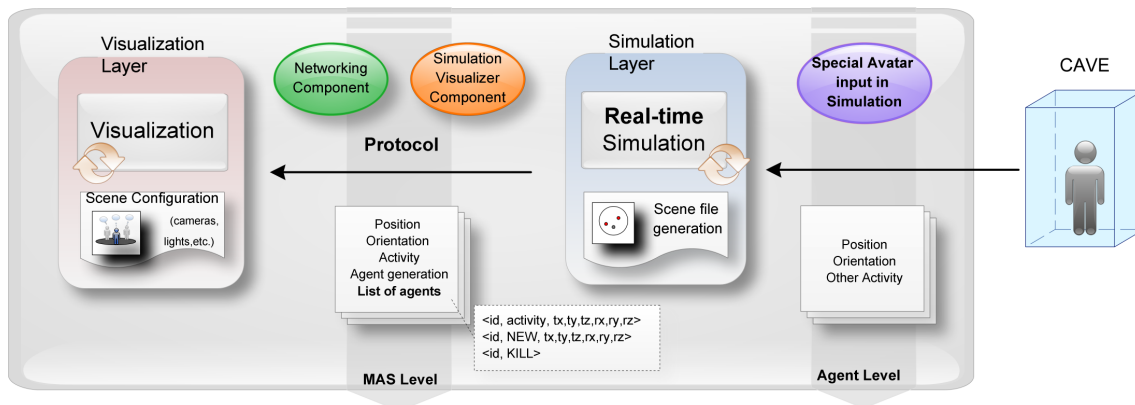


Figure 4. Schematic illustration of the framework architecture B

processing the agents state in  $t_S = 100$  while in the user connected in the visualization platform, is changing the flow of actions in  $t_V = 10$ . The simulation has to adapt to this change and continue from  $t_S = t_V = 10$ . Nevertheless, despite the increase of complexity, the active human immersion in the system appears very attractive and with great potential in the process of evaluating the plausibility of a model and thus such prospect has to be considered in future work.

### B. Architecture B

The second alternative is depicted in Fig. 4 and proposes a completely different architecture. Main characteristic in this schema is the involvement of a human user in the more qualified representation; the simulation layer. In this case the human controls one agent, perceives what the agent may perceive and manipulates the simulation through the agent's effectors.

The simulation layer is enhanced by the use of a *Special Avatar Input Component*. This component enables the human immersion in the agent-based simulation. Information is sent via the sensors in the CAVE affecting the models with primitive calls.

In this schema, we still have to deal with all the problems of configuring the simulated scene in both platforms (visual representation issues) similarly to the previous architecture. The *Simulation Visualizer Component* similarly to architecture A, receives the information from the simulation platform and visualizes it an appropriate way.

Central to this approach is that the simulation has to be adjusted so that it runs in real-time. The time resolution issue between the two platforms is hence eliminated and there is no need for a *Control Layer* as in the previous architecture. In a technical level, a time advance that is similar to real-time is advisable for validation purposes as this is what is the least confusing or distracting the human observer from the dynamics.

An important parameter that should be taken into consideration in the proposed architecture is that the visualization is depending on the simulation in order to start. This means that the visualization system is not treated as an active platform but

it rather plays the role of an external visualization component upon the simulation which doesn't need to hold any status. In addition to that, the functionality of the overall system, is totally based on the simulator used and the generic character of the framework is totally depending on the protocols used.

### V. PROTOTYPE

In an initial attempt, we tried to realize the generic coupling of two such systems in a prototype. Our work was grounded by the use of SeSAM<sup>1</sup>; a modelling and simulation platform and the Horde3D<sup>2</sup> GameEngine.

It uses client-server communication for transmitting information about the visible effects of agent actions from the simulation to visualization. As depicted in Fig. V the server maintains the overall scene and updates are sent by the simulation client. The scene is originally created within the simulation and then transformed into the game engine scene format. To do so, we developed an export function that generates a description of the visualization scene out of the simulated situation. Tokens replace object models, that means pointers to the object models have to be inserted manually. However, as long as there is no automatic generation of geometries from the simulation engine, this export is restricted as it assumes that appropriate object models are existing. A future version must either generate appropriate object models or must be able to scale object models based on the precise information from the simulation situation. In the latter case, simulation entities might have to be augmented with a height value describing the scaling. The protocol used here includes each relevant entity's position, orientation and animation in each update. It also contains events that are sent only once, such as changes in the environment or the generation or removal of agents from the world scene.

The next step was to bring the Horde3D GameEngine to an immersive virtual environment where a human can observe the simulation model from a very close point of view. To accomplish this task, a pre-existing multiplayer architecture

<sup>1</sup>SeSAM: <http://www.simsesam.de>

<sup>2</sup>Horde3D GameEngine: <http://hcm-lab.de/projects/GameEngine>

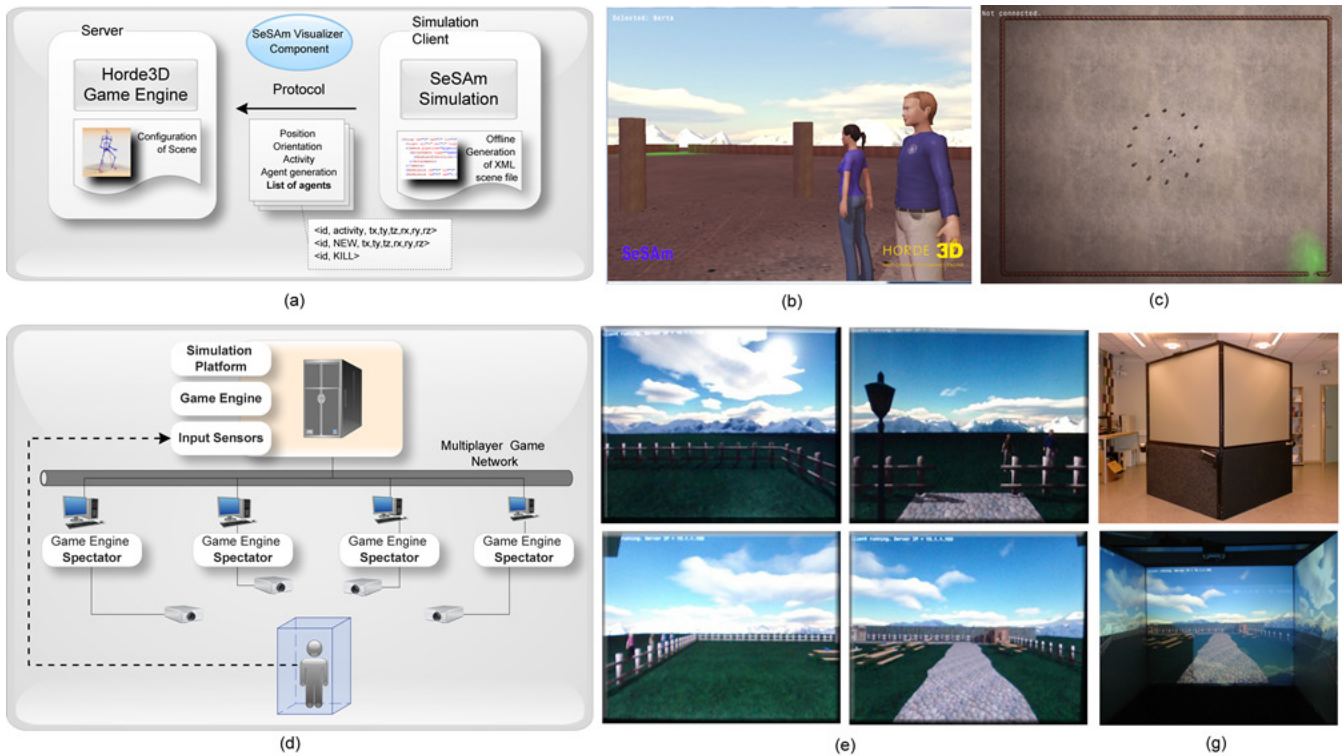


Figure 5. Prototype aspects. (a) Initial architecture; SeSAM-Horde3D GameEngine coupling, (b) Observation of dynamics through agent's perspective, (c) Visual representations of the same multiagent model as seen from bird's eye view, (d) Horde3D GameEngine in the CAVE, (e) The four "spectator's view", (f) CAVE (upper) and scene projected in the CAVE (down).

within the GameEngine was utilised in order to project the simulated situation to the CAVE. The protocol in this multiplayer architecture is state-based, i.e. the updates are sent at regular intervals, each containing the status of every relevant agent or entity in the scene. We use one server and five clients. The server hosts the virtual scene and is responsible for distributing changes happening in that scene to the clients. Four of them simply render the scene provided by the server in each frame without providing input on their own. This mirrors the concept of "spectators" in multiplayer games and these four clients are used to provide the data for each of the CAVE's projectors. The fifth client is connected to the CAVE's sensors and provides the sensor input for the human to navigate in the scene.

## VI. RELATED WORK

### A. Multiagent systems, visualization and 3D virtual worlds

Multiagent models in physically simulated 3D worlds are popular since the seminal evolving creatures of Karl Sims [10]. Later developments in microscopic pedestrian simulation produced an increasing need to provide 3D spatial representations as a visualization method on generic multiagent simulation platforms such as Repast<sup>3</sup> or MASON<sup>4</sup> which

embedded Java3D into their simulation platforms. Similarly, game engines were used but mostly as a mean to model the environment of the agents, while sensor data and action commands are communicated between the agent and the game engine [11]. Contrarily, in our case, the game engine is responsible only for visualizing the dynamics of the simulated situation that is apparently modelled in an external simulation platform. The combination of virtual worlds and simulation is prominent in crowd simulation as well [6],[12]. Main consideration in our approach is the generic character of the solutions thus our system should not just be applicable for crowd simulations.

Korhauer et al. [13], give design guidelines for multiagent simulation visualizations, adapting general design principles about shapes and colours of the agents, grouping the entities for giving advice when visualizing agent system simulation dynamics. Our research is also related to early works on user interfaces for multiagent systems. Avouris [14] classify different multiagent system architectures for identifying special challenges in designing interfaces to multiagent systems - including multiagent simulations focusing on the bird's eye view.

Consistency plays a major role in our work. Thereby, a relation exists between the proposed problem and solutions from the area of distributed interactive systems such as mul-

<sup>3</sup>Repast: <http://repast.sourceforge.net/>

<sup>4</sup>MASON: <http://cs.gmu.edu/eclab/projects/mason/>

tiplayer network games, where consistency has to be assured for presenting the same situation to different users. Several techniques have been developed for avoiding or dealing with inconsistencies coming from latency and jitter [15],[16]. In our case, we identify the major problems to be the different resolutions and synchronization problems between two full representation of the same system, not a distributed representation. Nevertheless, the synchronization problem has similarities with the consistency problem of distributed interactive systems. Clearly we will have a closer look onto techniques of dead reckoning, etc.

The idea underlying our work is relevant to some degree with the principles of the Model View Controller (MVC) paradigm [17]. MVC design pattern have been widely used in Web applications which promoting the separation of visual presentation from logic. However our work has a core difference. The simulation platform is already separated from the visualization.

### B. User involvement in agent-based simulation

Our vision is similar to Repenning and Ioannidou [18] who aim at enabling an end-user to create complex visualizations. They are proposing a tool that facilitates the distortion of existing object models for creating process visualizations in an accessible way. They are apparently addressing the same step in a simulation visualization process that is directed by professional in animation programs such as Maya or 3D Studio Max<sup>5</sup>.

Moreover, several methodologies on how to incorporate human actors in large scale simulations with autonomous agents are present in literature [19],[20]. In most of the cases, the agents are controlled by a human sitting in front of a computer. However, we consider a full immersion of a human in a CAVE. Nevertheless, in our approaches the main consideration is the model validation and the evaluation of the plausibility.

## VII. DISCUSSION AND FUTURE WORK

In this work we analysed the challenges which have to be solved when the dynamics of an agent-based simulation are visualized in a CAVE. An immersive face validation complements the usual data-driven validation on the macro level due to the fact that it allows to check the plausibility of individual behaviour trajectories. Several architectures were also discussed with the main goal to frame the design process towards the realization of the intended system.

Our future work is oriented towards the collection of the building blocks for a modelling language that supports the generic connection to animations. The testing and evaluation of the deployed system is also an important aspect of the coming work. Tests are going to be performed in a simple scenario (eg. evacuation or flocking model) with different numbers of agents/characters ranging from 5 to 100 individuals.

## VIII. ACKNOWLEDGEMENTS

This research work is part of the "Human-in-the-Loop Modeling and Simulation" project funded by MINNOVA and whose support the authors gratefully acknowledge. We also thank Augsburg University for offering their Horde3D GameEngine used in this work and particularly Michael Wißner for his valuable help.

## REFERENCES

- [1] O. Balci, "Validation, verification, and testing techniques throughout the life cycle of a simulation study," in *Simulation Conference Proceedings, 1994. Winter*, dec. 1994, pp. 215 – 220.
- [2] F. Klügl, "A validation methodology for agentbased simulations," in *Proceedings of the 2008 ACM Symposium on Applied Computing*, R. L. Wainwright and H. Haddad, Eds. ACM, 2008, pp. 39–43.
- [3] T. Yapó, Y. Sheng, J. Nasman, A. Dolce, E. Li, and B. Cutler, "Dynamic projection environments for immersive visualization," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, june 2010, pp. 1 – 8.
- [4] H. Lee, Y. Tateyama, and T. Ogi, "Realistic visual environment for immersive projection display system," in *Virtual Systems and Multimedia (VSM), 2010 16th International Conference on*, oct. 2010, pp. 128 – 132.
- [5] F. Klügl and G. Rindsfuser, "Large scale pedestrian simulation," in *Proceedings of MATES 2007*, ser. LNAI, J. Müller, P. Petta, M. Klusch, and M. Georgeff, Eds., vol. 4687. Springer, 2007.
- [6] N. Pelechano, J. Allbeck, and N. I. Badler, *Virtual Crowds: Methods, Simulation, and Control*. Morgan and Claypool Publishers, 2008.
- [7] M. Berland and W. Rand, "Participatory simulation as a tool for agent-based simulation," Setubal, Portugal, 2009, pp. 553–7.
- [8] M. Wißner, F. Kistler, and E. André, "Level of detail ai for virtual characters in games and simulation," in *Proceedings of the Third international conference on Motion in games*, 2010, pp. 206–217.
- [9] *3D Game Engine Design, Second Edition: A Practical Approach to Real-Time Computer Graphics*. The Morgan Kaufmann Series in Interactive 3D Technology.
- [10] K. Sims, "Evolving 3d morphology and behavior by competition," in *Artificial Life IV Proceedings*, R. A. Brooks and P. Maes, Eds., 1994, pp. 28–39.
- [11] E. Norling, "Capturing the quake player: using a bdi agent to model human behaviour," in *Proc. of AAMAS'03*. New York, NY, USA: ACM, 2003, pp. 1080–1081.
- [12] D. Thalmann and S. R. Musse, *Crowd Simulation*. Springer, 2007.
- [13] D. Kornhauser, U. Wilensky, and W. Rand, "Design guidelines for agent based model visualization," *Journal of Artificial Societies and Social Simulation*, vol. 12, no. 2, 2009.
- [14] N. M. Avouris, "User interface design for DAI applications," in *Distributed Artificial Intelligence: Theory and Practice*, N. M. Avouris and L. Gasser, Eds. Kluwer Academic Publisher, 1992, pp. 141–162.
- [15] C. Diot and L. Gautier, "A distributed architecture for multiplayer interactive applications on the internet," *Network, IEEE*, vol. 13, no. 4, pp. 6 –15, jul/aug 1999.
- [16] D. Delaney, T. Ward, and S. McLoone, "On consistency and network latency in distributed interactive applications: a survey–part i," *Presence: Teleoper. Virtual Environ.*, vol. 15, pp. 218–234, 2006.
- [17] A. Doray, "The mvc design pattern," in *Beginning Apache Struts*. Apress, 2006, pp. 37–51.
- [18] A. Repenning and A. Ioannidou, "End-user visualizations," in *2008 Int. Conf. on Advanced Visual Interfaces (AVI 2008), Napoli, Italy*. ACM Press, 2008.
- [19] P. Guyot and A. Drogoul, "Multi-agent based participatory simulations on various scales," vol. 3446 LNAI, Kyoto, Japan, 2005, pp. 149 – 160.
- [20] T. Ishida, Y. Nakajima, Y. Murakami, and H. Nakanishi, "Augmented experiment: Participatory design with multiagent simulation," in *Int. Joint Conf. on Artificial Intelligence (IJCAI-07)*, 2007.

<sup>5</sup>Autodesk: www.autodesk.com



# SimConnector: An Approach to Testing Disaster-Alerting Systems Using Agent Based Simulation Models

Muaz Niazi  
University of Stirling,  
Scotland, UK  
Email:  
man@cs.stir.ac.uk

Qasim Siddique  
Foundation University,  
Islamabad  
Email:  
qasim\_1987@hotmail.com

Amir Hussain,  
University of Stirling,  
Scotland, UK  
Email:  
ahu@cs.stir.ac.uk

Giancarlo Fortino  
University of Calabria,  
Italy  
Email:  
g.fortino@unical.it

**Abstract**—The design, development and testing of intelligent disaster detection and alerting systems pose a set of non-trivial problems. Not only are such systems difficult to design as they need to accurately predict real-world outcomes using a distributed sensing of various parameters, they also need to generate an optimal number of timely alerts when the actual disaster strikes. In this paper, we propose the SimConnector Emulator, a novel approach for the testing of real-world systems using agent-based simulations as a means of validation. The basic idea is to use agent-based simulations to generate event data to allow the testing of responses of the software system to real-time events. As proof of concept, we have developed a Forest Fire Disaster Detection and Alerting System, which uses Intelligent Decision Support based on an internationally recognized Fire rating index, namely the Fire Weather Index (FWI). Results of extensive testing demonstrate the effectiveness of the SimConnector approach for the development and testing of real-time applications, in general and disaster detection/alerting systems, in particular.

## I. INTRODUCTION

IF the past decade were to serve as an example, it is clear that it is perhaps impossible to be over-prepared for disasters. Regardless of the available technologies and preparations, disasters, natural or otherwise are known to hit any part of the world without any prior warning. At times, perhaps getting an early warning might be of little use but at others, even a few additional minutes of warning can make a difference in saving the lives of hundreds, if not thousands of human beings. The earlier a disaster can be detected, the higher the chance of saving precious lives.

However, intelligent detection of disasters is only part of the problem. The other problem is how to effectively test systems which have been developed specifically for responding to disasters. As an example, recently in the case of Mentawai tsunami, because of a lack of an efficient alerting system to go along with the already installed sophisticated detection system, a large number of casualties occurred. According to the Jakarta Post:

“Two days after the disaster struck, the death toll listed 282 and 411 were missing. The tsunami badly damaged 25,426 houses, flattened six hamlets and forced 4,500 residents to evacuate to makeshift shelters”.

According to an official report:

“The most sophisticated system currently available needs five minutes to process information from an earthquake before issuing a tsunami warning — and issuing a command to respond to the field would take more than 15 minutes. It would have been too late for Mentawai.”

Thus in the Mentawai case, there could perhaps be still possibility of saving a number of lives if there were an efficient early warning and alerting system, which would have generated alerts earlier on. So, if people were given say 10 minutes of warning using possibly almost real-time alerts, these might have helped save at least some of these lost lives. So, the key problem here is that while there were intelligent disaster alerting systems available for early warning, how to test such systems in the absence of actual events.

While any software system needs to be thoroughly tested before it is deployed, the case for testing an entire early warning system end-to-end is extremely strong. However software testing of user-driven software requires user testing. In the case of software driven into action primarily by real world events such as disasters, it can be fairly hard to replicate such events. If such an event is simulated manually by say, a person, generated the event, it is not guaranteed to test the system enough. Although considerable data collection exercises have previously taken place, to the best of our knowledge, there is no direct way of using this data to actually test such a system using it. As such, perhaps, the only way to test these applications is by the use of simulation of such rare events. While there are a number of simulation approaches, agent-based modeling and simulation is well-known to develop models of complex adaptive systems. As such, in this paper, we propose the Simulation Connector Emulator approach. The Simulation Connector approach was engineered in response to a practical problem faced in the testing of an actual forest fire alerting system. This approach is proposed as an advancement in Simulation-based Software Engineering by allowing testing of real-world systems using agent-based simulation models. The idea is based on using a validated agent-based model to simulate disaster events in

real time and subsequently using this real-time data stream to test the latency of the response times of the actual disaster alerting system. In other words, this approach assists in the testing of disaster response systems by allowing for testing scenarios using a simulation of rare and unpredictable real-world events. The parameters of the agent-based simulation can be subsequently validated[3], tweaked and calibrated to match with different disaster scenarios[4].

As proof of concept of the SimConnector approach, we have developed it in conjunction with a comprehensive forest fire detection and alerting system. Using an agent-based simulation model, we simulate forest fires and extract the Internationally recognized Fire Weather Index (FWI) [5] values from the simulation. The forest fires simulation has been developed to give extensive variability by catering for a large number of factors responsible for the duration and intensity of forest fires. These include tree cover re-growth, snowfall, rainfall, wind speed, fuel types and other parameters. Using the SimConnector approach, the simulation model is connected to the distributed forest fire alerting system; a system which generates alerts for forest professionals by making intelligent decisions based on the FWI values retrieved from the simulator in real-time. Our extensive simulation experiments demonstrate the usefulness and effectiveness of the proposed architecture in the design, development and testing of disaster-alerting systems. Results showing the reduced latency in the forest fire alerting system demonstrate the effectiveness of the proposed architecture in the testing of the disaster alerting for real-time monitoring of rare events.

The structure of rest of the paper is as follows:

In the next section, we provide the background followed by a discussion of the system architecture. In the next section we present a discussion of the simulation experiments and results followed by a conclusion.

## II. BACKGROUND AND RELATED WORK

In this section, we discuss basic background information and related work.

### A. Tackling Disasters

Disasters can be classified in many different ways[6]. However, regardless of the specific type of disaster, recent examples of disasters around the globe clearly demonstrate the need of care in the designing of alerting systems. Designing software systems with real-time early warning and alerting capabilities can be challenging. These systems cannot be classified as traditional software systems due to a number of reasons. Firstly, such systems require both hardware as well as software to provide decision support to the human end users but they are unlike traditional embedded systems. Secondly, not only do they need to provide real-time monitoring and alerting capabilities, the events which these systems are expected to report are rare events. These events are rare because they may perhaps

occur only once in the future of the system (or hopefully never). Thus, while the systems are required to be very robust, most times, there is not a whole lot that can be done to test these systems in real-world situations before an actual deployment. As such, in the absence of real test situations, it is logical that designers of such systems need to rely extensively on novel approaches such as simulation-based software engineering[7]. However, this approach is not as well studied in literature as is desirable[8] especially in this particular use case of disaster alerting systems.

### B. Simulation Design of Disasters

Among the various approach to the simulation of disasters for the development of disaster alerting systems, designers are faced with two key possibilities:

- The first possibility is to develop a specialized simulator from the grounds up (e.g. using some high level language such as C++, Java or C# etc.).
- The second option is to use a general purpose methodology to develop a simulator.

While the first option is guaranteed to follow the design and development of the system closely, it can also lead to several setbacks. Firstly, such an approach can result in considerable extra costs, something which might not be available for disaster alerting systems, which are expected to be funded by public funds. Secondly, technically speaking, the reliability of custom-built simulators is, at best, debatable since they will essentially bypass a peer-review or community review, something which could have actually helped improve the design of the system in addition to the discovery and eventual removal of any software bugs. As a result, additional testing might be needed for testing the software raising the costs.

### C. Choosing a modeling paradigm

In terms of using a general purpose modeling paradigm, it would be useful if the chosen paradigm qualifies based on certain specific requirements. Unlike modeling of traditional software engineered systems, the peculiar modeling paradigm might entail developing of system models capable of flexibility as well as the representation of complex human social systems as well as computing infrastructures. One such suitable paradigm is the agent-based modeling and simulation (ABM) paradigm[9]. ABM (or ABMS or IBM as it is called in various communities) reflects a specific modeling paradigm which has found extensive use in a large number of scientific domains as described in a recent visualization-based survey on agent-based computing[10]. It has previously been used successfully by researchers in domains as diverse as Biological systems[11], Ecology[12], Social Sciences[13], Business Systems[14] and Computer Sciences[15, 16]. It has been labeled a revolution in the esteemed Journal Proceedings of the National Academy of Sciences [17]. Agent-based models have also previously been shown to model the quantification of complex concepts such as emergence in complex adaptive systems [18]. ABM has also previously been used as a stand-alone simulation tool as

well as for decision support for non-technical Biologist end-users[19]. It has also found extensive use in the discovery of emergent behavior in systems consisting of a large number of interacting entities[20]. PASSIM, a simulation-driven methodology for building Multiagent Systems has been discussed in [21]. However, to the best of our knowledge, ABM has not been previously been used in the context of testing and evaluation of distributed disaster alerting systems.

Agent-based modeling is an advanced programming paradigm where the focus of the simulation design is on modeling individual entities as “agents”. In other words, it allows for a more realistic modeling of real-world objects and complex systems. Agent-based development entails development of behaviors of individual agents, which change the internal state of the agents as well as interact with the environment (e.g. In the case of a programming environment agents might consist of patches, an abstraction quite commonly used in Logo-based agent-based modeling environments[22]).

*D.Related Work*

Rinaldi et al. discuss how models and simulations can provide considerable insight into the complex nature of the behaviors and operational characteristics of critical infrastructures [23]. Bodrozic et al. uses a multi-agent system for data retrieval and processing in forest fire monitoring [24]. Schaeffer et al. discusses the use of extensive simulations to give recommendation on tuning wireless sensor network parameters on the issue of false positive detection due to malfunctioning sensor networks [25]. Angayarkkani et al. approaches the detection of forest fires from spatial data using a combination of data mining, image processing and artificial intelligence techniques [26]. Torii et al. proposes a multi-layer cellular automata based socio-environmental simulation by layering the social interaction scenario on environmental simulation in the domain of fire fighter simulation[27]. One important result in this paper is the control of flow of information between two systems in developing various types of simulations. Hochrainer et al. examines catastrophe modeling and how it implicitly incorporates simulations due to the sheer complexity of the system to be analyzed [28]. Cossentio et al. describe how simulation can be useful in the design and development of multiagent systems [29]. Mata et al. present an approach to forest fire prediction using Case based reasoning[30].

**III.SYSTEM ARCHITECTURE**

This section presents the proposed architecture for the SimConnector in the light of an actual application system. We have chosen the application domain as Forest Fire Detection and Alerting. One key benefit of this domain is the availability of a large amount of historical data as well as a large number of previous studies on forest fire simulations.

The components of the proposed system are shown in Figure 1. The Forest Fire Detection System is made up of several key components such as two separate User Interfaces

(web and Windows based applications) front ends, a backend database, a web service and the Simulation based Decision Support System (DSS). DSS is based on the SimConnector in addition to an Agent-based Simulation model for intelligent fire detection.

The individual components are given as follows:

1. Agent Based Model(Forest Fire Simulation)
2. Decision Support System and SimConnector
3. Windows UI & Web UI
4. Web Service

In the following subsections, we give details of these components.

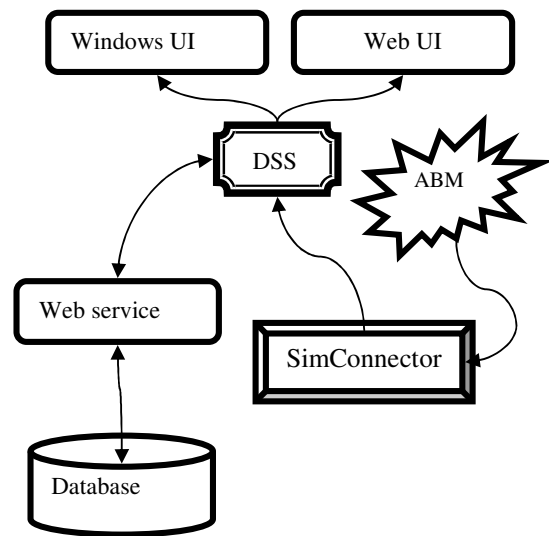


Figure 1 Main Components of Forest Fire Detection System

*A. Agent Based Model (Forest Fire Simulation)*

The basic implementation of this fire spread was extended from an open source NetLogo model[31]. However the model was extended significantly by adding the measurement of indices as well as numerous factors which can affect actual fires. This model was validated using a Virtual Overlay Multiagent system [4]. The model is based on a Cellular Automata (CA) model of a forest. As time progresses, fires appear in the model. Fires affect a random area based on specified spread rates. However the forest fires are validated using a Forest Fire standard index, namely the Fire Weather Index (FWI). FWI is a means of measuring fire intensity based on a number of different factors. FWI is calculated using complex mathematical calculations. One benefit of using FWI is that it is considered as a standard fire index in a large number of countries and has previously been used as a measure suitable for use by Wireless Sensor Networks for forest fire detection[32]. As soon as the fire occurs, a change in the local temperature is detected using a



simulated Wireless Sensor Network. Forest fire is also affected by snowfall and rainfall, which are also simulated and can be adjusted by means of various parameters.

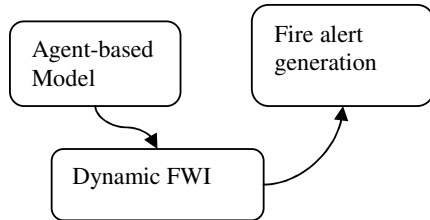


Figure 2 Main components of the Decision Support System

### B. Decision Support System & SimConnector

The detection of forest fires in the proposed system is based on the Decision support system. The simulation environment has a number of distributed agents simulating wireless sensor nodes. These sensor nodes (agents) in the simulation environment detect parameters such as current temperature, rain, wind speed, average humidity subsequently used to calculate the FWI. The calculated FWI is sent over to the SimConnector. The SimConnector is responsible for communicating the fire information for onward generation of fire alerts as can be seen in Figure 2.

As the DSS sends fire information to the actual forest fire detection and alerting system, the desktop application depicts a graphical display of the placement of the sensors in the forest as well as the locations where forest fires have been intelligently detected.

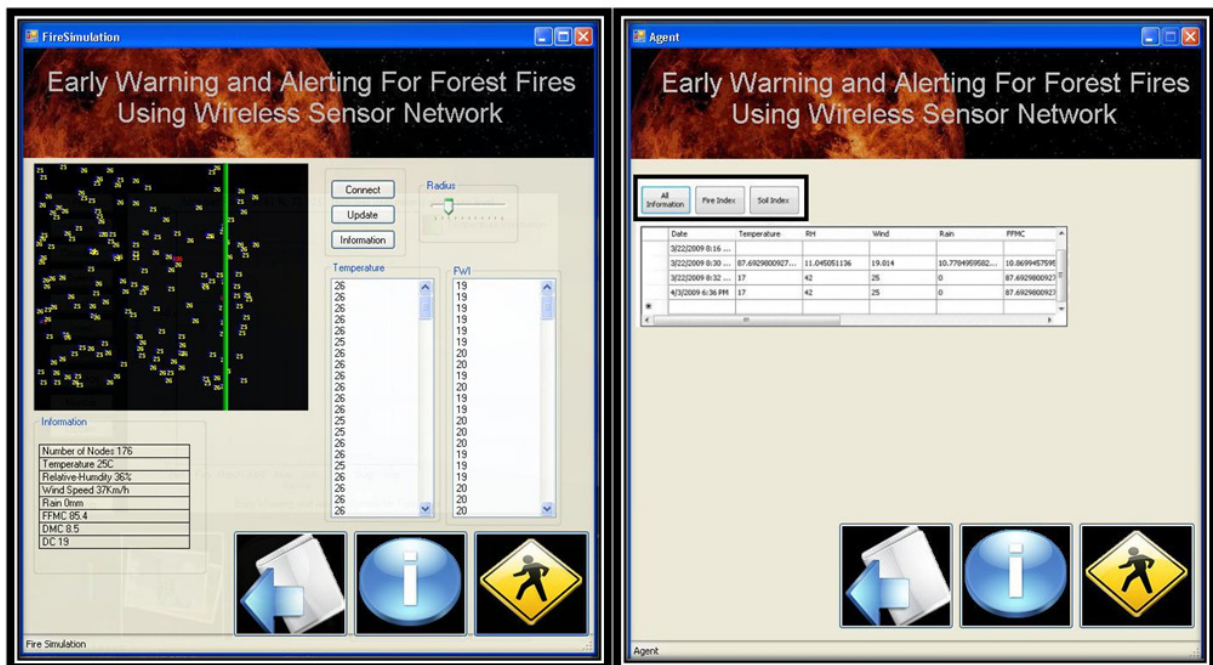


Figure 3(a) Windows UI detecting current location of the agent present in the model (b) UI demonstrating the various Fire Indices calculated dynamically.

### C. Windows UI & Web UI

The design goal of the Web and windows Application are based on a continuous monitoring of the forest using the FWI as an indicator of fire. If the FWI value is detected as within a dangerous level, then the desktop application sends a warning message to the web application.

The Figure 3a shows the Windows User Interface. On the left side of the UI, we can see the location of various simulated wireless sensors. The moving green line represents the scanning of the intelligent DSS as it uses SimConnector

to get data about specific coordinates. The update, connect and information buttons are used to select between a continuous update or for refreshing the system if continuous updates is not selected. The Windows UI also depicts the actual values of two parameters (the temperature and fire Weather index) for easy validation of fires by the user. All information is updated in the user interface on a periodic basis. The grid view at the lower left presents the average information retrieved from the sensor nodes deployed in the forest, the average temperature of the forest, average humidity, the wind speed and rain measurements thus

allowing the user to have a complete picture of the current situation in the simulated forest.

We can see the results of the calculated Fire indices in Figure 3b. The system also contains user interface for web and mobile devices for use by emergency support personnel accessing the website. This information is again tied in with information received via the intelligent decision support based on the FWI.

*D. Web Service*

To have a uniform API for data retrieval, a web service has been developed which connects to the database for storing and retrieving data. The API also allows the user to execute queries against stored FWI Information. Building a client for a web service is easy with the availability of automated tools such as Microsoft Visual Studio.NET which automatically generate client code based on a Web Service Descriptive Language (WSDL) page exported by the web service. By using a standard B2B interface, new clients can easily be developed in the future using other web services or else clients using other languages/technologies.

IV. RESULTS & DISCUSSION

In this section, we evaluate various aspects of the SimConnector implementation using extensive simulation experiments. The goal of using SimConnector is to allow for the performing of real-time black box testing evaluating the timing of the alerts generated by the system for comparison with the time when the forest fires are actually simulated.

*A. Experimental design*

The goal of the first set of simulation experiments is to empirically discover the latency in the fire detection. The latency measurement is the time taken from the simulated fire to the time the system actually generates alerts which are transmitted to the web and windows User Interfaces.

In addition, the forest fire area was also calculated so that a variety of forest types and scenarios can be evaluated. The calculations are based on finding the total number of trees in the simulation. Once the fire is ignited and the tree begins to burn out after a specific time interval, the number of tree burn out due to the fire allows for the calculation of the forest fire area as given below:

$$A_f = A_{f1} - A_{f2} \tag{1.1}$$

Here, in this equation  $A_f$  represents the area of the fire. The area is found by subtracting the area covered by the forest after the fire from the area before the fire. The duration of the fire is subsequently calculated with help of the fire start and the fire finish times. Using the actual time when the

system detects the fire, we can then calculate the latency of the system in detecting the fire.

For the evaluation of various scenarios of forest fires, we need to calculate the rate of spread of forest fires as given in the following equation.

$$\frac{dS_f}{dt} = \lim_{t_f \rightarrow 0} \frac{A_f}{t_f} \tag{1.2}$$

Where  $S_f$  represents the spread of fire and  $A_f$  represents the area of the fire. Thus this equation demonstrates the rate of spread of a forest fire.

*B. Simulation parameters*

A few of the baseline simulation parameters using real forest data from the Islamabad Margalla hills forests are given below in Table 1.

Table 1 System Parameters of Fire simulation model

Parameter	Default Value
Fire delay	40 hrs
Default temperature	16 degrees C
Temperature gradient	±5
Relative Humidity	37%

Here fire delay is a general parameter which allows for calibration of the simulation based on forest fire data. The temperature gradient parameter gives the variation in the temperature (i.e. if the default temperature is say set to 16°C a maximum of 21°C and a minimum of 11°C would be simulated in different sections of the simulation. Another important parameter is relative humidity. Description of other parameters and calculations are given in [5].

*C. Results and discussion*

In Figure 4, we can see the various simulated forest fires used in the system. Here we can see the relationship between the duration of fires along with the loss of tree cover

(Measured in terms of cells, which represent a unit simulation area i.e.  $km^2$ ). As can be observed from this plot, a large number of simulations were performed. Duration of simulated fires ranged from less than one hour to close to 12 simulated hours. The loss of tree cover ranged from less than  $2000 km^2$  to  $12,000 km^2$ . Here, we can see that these random fires allowed covering a large array of scenarios. In other words, if the system performs well within required parameters in these fires, there is a high probability that the system will perform quite well in actual practice as well.

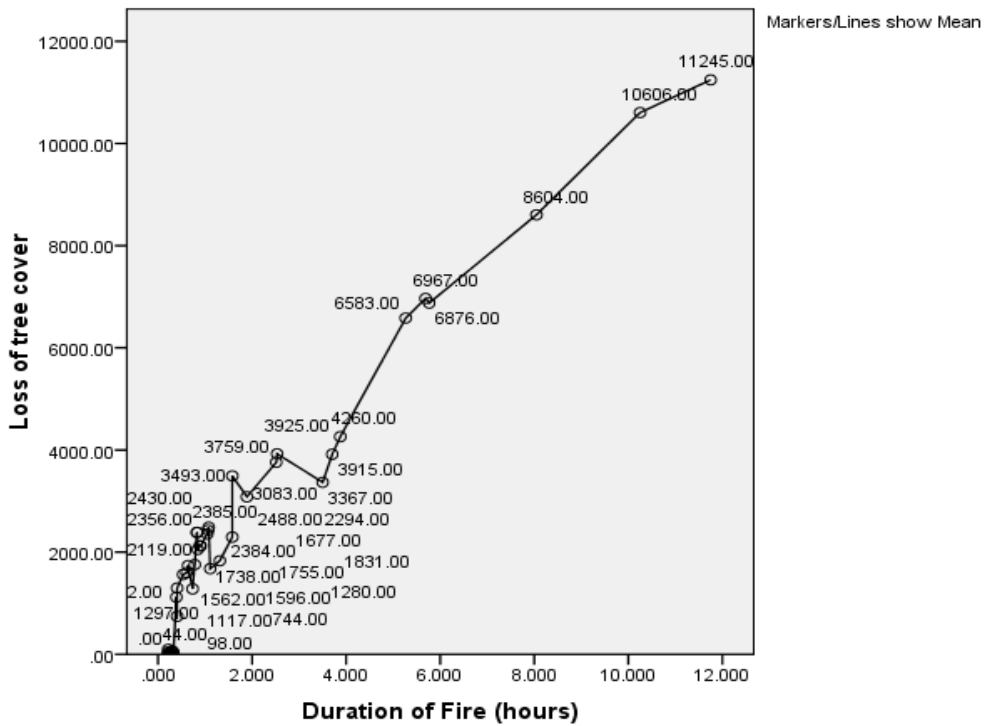


Figure 4 Plot shows the duration of various simulated fires plotted against the loss of tree cover

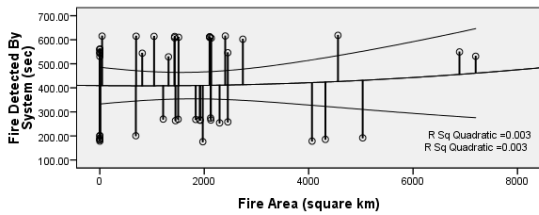


Figure 5 Latency in fire detection with a quadratic fit line plotted in 95% Confident Interval lines

In contrast to this graph, which can be considered as describing the random forest fires generated by the system for inputs to the SimConnector, Figure 5 represents the latency in the forest fires. The x-axis represents the forest fire area in square kilometer while the y-axis represents the actual latency in seconds. The plot itself is drawn in a 95% confidence interval along with quadratic fit lines. As can be seen, in general fire detection is possible in around 400 seconds up to a maximum of around 620 seconds, which amounts to around ten minutes. This is actually a very reasonable time since it is important that the forest fire should be stabilized and classified as a proper fire in contrast to perhaps a small fire which can die down on its own. If the system were to generate too many alerts, it would be completely useless since it would never be possible to find out when the actual fires were there. Domain experts confirmed that this is a reasonable time frame for the detection of forest fires.

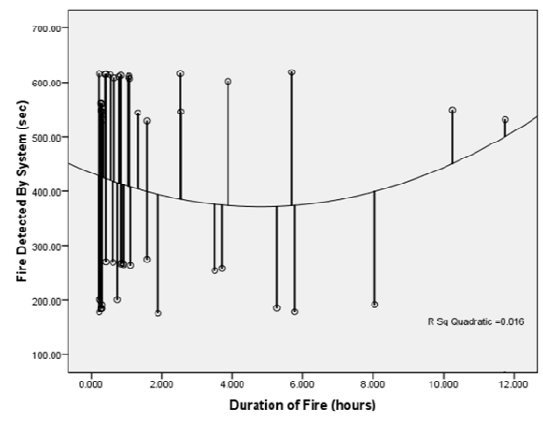


Figure 6 Fire duration vs. time taken by the system to detect and generate alarms

In Figure 6, we see a plot of the fire duration plotted against the latency of the proposed system in detecting fires. Here, we can observe that even for very small fires, the system aptly detects fires. While the fire may range from less than an hour to even 12 hours duration, the detection of the smaller fires is still efficiently performed by the system. By seeing the plot which also gives a quadratic fit line, we can observe the efficiency of the proposed system. Essentially detecting forest fires which are of shorter duration are important because personnel can be put on the lookout in the region just in case the fire goes out of hand. However, in real life, such fires are difficult to detect in large forests. The

proposed approach thus offers a better way of testing disaster alerting systems, in general, and forest fire systems in particular.

V.CONCLUSIONS AND FUTURE WORK

In this paper, we have presented the SimConnector approach to using agent-based modeling for the testing of a real-world disaster alerting system. As proof of concept, we have applied this approach to a prototype application for the detection of forest fires. The architecture of the forest fire system consists of two user interface front ends, a web service, a backend database as well as an agent-based model. We have demonstrated the application of the SimConnector approach by implementing a system using an agent-based simulation model for the validation. Using a simulated Wireless Sensor Network in a forest fire, the system demonstrates the effectiveness of the SimConnector approach. The latency of fire discovery using intelligent decision support based on the international standard Fire Weather Index is minimized using the proposed approach. In addition, extensive simulation experiments also demonstrate the effectiveness of the approach in fires of both shorter durations as well as longer durations. The key goal of the system is to provide early detection of forest fires and generate alerts for forest fighters. The application allows users to monitor and test various parameters in the forest fire model such as intensity, location, date and time of the fire whereas the web application. While in this paper, we have developed the system in the domain of forest fires, we believe the approach is equally valid for develop other types of disaster early warning and alerting systems. In the future, we plan on demonstrating the SimConnector approach in other domains of disaster alerting systems.

REFERENCES

[1]N. Kerle and C. Oppenheimer, "Satellite Remote Sensing as a Tool in Lahar Disaster Management," *Disasters*, vol. 26, pp. 140-160, 2002.  
 [2]S. B. Jb, "Mentawai tsunami death toll triples," in *The Jakarta Post*, ed. Padang, 2010.  
 [3]M. A. Niazi, *et al.*, "Verification &Validation of Agent Based Simulations using the VOMAS (Virtual Overlay Multi-agent System) approach," presented at the MAS&S 09 at Multi-Agent Logics, Languages, and Organisations Federated Workshops, Torino, Italy, 2009.  
 [4]M. Niazi, *et al.*, "Verification and Validation of an Agent-Based Forest Fire Simulation Model," presented at the SCS Spring Simulation Conference, Orlando, FL, USA, 2010.  
 [5]C. E. Van Wagner, "Development and structure of the Canadian forest fire weather index system," *Forestry technical report*, vol. 35, p. 37, 1987.  
 [6]W. H. Rutherford and J. de Boer, "The definition and classification of disasters," *Injury*, vol. 15, pp. 10-12, 1983.  
 [7]B. Boehm, "Software engineering economics," *Software Engineering, IEEE Transactions on*, pp. 4-21, 2009.  
 [8]L. Duclos, "Simulation cost model for the life-cycle of the software product: A quality assurance approach," *Dissertation Abstracts International Part B: Science and Engineering*[DISS. ABST. INT. PT. B-SCI. & ENG.], vol. 43, 1983.  
 [9]R. Axelrod, *The complexity of cooperation: agent-based models of competition and collaboration*. Princeton, NJ: Princeton University Press, 1997.

[10] M. Niazi and A. Hussain, "Agent-based Computing from Multi-agent Systems to Agent-Based Models: A Visual Survey," *Springer Scientometrics*, vol. In-press, 2011.  
 [11] R. Mukhopadhyay, *et al.*, "Promotion of variant human mammary epithelial cell outgrowth by ionizing radiation: an agent-based model supported by in vitro studies," *Breast Cancer Res*, vol. 12, p. R11, Feb 10 2010.  
 [12] D. Goulson, *et al.*, "Effects of land use at a landscape scale on bumblebee nest density and survival," *Journal of Applied Ecology*, vol. 47, pp. 1207-1215, 2010.  
 [13] V. Quera, *et al.*, "Flocking Behaviour: Agent-Based Simulation and Hierarchical Leadership," *Journal of Artificial Societies and Social Simulation*, vol. 13, p. 8, 2010.  
 [14] M. J. North and C. M. Macal, *Managing business complexity: discovering strategic solutions with agent-based modeling and simulation*: Oxford University Press, USA, 2007.  
 [15] M. A. Niazi and A. Hussain, "A Novel Agent-Based Simulation Framework for Sensing in Complex Adaptive Environments," *Sensors Journal, IEEE*, vol. 11, pp. 404-412, 2011.  
 [16] M. Niazi and A. Hussain, "Agent based tools for modeling and simulation of self-organization in peer-to-peer, ad-hoc and other complex networks," *IEEE Communications Magazine*, vol. 47(3), pp. 163 – 173, 2010.  
 [17] S. C. Bankes, "Agent-based modeling: A revolution?," vol. 99, ed: National Acad Sciences, 2002, pp. 7199-7200.  
 [18] M. A. Niazi and A. Hussain, "Sensing Emergence in Complex Systems," *IEEE Sensors Journal*, 2011.  
 [19] A. Siddiqua, *et al.*, "A new hybrid agent-based modeling & simulation decision support system for breast cancer data analysis," in *Information and Communication Technologies, 2009. ICICT '09. International Conference on*, 2009, pp. 134-139.  
 [20] S. Bandini, *et al.*, "Agent Based Modeling and Simulation: An Informatics Perspective," *Journal of Artificial Societies and Social Simulation*, vol. 12, p. 4, 10/31/ 2009.  
 [21] M. Cossentino, *et al.*, "PASSIM: a simulation-based process for the development of multi-agent systems," *International Journal of Agent-Oriented Software Engineering*, vol. 2, pp. 132-170, 2008.  
 [22] U. Wilensky, "NetLogo," *Center for Connected Learning Comp-Based Modeling, Northwestern University*, vol. Evanston, IL, 1999.  
 [23] S. M. Rinaldi, "Modeling and simulating critical infrastructures and their interdependencies," in *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, 2004, p. 8 pp.  
 [24] L. Bodrozic, *et al.*, "Agent based data collecting in a forest fire monitoring system," 2007, pp. 326-330.  
 [25] S. E. Schaeffer, *et al.*, "Decision making in distributed sensor networks," presented at the Proceedings of the Santa Fe Institute Complex Systems Summer School, Santa Fe, NM, USA, 2004.  
 [26] K. Angayarkkani and N. Radhakrishnan, "Efficient Forest Fire Detection System: A Spatial Data Mining and Image Processing Based Approach," *IJCSNS*, vol. 9, p. 100, 2009.  
 [27] D. Torii, *et al.*, "Layering social interaction scenarios on environmental simulation," *Multi-Agent and Multi-Agent-Based Simulation*, pp. 78-88, 2005.  
 [28] S. Hochrainer, "Catastrophe modeling and simulation," in *Macroeconomic Risk Management Against Natural Disasters*, ed: DUV, 2006, pp. 105-144.  
 [29] M. Cossentino, *et al.*, "Simulation-based design and evaluation of multi-agent systems," *Simulation Modelling Practice and Theory*, vol. 18, pp. 1425-1427, 2010.  
 [30] A. Mata, *et al.*, "Forest Fire Evolution Prediction Using a Hybrid Intelligent System," in *Balanced Automation Systems for Future Manufacturing Networks*. vol. 322, Á. Ortiz, *et al.*, Eds., ed: Springer Boston, 2010, pp. 64-71.  
 [31] U. Wilensky, "NetLogo Fire model," *ed. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University*, 1997.  
 [32] M. Hafeeda and M. Bagheri, "Wireless sensor Network for early detection of forest fire," presented at the In Proc of the International Conferences of Mobile Adhoc and Sensor System, 2007.



# A Chemical Inspired Simulation Framework for Pervasive Services Ecosystems

Danilo Pianini  
 DEIS–Università di Bologna  
 via Venezia 52, 47521 Cesena, Italy  
 Email: danilo.pianini@unibo.it

Sara Montagna  
 DEIS–Università di Bologna  
 via Venezia 52, 47521 Cesena, Italy  
 Email: sara.montagna@unibo.it

Mirko Viroli  
 DEIS–Università di Bologna  
 via Venezia 52, 47521 Cesena, Italy  
 Email: mirko.viroli@unibo.it

**Abstract**—This paper grounds on the SAPERE project (Self-Aware PERvasive Service Ecosystems), which aims at proposing a multi-agent framework for pervasive computing, based on the idea of making each agent (service, device, human) manifest its existence in the ecosystem by a *Live Semantic Annotation (LSA)*, and of coordinating agent activities by a small and fixed set of so-called *eco-laws*, which are sort of chemical-like reactions evolving the distributed population of LSAs. System dynamics in SAPERE is complex because of openness and due to the self-\* requirements imposed by the pervasive computing setting: a simulation framework is hence needed for what-if analysis prior to deployment. In this paper we present a prototype simulator which – due to the role of chemical-like character of *eco-laws* – is based on a variation of an existing SSA (Stochastic Simulation Algorithm), tailored to the specific features of SAPERE, including dynamicity of network topology and pattern-based application of *eco-laws*. The simulator is tested on a crowd steering scenario where groups are guided, through public or private displays, towards the preferential destination and by emergently circumventing crowded regions.

## I. INTRODUCTION AND MOTIVATION

**T**HE INCREASING evolution of pervasive computing is promoting the emergence of decentralised and complex infrastructures for pervasive services composed by new communication devices (e.g. mobile phones, PDA's, smart sensors, laptops). Such infrastructures include traditional services with dynamic and autonomous context adaptation (e.g., public displays showing information tailored to bystanders), as well as innovative services for better interacting with the physical world (e.g., people coordinating through their PDAs). Mainstream languages and software infrastructures are often inadequate to face requirements of scalability, openness, adaptivity and self-organisation typical of pervasive systems. In order to better handle these scenarios, a paradigm shift towards agent world is receiving more and more attention in the scientific community. Agents support the implementation of distributed and communicating environments where different kind of autonomous entities (i.e. agents) are located. In this context, one of the hottest research topics regards agent coordination, namely the way an infrastructure can be built to allow agents to produce, consume and exchange information inside the pervasive system [31].

Different approaches were proposed in the area of coordination models and middlewares for pervasive computing scenarios: they try to account for issues related to spatiality

[17], [21], spontaneous and opportunistic coordination [2], [10], self-adaptation and self-management [25]; however, they typically propose ad-hoc solutions to specific problems in specific areas, and lack generality.

The SAPERE project (“Self-adaptive Pervasive Service Ecosystems”) addresses the issues above in a uniform way by means of a self-adaptive pervasive substrate, namely, a space bringing to life an ecosystem of individuals, which are pervasive services and devices able to interact with humans. The key idea is to coordinate agents in a self-organising way by basic laws (called *eco-laws*) that evolve the population of individuals in the system, enacting mechanisms of coordination, communication, and interaction [32]. Technically, such *eco-laws* are structured as sort of chemical reactions, working on the “interface annotation” that each agent injects in neighbouring localities, called *LSA* (Live Semantic Annotation).

In this context, simulation can be useful in supporting the design of *eco-laws* and agent behaviour, and ultimately, of whole pervasive service ecosystems. They give the possibility to experiment the idea of exploiting ecological mechanisms, such as those inspired by biology [9], showing through simulation the overall behaviour of a system designed on top of *eco-laws*, as well as to elaborate what-if scenarios. Moreover, a well designed framework will enable researchers to formally analyse the properties of such pervasive systems through stochastic model checking [13].

To capture the whole complexity of the SAPERE approach the model has to support in a coherent model the following abstractions: (i) highly dynamic environments composed of different, mobile, communicating nodes; (ii) reactive behaviours expressed by chemical-like reactions over LSAs; and (iii) autonomous behaviour of agents.

On one hand the adoption of standard Agent-Based Models (ABM) [16] seems to be quite natural, since the pervasive system itself is engineered adopting the agent paradigm and relying on a mediated form of interaction—as typical when using, e.g., the A&A metamodel [24]. There are several works which apply this approach in different contexts, from social systems (see, e.g., [3]) to biological systems [19], [4]. An ABM grounds around autonomous and possibly heterogeneous agents that can be situated in an environment. They carry out the most appropriate line of action, possibly interacting with other agents as well as the environment itself. The agent

behaviour is modelled through a set of rules which describe how the agent behaves according to environmental conditions. These rules can be of different types, according to the specific model / architecture: from simple reactive rules specifying how the agent must react to environmental stimuli or perceptions, to pro-active ones specifying how the agent must behave with respect to its goals and tasks [33]. In ABM the environment is also a first class abstraction whose structure, topology and dynamic can be explicitly modelled. To develop and simulate ABMs different simulation frameworks have been developed, such as MASON [15], Repast [22], NetLogo [26] and Swarm [28].

On the other hand, ABMs do not typically provide tools for sophisticated design of behavioural rules in the environment, at least up to the point of supporting an efficient simulation of chemical-like reactions as studied in [12] and its extensions. Examples of works going in this direction escape the field of ABMs, entering the scope of formal models for stochastic and bio-inspired concurrency models, namely, based on Stochastic Simulation Algorithms (SSA) such as BioPEPA [7] and BetaWB [8]. However, in this field few simulators allow to flexibly define network topologies [1] – they mostly deal with a single or few chemical compartments – and to the best of our knowledge no one provides features of network mobility and fine-tuning control over different behaviour in different nodes, for this typically escapes the context of biological systems. Additionally, SAPERE eco-laws do not fit exactly chemical reactions, for they handle structured molecules and advanced matching algorithms in a way that existing chemical simulators can hardly tackle.

To take the best of both approaches we developed a brand new simulation framework, called ALCHEMIST, meant to face natively the above requirements. It implements an optimised version of the Gillespie's SSA, namely the Next Reaction Method [11], extended with the possibility to have dynamic reactions, *i.e.* reactions that can be added or removed once the simulation runs due to network mobility, and adapted to the semantics of the eco-law language.

We exemplify the approach in a case study of crowd steering in pervasive computing, in which groups are guided towards locations based on their preference, along optimal paths and taking into account the presence of crowded regions which should be circumvented. We provide the set of eco-laws solving the problem (which adopts mechanisms proposed in the context of computational fields and spatial computing [17], [20]) and validate it via simulation of the associated stochastic model.

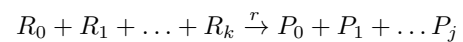
The remainder of this paper is organised as follows: Section II presents details about the computational model we defined and the simulation engine, Section III reports the model application at the crowd steering scenario and the simulation results and Section IV provides concluding remarks and discusses future works.

## II. ENGINE ARCHITECTURE

In this section we first introduce how to model a chemical system in both deterministic and stochastic ways, then we show the known algorithms for stochastic simulation and our choices for a full featured high performance engine.

### A. Stochastic Simulation Algorithms

A chemical system can be modelled as a single space filled with molecules that may interact through a number of reactions describing how they combine. The instantaneous speed of a reaction is called propensity and depends on the kinetic rate of the reaction and on the concentrations of all the reagents involved. For a reaction  $i$  with  $k$  reactants,  $j$  products, rate  $r$  of the form



the propensity  $a_i$  is defined as:

$$a_i = r \cdot [R_0] \cdot [R_1] \cdot \dots \cdot [R_k]$$

where  $[R_a]$  is the number of molecules of species  $R_a$ .

Since classical ODE (Ordinary Differential Equation) models are not accurate when the number of molecules in the system is low, a stochastic model has been proposed in [12]. This kind of description considers the whole system as a CTMC (Continuous-Time Markov Chain), in which the rate of the transition representing the  $i$ -th reaction is the propensity function  $a_i$ —and represents the average frequency at which the transition should be scheduled, following a negative exponential distribution of probability.

In [12], two algorithms are proposed in order to correctly simulate a stochastic path of a chemical system. Those algorithms were successively improved, but even their optimized versions, they rely on the idea that the system can be simulated by effectively executing the reactions one by one and changing the system status accordingly. Every algorithm follows four main steps:

- 1) select the next reaction  $\mu$  to be executed;
- 2) calculate the time of occurrence of  $\mu$  according to an exponential time distribution and make it the current simulation time;
- 3) change the environment status in order to reflect this execution;
- 4) update the propensities of the reactions.

The known techniques differ in the implementation of first and fourth steps. We will briefly present them and then justify our choice for the engine.

1) *Direct Method:* The direct method was first proposed in [12]. It chooses the next reaction to be executed by throwing a random number  $r \leq \sum_i a_i$  and selecting the first reaction  $\mu$  which verifies the property that  $r \geq \sum_{i=0}^{\mu} a_i$ . After the execution of  $\mu$ , it updates propensities for each reaction.



2) *Optimized Direct Method*: The direct method can be optimised as proposed in [11] and [29] by introducing a binary search tree and a dependency graph. The former allows one to choose the next reaction  $\mu$  to be executed in logarithmic time, the latter to update only the propensities of those reactions in which concentration of reagents is modified by the execution of  $\mu$ .

3) *Composition-Rejection Method*: In [27] a constant time method relying on composition-rejection algorithm is proposed. The separation between the number of reactions  $R$  and the computational complexity of the algorithm is obtained by splitting the whole set of reactions into  $G$  groups, and then arguing that  $G$  does not depend (or depends loosely) by  $R$ . It may rely on a dependency graph in order to improve the update phase.

4) *First Reaction*: The First Reaction Method is the dual form of the Direct Method, and was proposed first in [12]. The key idea is to calculate immediately the time of occurrence for each reaction and select the next one to be executed using the lowest time. It is demonstrably the same of the Direct Method both in soundness and in time complexity.

5) *Next Reaction*: The Next Reaction Method is an optimised form of the First Reaction Method first proposed in [11]. It relies on an Indexed Priority Queue (IPQ) in order to smartly sort the reactions by time and has constant time in the selection phase since the root of the IPQ is always the next reaction to execute. A dependency graph can be used in order to update only the required propensities, and moreover a random-number re-usage is allowed, speeding up consistently the times recalculation.

## B. Computational Model

Before discussing our choice of basic SSA algorithm, we describe the computational model we propose in order to fill the gap between the SAPERE world and chemical simulation. In fact, requirements on the model will necessarily influence some aspects of the design choices behind the engine itself.

First of all, we want to motivate the choice of chemical-resembling laws to model self-\* behaviours. It has been proved that most ODE equations describing population dynamics can be translated in an equivalent CTMC passing through a set of chemical like reactions that describes the way the population entities interact [6]. This expressive power of chemical reactions is very interesting as soon as it allows us to use them in the design of pervasive systems inspired at ecological systems.

Our model extends the classic model of chemical reactions in three main directions.

First, in the classic chemical formulation [12], the environment is a single compartment that contains the molecules soup. This description is pretty far from the world we want to model, which is a pervasive service ecosystem. The natural extension is to consider many compartments (nodes) placed in a space (environment) which is responsible of linking them based on some rule. Depending on the specific environment, nodes can be dynamically added, moved or removed. A neighbourhood

is consequently a structure which contains a node “centre” and a list of all linked compartments.

Second, in classical chemical models, a reaction lists a number of reactant molecules which, combined, produce a set of product molecules. This kind of description is too strict for our purposes. A more generic concept is to consider a reaction as a set of conditions about the environment which, when matched, may allow the execution of a set of actions. A condition is a function which associates a boolean to the current state of the environment, an action is a procedure which modifies it. The propensity function can no longer be simply the product of the reaction rate with the concentrations of the reactants, but needs a more generic definition too: propensity in our model is a function of the reaction rate, the conditions, and the environment state.

Third, we want to deal with events whose occurrence time does not follow an exponential law, like triggers events happening at a specific time regardless the previous evolution of the system—we want to occasionally depart the CTMC model for the sake of flexible configuration of simulations. For instance, we may want to simulate an alarm event at a specific simulation time, such that one can run multiple simulations in order to understand how the system will react. Another usage of triggers appears when considering the possibility to interact with a running simulation pausing it and, exploiting triggers, interact with the environment in its current status, then resume the simulation. Even if this approach is not useful when the goal is to check the properties of a model, it could be very handy when exploring and testing it.

## C. Dynamic engine

Given the model we want to simulate described in Section II-B and the algorithms presented in Section II-A, we can argue that no existing algorithm as-is is appropriate to support our simulations. In particular, no algorithm provides facilities to add and remove reactions dynamically and to inject triggers and other non-exponential time distributed events. Our choice for the engine algorithm to extend was then restricted between the First Reaction and the Next Reaction, because they are the only that choose the next reaction to execute explicitly considering the time of occurrence, which makes a lot simpler to support non-exponential time distributed events. The latter is an optimization of the former, offers a lower computational complexity in every case and consequently can achieve higher performance. Our work had the primary goal to extend Next Reaction providing the possibility to add and remove reactions dynamically, since to the best of our knowledge no work in this sense have been ever made. In order to add this support, it is a mandatory task to provide methods to add and remove reactions from the indexed priority queue and the dependency graph.

1) *Dynamic Indexed Priority Queue*: A key property of the original Indexed Priority Queue proposed in [11] is that the swap procedure used to update the data structure does not change the balance of the tree, ensuring optimal update times in every situation. This feature was easily achieved because no

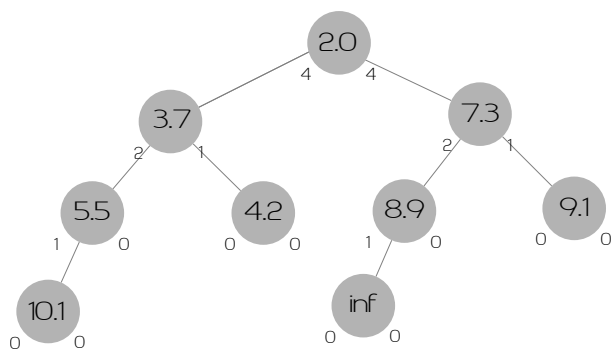


Fig. 1. Indexed Priority Queue extended with children count per branch

nodes were ever added neither removed from the structure. As a consequence, once the tree is balanced at creation time no event can occur to change its topology. This is no longer the case, and we have to provide a small extension to the structure in order to manage the balancing. Our idea is, for each node, to keep track of the number of children per branch, having in such way the possibility to keep the tree balanced when adding nodes. In figure 1 we show how the same IPQ drawn in [11] would appear with our extension. In the following algorithms, the procedure `UPDATE_AUX(n)` is the same described in [11]. Given this data structure, the procedure to add a new node  $n$  is the following:

```

IF root does not exist
  n is the new root
ELSE
  c ← root
  WHILE c has two children
    IF c.right < c.left
      dir ← right
    ELSE
      dir ← left
    add 1 to count of dir children
    c ← c.dir
  IF c has no left child
    n becomes left child of c
    set count of left nodes of c to 1
  ELSE
    n is right child of c
    set count of right nodes of c to 1
UPDATE_AUX(n)

```

The removal procedure for a node  $n$  is the following:

```

c ← root
WHILE c is not a leaf
  IF c.left > c.right
    dir ← left
  ELSE
    dir ← right
  subtract 1 to count of dir children
  c ← c.dir
IF c != n
  swap c and n
  remove n
  UPDATE_AUX(c)
ELSE
  remove n

```

Using the two procedures described above, the topology of the whole tree is constrained to remain balanced despite the dynamic addition and removal of reactions.

2) *Dynamic Dependency Graph*: Since we want to support natively and efficiently the dependencies among multiple

compartments, we defined three contexts (also called scopes): local, neighborhood and global. Each reaction has an input context and an output context, meaning respectively where data influencing the rate calculus is located and where the modifications are made.

The first issue to address is to evaluate if a reaction  $r_1$  may influence another reaction  $r_2$ , considering their contexts. We introduced a boolean procedure `mayInfluence(r1, r2)` which operates on two reactions and returns a true value if:

- $r_1$  and  $r_2$  are on the same node OR
- $r_1$ 's output context is global OR
- $r_2$ 's input context is global OR
- $r_1$ 's output context is neighborhood and  $r_2$ 's node is in  $r_1$ 's node neighbourhood OR
- $r_2$ 's input context is neighborhood and  $r_1$ 's node is in  $r_2$ 's node neighbourhood OR
- $r_1$ 's output context and  $r_2$ 's input context are both neighborhood and the neighbourhoods of their nodes have at least one common node.

Given this handy function, we can assert that a dependency exists between the execution of a reaction  $r_1$  and another reaction  $r_2$  if `mayInfluence(r1, r2)` is true and at least a molecule whose concentration is modified by  $r_1$  is among those influencing  $r_2$ .

Adding a new reaction implies to verify its dependencies against every reaction of the system. In case there is a dependency, it must be added to the graph. Removing a reaction  $r$  requires to delete all dependencies in which  $r$  is involved both as influencing and influenced. Moreover, in case of change of the system topology, a dependencies check among reactions belonging to nodes with modified neighbourhood is needed. It can be performed by scanning them, calculating the dependencies with the reactions belonging to new neighbours and deleting those with nodes which are no longer in neighbourhood.

#### D. Engine architecture

The whole framework has been designed to be fully modular and extensible. The whole engine or parts of it can be re-implemented without touching anything in the model, and on the other hand the model can be extended and modified without messing with the engine. This modularity allows to easily make some experiments with other engines, such as Composition-Rejection.

The framework, called *ALCHEMIST*, was developed from scratch using Java. Being performances a critical issue for a simulator, we compared some common languages in order to evaluate their performance level. Surprisingly, Java performance are at same level of compiled languages such as C/C++ [5], [23]. The Java language was consequently chosen because of the excellent trade off among performances, easy portability and maintainability of the code, plus the support for concurrent programming at language level. The *COLT* Java library [14] provided us the mathematical functions we needed. In particular, it offers a fast and reliable random number generation algorithm, the so called Mersenne Twister [18].

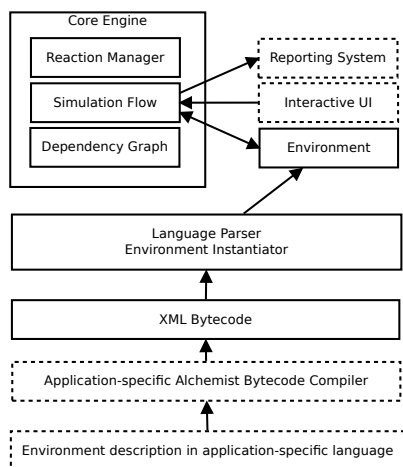


Fig. 2. ALCHEMIST architecture. Elements drawn with continuous lines indicates components common for every scenario and already developed, those with dotted lines are extension-specific components which have to be developed with the specific application in mind.

ALCHEMIST is still actively developed and currently consists of 188 classes for a total of 16527 lines of code.

As shown in Figure 2, at the current status of development the simulations are written in a specific XML language containing a complete description of environment and reactions. This code is interpreted in order to produce an instance of an environment, once it is created, no further interpretation is needed in order to run the simulation. This XML code is not meant to be directly exploited by users, but it represents a way to describe environments in a machine-friendly way and is a formalisation of the generic model of ALCHEMIST. The idea behind this choice is that ALCHEMIST is flexible enough to be used in various contexts, each one requiring a personalised language and a different instantiation of the model. It's up to the extensor to write a translation module from its personalised language to the ALCHEMIST XML.

### III. CASE STUDY

We propose a crowd steering scenario as a case study to demonstrate the possibility to exploit eco-laws to lead people in the desired location within a complex environment in short time, avoiding obstacles such as crowded regions and without global supervision.

Consider a museum with a set of rooms, whose floor is covered with a network of computational devices (infrastructure nodes). These devices can exchange information with each other based on proximity, sense the presence of visitors, and hold information about expositions currently active in the museum. Each room has four exits and they are connected via external corridors. Visitors wandering the museum are equipped with a hand-held device that holds the visitor's preferences. By interaction with infrastructure nodes, a visitor can be guided towards rooms with a target matching their interest, thanks to signs dynamically appearing on his smartphone. This is done, using techniques suggested in the field of spatial

computing [30]—namely, computational gradients injected in a source and diffusing around such that each node holds the minimum distance from source.

#### A. A SAPERE model

The environment is made of infrastructure nodes. Smartphones are agents dynamically linked with the nearest sensors – the neighbours are the sensors inside a certain radius  $r$ , parameter of the model – from which they can retrieve data in order to suggest visitors where to go. Visitors are agents which tend to follow the advices of their hand-held device. They can move of discrete steps inside the environment. It is also defined a minimum possible distance between them, so to model the physical limit and the fact that two visitors can't be in the same place at the same time.

In the SAPERE approach, all the information exchanged is in form of LSAs, and the rules are expressed in form of eco-laws. Although in the SAPERE framework LSAs are *semantic* annotations, expressing information with same expressiveness of standard frameworks like RDF, we here consider a simplified notation. Namely, an LSA is simply modelled as a tuple  $\langle v_1, \dots, v_n \rangle$  (ordered sequence) of typed values, which could be for example numbers, strings, structured types, or function names.

There are three forms of LSAs used in this scenario:

$$\begin{aligned} &\langle \text{source}, id, type, N_{max}, \pi, \mu, type' \rangle \\ &\langle \text{field}, id, type, value, \pi, \mu, type', tstamp \rangle \\ &\langle \text{pre\_field}, id, type, value, \pi, \mu, type', tstamp \rangle \end{aligned}$$

A **source** LSA is used as a source with the goal of generating a field:  $id$  labels the source so as to distinguish sources of the same type;  $type$  indicates the type of fields (`target` is used to advertise expositions, and `crowd` to diffuse information about crowding);  $N_{max}$  is the field's maximum value;  $\pi$  and  $\mu$  are two functions used respectively to compute the new field value once it has to be propagated or transformed according to the value of another field of type  $type'$ —their purpose will be described more in details later, along with eco-laws. A **field** LSA is used for individual values in a gradient:  $value$  indicates the individual value; the  $tstamp$  reflects the time of creation of the LSA; the other parameters are like in the source LSAs. A **pre\_field** LSA is used to diffuse the field before it is influenced by the transformation rule.

An eco-law is a chemical-resembling reaction working over patterns of LSAs. One such pattern  $P$  is basically an LSA which may have some variable in place of one or more arguments of a tuple, and an LSA  $L$  is matched to the pattern  $P$  if there exists a substitution of variables which applied to  $P$  gives  $L$ . An eco-law is hence of the kind  $P_1, \dots, P_n \xrightarrow{r} P'_1, \dots, P'_m$ , where: (i) the left-hand side (reagents) specifies patterns that should match LSAs  $L_1, \dots, L_n$  to be extracted from the LSA-space; (ii) the right-hand side (products) specifies patterns of LSAs which are accordingly to be inserted back in the LSA-space (after applying substitutions found when extracting reagents, as in standard logic-based rule approaches); and (iii)

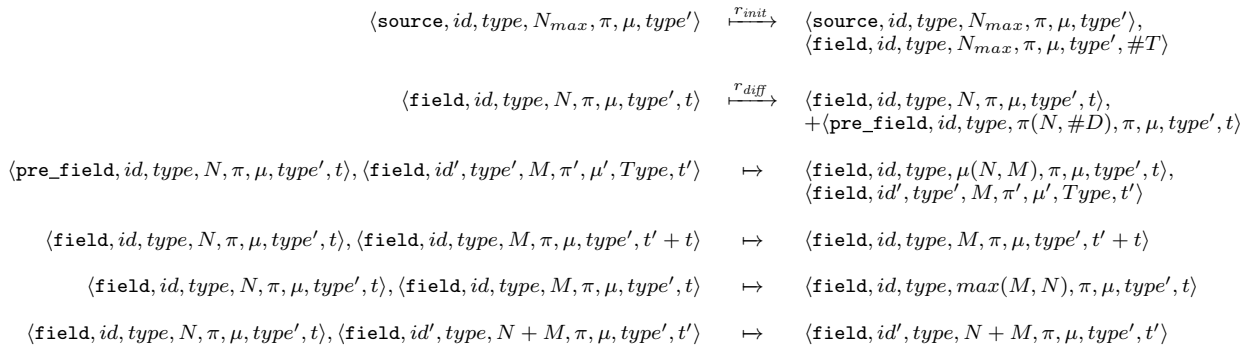


Fig. 3. Eco laws describing the museum application.

rate  $r$  is a numerical positive value indicating the average frequency at which the eco-law is to be fired—namely, we model execution of the eco-law as a CTMC transition with Markovian rate (average frequency)  $r$ . If no rate is given the reaction is meant to be executed “as soon as possible”, which means that the rate that associated with the reaction tends to infinite. To allow interaction between different LSA-spaces, we introduce the concept of *remote pattern*, written  $+P$ , which is a pattern that will be matched with an LSA occurring in a neighbouring LSA-space. In Figure 3, the eco-laws for our case study are given.

As sources LSAs are injected in nodes, gradients are built by the first three rules in Figure 3. The first eco-law, given a source, initiates the field with its possible maximum value. The second eco-law, when a node contains a field LSA, spreads a *pre\_field* LSA to a neighbouring node picked up randomly with a new value computed according to the propagation function,  $\pi$ , which elaborates the distance between sensors – indicated by the variable  $\#D$  – and the actual value of the field LSA. The third eco-law, when a node contains a *pre\_field* LSA of type  $type$  and a field LSA of type  $type'$  by which the *pre\_field* depends, removes the *pre\_field* LSA and creates a new field LSA with a value computed according to the transformation function,  $\mu$ , which elaborates the values  $N$  and  $M$  of the two reactants. The purpose of this law is to model the interactions between fields. For instance we may assume that if there is a crowd which jams a region of the museum, that path towards the target should have less probability of being picked, so the value of the target field has to be reduced. As a consequence of these laws, each node will carry a field LSA indicating the topological distance from the source. The closest is the field value to  $N_{max}$ , the nearest is the field source. When the spread values reach the minimum value 0, the gradient has to become a plateau.

To address the dynamism of the scenario where people move, targets being possibly shifted, and crowds forming and dissolving, we introduced the following mechanism. We expect that if a gradient source moves the diffused value has to change according to the new position. This is the purpose

of the *tstamp* parameter which is used in the fourth eco-law, continuously updating old values by more recent ones (*youngest* eco-law). In this way we ensure that the system is able to adapt to changes of the source states. Finally, the spreading eco-law above may produce duplicate values in locations, due to multiple sources of the same type (indicated by different ids), multiple paths to a source, or even diffusion of multiple LSAs over time. For this reason we introduced the last two eco-laws. They retain only the maximum value, i.e. the minimum distance, the former when there are two identical LSAs with only a different value, the latter when the id is different (*shortest* eco-laws).

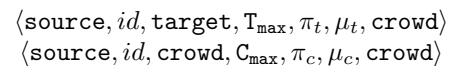
Eco-laws in Figure 3 describe the behaviour of the museum ecosystem in a chemical-oriented fashion, and are accordingly modelled in the simulator as reactions.

People are modelled as agents. They move according to the field value computed for their target probabilistically choosing the neighbour with higher field value. The behaviour of visitors is modelled as a reaction too, featuring a special action in which the behaviour of the visitors is expressed.

The proposed architecture is intrinsically able to dynamically adapt to unexpected events (like node failures, network isolation, crowd formation, and so on) while maintaining its functionality.

### B. Simulator configuration and results

The behaviour of each node is programmed according to the eco-laws coordination model explained in Figure 3. Each node in the environment contains by default, for each type in the system – *target* and *crowd* –, an LSA of the form  $\langle \text{field}, id, type, 0, \pi, \mu, type', 0 \rangle$ . The sources of the gradients are injected by sensors when a target or a number of persons is perceived, with the values



For the first kind of source we assume that we can have different targets according to different preferences of users. For the crowding source instead, we may assume that sensors are calibrated so as to locally inject an LSA indicating the level

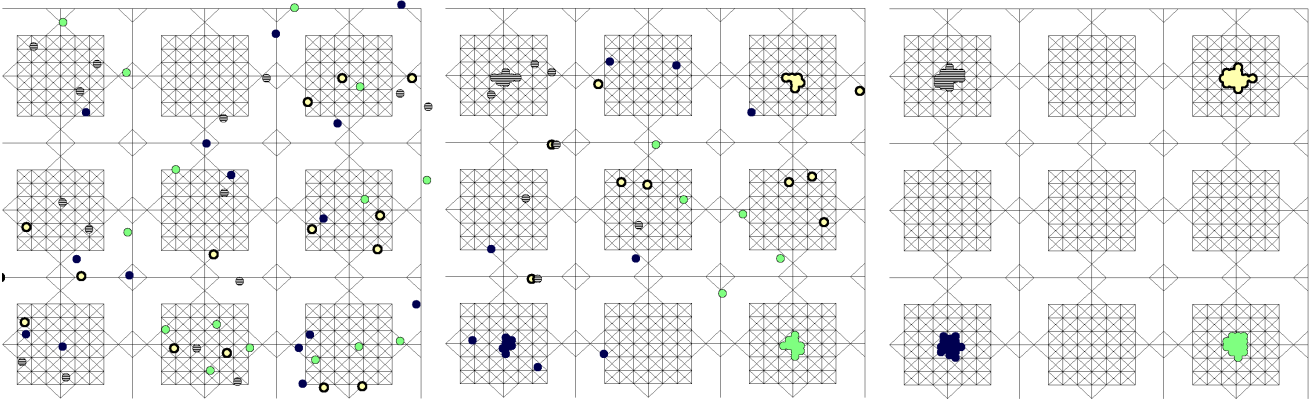


Fig. 4. A simulation run of the reference exposition: three snapshots of the ALCHEMIST graphic reporting module with this simulation

of crowding, *i.e.* if the number of persons, and is periodically updated by the sensors.

The propagation and transformation functions have the following form:

$$\begin{aligned}\pi_t &= \pi_c = N_{t,c} - \#D \\ \mu_t &= N_t - k * N_c \\ \mu_c &= N_c\end{aligned}$$

where  $N_{t,c}$  are the actual values of the two types of fields, and  $k$  is a model parameter use to modulate the effect that the crowd can have on the target field. Other parameters are:  $T_{max} = 1000$ ,  $C_{max} = 1$ . When  $\mu_t, \pi_{t,c} < 0$ , we impose them to be zero.

The reaction rates are identified by hand performing different simulations with different parameters. The results reported below are obtained with  $r_{init} = 1$  and  $r_{diff} = 50$ . The other laws show no rate because it is assumed to be infinite.

We here present simulations conducted over an exposition, where nine rooms are connected via corridors. People can express different preferences represented by their colour.

Four snapshots of a first simulation run are reported in Figure 4. We here consider four different targets that are located in the four rooms near environment angles. People are initially spread randomly in the museum, as shown in the first snapshot, and they eventually reach the room in which the desired target is hosted, as shown in the last snapshot.

Figure 5 shows a simulation experimenting with the effect of crowding in the movement of people. Two groups of people – denoted with empty and full circles – with common interests are initially located in two different rooms, as shown in the first snapshot. The target for the dark visitors is located in the central room of the second row, while the others’ is in the right room of the second row. In the simulation, dark visitors reach their target soon before it is nearer, though forming a crowded area intersecting the shortest path towards the target for the other visitors. Due to this jam the latter visitors choose a different path that is longer but less crowded.

#### IV. CONCLUSION

In the SAPERE metaphor, the ideal level of abstraction to reach in order to easily and correctly model and simulate pervasive systems requires both the rich environment of ABM and the native CTMC model support of biochemistry-oriented stochastic simulators. In this work we shown the ALCHEMIST simulation framework, meant to fully support this way to think pervasive systems. This framework embraces the SAPERE vision and allows to approach the simulation of agent systems in a new flavour, describing the system in terms of reaction-like laws and having consequently the possibility to rely on all the work already made about CTMC. We shown a case study whose complexity overcomes the expressiveness possibility of classical biochemistry-oriented simulation frameworks, and we analysed it exploiting the same CTMC mathematical support. Perspectives for the immediate future include a deeper analysis of performance for the proposed case study, tuning parameters so as to identify the most proper extent to which a crowd should influence movement of people. Then we mean to compare performance and expressiveness with respect to ABM simulation frameworks, such as Repast and NetLogo. Finally, we shall analyse, model and simulate further scenarios, with different types of complexity so as to stress the potentialities of ALCHEMIST.

#### ACKNOWLEDGMENT

This work has been supported by the EU-FP7-FET Proactive project SAPERE Self-aware Pervasive Service Ecosystems, under contract no.256873

#### REFERENCES

- [1] R. Alves, F. Antunes, and A. Salvador. Tools for kinetic modeling of biochemical networks. *Nature Biotechnology*, 24(6):667–672, June 2006.
- [2] M. Autili, P. Benedetto, and P. Inverardi. Context-aware adaptive services: The plastic approach. In *FASE '09 Proceedings*, pages 124–139, Berlin, Heidelberg, 2009. Springer-Verlag.
- [3] S. Bandini, S. Manzoni, and G. Vizzari. Crowd Behavior Modeling: From Cellular Automata to Multi-Agent Systems. In A. M. Uhrmacher and D. Weyns, editors, *Multi-Agent Systems: Simulation and Applications*, Computational Analysis, Synthesis, and Design of Dynamic Systems, chapter 13, pages 389–418. CRC Press, June 2009.

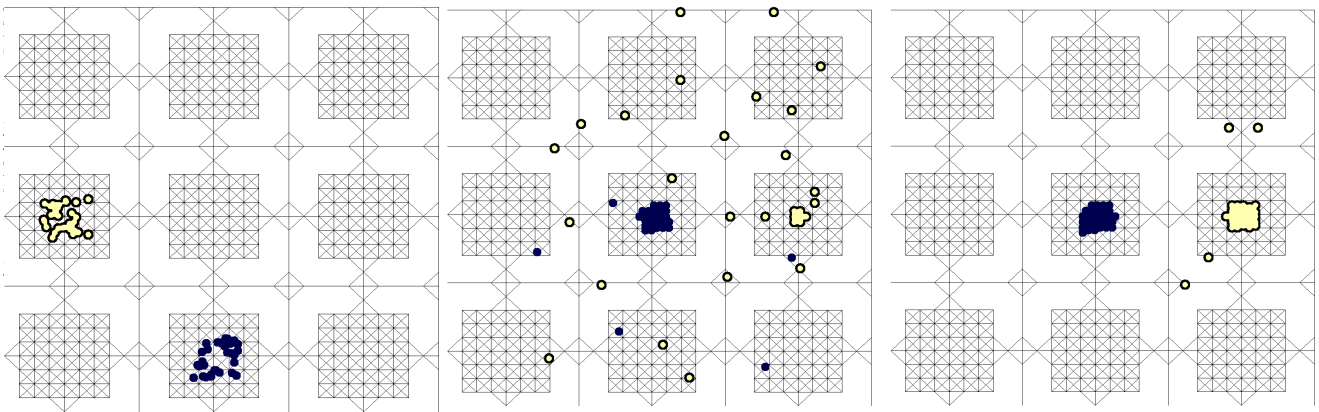


Fig. 5. A run showing the effect of crowding: dark visitors occupy a central room, making other visitors moving left to right by a longer, less crowded path

- [4] G. Beurier, F. Michel, and J. Ferber. A morphogenesis model for multiagent embryogeny. In L. M. Rocha, L. S. Yaeger, M. A. Bedau, D. Floreano, R. L. Goldstone, and A. Vespignani, editors, *Artificial Life X*, pages 84–90. MIT Press, Cambridge, MA, 2006.
- [5] J. M. Bull, L. A. Smith, C. Ball, L. Pottage, and R. Freeman. Benchmarking java against c and fortran for scientific applications. *Concurrency and Computation: Practice and Experience*, 15(3-5):417–430, 2003.
- [6] L. Cardelli. From processes to odes by chemistry. In *IFIP TCS*, pages 261–281, 2008.
- [7] F. Ciocchetta and M. L. Guerriero. Modelling biological compartments in Bio-PEPA. *Electronic Notes in Theoretical Computer Science*, 227:77–95, 2009.
- [8] L. Dematté, C. Priami, A. Romanel, and O. Soyer. Evolving blenx programs to simulate the evolution of biological networks. *Theoretical Computer Science*, 408(1):83–96, 2008.
- [9] J. L. Fernandez-Marquez, J. L. Arcos, G. Di Marzo Serugendo, M. Viroli, and S. Montagna. Description and composition of bio-inspired design patterns: the gradient case. In *Proceedings of the 3rd Workshop on Bio-Inspired and Self-\* Algorithms for Distributed Systems*, Karlsruhe, Germany, 14 June 2011. ACM.
- [10] C.-L. Fok, G.-C. Roman, and C. Lu. Enhanced coordination in sensor networks through flexible service provisioning. In J. Field and V. T. Vasconcelos, editors, *Proceedings of COORDINATION 2009*, volume 5521 of *LNCS*, pages 66–85. Springer-Verlag, 2009.
- [11] M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A*, 104:1876–1889, 2000.
- [12] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, December 1977.
- [13] T. Héroult, R. Lassaigne, F. Magniette, and S. Peyronnet. Approximate probabilistic model checking. In B. Steffen and G. Levi, editors, *Proc. 5th International Conference on Verification, Model Checking and Abstract Interpretation (VMCAI'04)*, volume 2937 of *Lecture Notes in Computer Science*, pages 73–84. Springer, 2004.
- [14] W. Hoschek. *The Colt Distribution: Open Source Libraries for High Performance Scientific and Technical Computing in Java*. CERN, Geneva, 2004. Available at <http://acs.lbl.gov/software/colt/>.
- [15] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. C. Balan. Mason: A multiagent simulation environment. *Simulation*, 81(7):517–527, 2005.
- [16] C. M. Macal and M. J. North. Tutorial on agent-based modelling and simulation. *Journal of Simulation*, 4:151–162, 2010.
- [17] M. Mamei and F. Zambonelli. Programming pervasive and mobile computing applications: The tota approach. *ACM Trans. Softw. Eng. Methodol.*, 18(4):1–56, 2009.
- [18] M. Matsumoto and T. Nishimura. Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Model. Comput. Simul.*, 8(1):3–30, 1998.
- [19] S. Montagna, N. Donati, and A. Omicini. An agent-based model for the pattern formation in *Drosophila Melanogaster*. In H. Fellermann, M. Dör, M. M. Hanczyc, L. Ladegaard Laursen, S. Maurer, D. Merkle, P.-A. Monnard, K. Stoy, and S. Rasmussen, editors, *Artificial Life XII*, chapter 21, pages 110–117. The MIT Press, Cambridge, MA, USA, 2010.
- [20] S. Montagna, M. Viroli, M. Risoldi, D. Pianini, and G. Di Marzo Serugendo. Self-organising pervasive ecosystems: A crowd evacuation example. In *Proceedings of the 3rd International Workshop on Software Engineering for Resilient Systems*, Lecture Notes in Computer Science, Geneva, Switzerland, 2011. Springer. In Press.
- [21] A. L. Murphy, G. P. Picco, and G.-C. Roman. Lime: A model and middleware supporting mobility of hosts and agents. *ACM Trans. on Software Engineering and Methodology*, 15(3):279–328, 2006.
- [22] M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos. A declarative model assembly infrastructure for verification and validation. In S. Takahashi, D. Sallach, and J. Rouchier, editors, *Advancing Social Simulation: The First World Congress*, pages 129–140. Springer Japan, 2007.
- [23] B. Oancea, I. G. Rosca, T. Andrei, and A. I. Iacob. Evaluating java performance for linear algebra numerical computations. *Procedia CS*, 3:474–478, 2011.
- [24] A. Omicini, A. Ricci, and M. Viroli. Artifacts in the A&A meta-model for multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 17(3), June 2008.
- [25] P. V. Roy, S. Haridi, A. Reinefeld, J.-B. Stefany, R. Yap, and T. Coupaye. Self-management for large-scale distributed systems: an overview of the selfman project. In *Formal Methods for Components and Objects, LNCS No. 5382*, pages 153–178. Springer Verlag, 2008.
- [26] E. Sklar. Netlogo, a multi-agent simulation environment. *Artificial Life*, 13(3):303–311, 2007.
- [27] A. Slepoy, A. P. Thompson, and S. J. Plimpton. A constant-time kinetic monte carlo algorithm for simulation of large biochemical reaction networks. *The Journal of Chemical Physics*, 128(20):205101, 2008.
- [28] S. D. Team. [http://www.swarm.org/index.php/Main\\_Page](http://www.swarm.org/index.php/Main_Page). Swarm home page.
- [29] C. Versari and N. Busi. Efficient stochastic simulation of biological systems with multiple variable volumes. *Electr. Notes Theor. Comput. Sci.*, 194(3):165–180, 2008.
- [30] M. Viroli, M. Casadei, S. Montagna, and F. Zambonelli. Spatial coordination of pervasive services through chemical-inspired tuple spaces. *ACM Transactions on Autonomous and Adaptive Systems*, 6(2):14:1 – 14:24, June 2011.
- [31] M. Viroli, T. Holvoet, A. Ricci, K. Schelfhout, and F. Zambonelli. Infrastructures for the environment of multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 14(1):49–60, July 2007.
- [32] M. Viroli and F. Zambonelli. A biochemical approach to adaptive service ecosystems. *Information Sciences*, 180(10):1876–1892, 2010.
- [33] M. Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, 1st edition, June 2002.



# BioMASS: a Biological Multi-Agent Simulation System

Candelaria E. Sansores\*, Flavio Reyes\*, Hector F. Gómez\*, Juan Pavón† and Luis E. Calderín-Aguilera‡

\*Complex System Simulation Lab, Universidad del Caribe, Cancún 77528, México

Email: {csansores, freyes, fgomez}@ucaribe.edu.mx

†Facultad de Informática, Universidad Complutense de Madrid, 28040, Spain

Email: jpavon@fdi.ucm.es

‡Department of Marine Ecology, CICESE, 22860, México

Email: leca@cicese.mx

**Abstract**—This article presents an agent based model for the simulation of biological systems. The approach consists mainly of providing individual based models for each of the functional groups that conform an ecosystem. Functional groups (a term commonly used by ecologists) may represent a group of individuals (from the same or from different species) that share relevant attributes. This provides flexibility to configure different kinds of populations by parametrization without the need of programming, something useful for biologists. Additionally, a simulation tool implemented as a multi-agent system facilitates the analysis and understanding of ecological complexity. Multi-agent systems are proposed to address heterogeneity and autonomy demanded by the interdisciplinary individual-based modeling methodology. The objective of the system is to explore the intricate relationships among population and individuals, in an ecosystem approach. The main difference with other tools is the ability of incorporating individual decisions, based on metabolism and environmental conditions.

## I. INTRODUCTION

**S**IMULATION by means of computational tools is a widely used resource for ecologists to test hypotheses given the difficulty to carry out experiments directly in the natural environment. Some of the problems typically faced are those related to the impact of human activity on ecosystem health. Computational experimentation enables researchers to test their working hypotheses quickly, repeatedly and without affecting the environment. Given the large number of organisms and interactions that make up the most simple ecosystem, until recently the simulation tools have been based on numerical solutions to equations in which entire populations or groups of individuals are represented by means of statistical parameters. This is the case of Ecopath [1], a software tool that computes numerical solutions for a mass balance equation to estimate the changes in the biomass of an ecosystem such as reproduction and mortality rates, migration, predation and biomass extraction by anthropogenic activity. Ecopath models are initialized with data describing the groups of organisms belonging to the ecosystem under study. If there is available information about the spatial distribution of these populations,

it is possible to subdivide the study area in a grid and apply the computational tool for calculating time and space fluctuations of biomass for each population group inside the domain under study.

Ecopath is the most popular computer simulation tool for Ecology but is not unique. There are other tools such as EcoSim [2] or Goldsim [3] that use Monte Carlo techniques to simulate the uncertainty inherent to the environmental systems. However, despite its enormous value as research tools, models based on numerical calculations reduce to statistical parameters the description of the populations of organisms, leaving out of the model important information about the many and different forms of relationships among the individuals within an ecosystem. For example, Ecopath only considers one kind of relationship, predator-prey, and yet this is parameterized in the form of biomass transferred among as many food items in the diet. Many other relationships such as competition or symbiosis are ignored even if their importance in the functioning of an ecosystem might be substantial. But more importantly, these models ignore the ability of individuals within a population to interact with the environment, modify it while at the same time adapting to it.

*Individual-based modeling* (IBM) has appeared [4] to support more flexibility and analysis abilities when modeling ecosystems. IBM, rather than global parameters about the populations use descriptions at the individual level to build an ecosystem model and allocates computational resources to simulate the development of every organism included in the model, making explicit the relations that each one might establish with other organisms and the environment. The behavior exhibited by the populations, their statistical indicators and the cause-effect relationship with respect to the environment are properties that must *emerge* from the aggregation of the activities and interactions of all the individuals. It is expected that the macro-scale predictions produced by IBM show high concordance with those obtained by the numerical models. But models that are based on individuals are able to produce much more detailed information about the simulated entities and their interrelationships, both, in time and space scale. To model a biomass transfer between two population groups in an IBM it is necessary to define the process of hunting-

This research was supported by the Mexican Council for Science and Technology (CONACYT) with grant QROO-2008-01-92231, the Mexican Ministry of Education with grant PROMEP/103.5/09/6028 and by the Spanish Council for Science and Innovation with grant TIN2008-06464-C03-01



capture-ingestion that may occur when two individuals become spatially close at a given time. Similarly it is possible to model the corresponding menace detection-avoidance process. Thus, the predator-prey relationship is modeled with richer detail than in traditional numerical modeling. Just as this, in an IBM it is possible to model many other kinds of relationship for reproduction, competition for space, for collaboration, etc.

In the case of the Mexican Caribbean coast, there is great concern about the effect that the intense tourist activity in this area has on the health of the ecosystems associated with this area's coastal reefs. These reefs are part of the Mesoamerican Reef System which is the second largest barrier reef in the world after the Great Barrier Reef located on the northeast coast of Australia. In a typical application of numerical modeling programs, Arias-Gonzales et al [5] studied the relationship between fishing activities and the health of the coastal reef ecosystems at three locations of the Mexican Caribbean using the simulation tool Ecopath. The study provides statistical evidence on the negative effect of fishing activities on coral reef health and the ecosystems that sustain them. However, the computational tool cannot yield information about the intimate mechanisms that might explain the connecting thread between cause and effect.

The long-term goal of this project is to find explanations for the statistically-proven interrelated phenomena of human activity and the decline of coastal marine ecosystems of the Mexican Caribbean. To this end we initiated the construction of the BioMASS software tool for simulation of biological ecosystems by applying the IBM approach. This tool is based on a strict model for mass and energy transfer using a rich predator-prey relationship. This allows the study of trophic chains within ecosystems while allowing the specification of many other types of relationships among organisms.

The next section describes the IBM approach and proposes a set of techniques for its successful implementation under the paradigm of multi-agent systems. Section III presents the class structure that supports the simulation system and the composition relationships among the objects involved. It also describes the energy transfer, individual growth and emergent statistical information gathering mechanisms. Section IV shows the user interface on a prerelease version of the system and how it is used to set up an experiment. Section V presents some experimental results using a rather simple hypothetical ecosystem. Section VI addresses some conclusions and describes future work.

## II. INDIVIDUAL BASED MODELING

IBM is a term used in Ecology to name a class of models that describes individual organisms as computational entities. These models are the foundation for a new methodology to address the complexity of ecological systems [6].

Traditionally, ecological systems are studied by formulating simplified representations of these systems using mathematical equations. The equations are formally solved to answer the questions the model was devised for. However, to keep the mathematical equations explicitly soluble, analytically or

numerically, the assumptions of the models are very simple. These assumptions are symbolized through variables characterizing the state of the whole system. In this sense, traditional models are more like a macro perspective of a system limited to properties descriptions at this level.

On the other hand, IBM promotes computer simulation models that are suited to observe how unforeseen system level properties emerge from the adaptive behavior of individual organisms. IBM also allows ecologists to study how this macro-structural emergent phenomena influences individuals behavior, that is, the macro-micro causality. Thus, computer simulation can be an effective tool to build less simplified models overcoming certain limitations of traditional techniques to explain patterns, processes and to predict the behavior of a system in response to individual changes and their interactions.

In IBM, individual organisms belonging to a system and their autonomous behavior are explicitly represented. These individuals are also called agents and are commonly implemented as software objects. According to the IBM paradigm, agents should be described as heterogeneous and autonomous entities that exhibit adaptive behavior, usually through learning skills, adjusting their behavior to each other and to the changing environment. The agents interact locally, meaning they interact spatially only within a given neighborhood. Autonomous means that they act without a central direction or control, rather pursuing their goals, as organisms strive to survive and reproduce in nature. Being heterogeneous implies being unique among a population. This uniqueness should be described using its genetic content, its morphological and physiological characteristics, the physical space that occupies (two individuals cannot be situated at the same place at the same time) and the local nature of its interactions with its environment.

Despite the several advantages of IBM over mathematical theorizing, IBM is more complex to develop because IBM requires computer-programming abilities by the modeler. An IBM has to be implemented as a system of software agents and the individual uniqueness should be reflected as a set of computational attributes. Another issue related with IBM is the difficulty to analyze and understand the simulation model results, a hard task that requires multiple model executions over time and systematically varying initial conditions in order to assess the robustness of the results set.

To overcome these difficulties IBMs can be developed as a Multi-Agent Systems (MAS), a paradigm that came from the field of distributed artificial intelligence (DAI). A multi-agent system is a collection of software agents developed mostly to solve tasks in a distributed manner. Each one of these agents is a computer system situated in a given environment that is capable of autonomous action in order to meet its design objectives [7]. It has proof to be appropriate for the representation of ecological dynamics, which state modeling problems in terms of representations, communication and controls [8]. The abstractions of this paradigm are especially adequate to simulate at the individual level the physiological processes, the

life cycle, the decision-making mechanisms, the interactions with other individuals and with the environment, and other processes that determine the local state of an organism. There are many theories, methodologies, mechanisms and tools from the Agent Oriented Software Engineering (AOSE) to design and implement MAS [9]. Also there is another trend to provide discipline specific MAS developing tools, such as agent-based social simulation tools, agent-based computational economics simulation tools, and so on [10]. However, it is important to be especially careful with the selection of these tools since many of them lack the artifacts to achieve the mentioned autonomy and uniqueness.

The BioMASS tool is a Java application based on the Repast agent-based modeling and simulation platform [11] with the purpose of building individual based models (IBMs) for ecological systems. It considers that for implementing IBM under the MAS paradigm, IBM simulation software should meet the following requirements as formulated in [12]:

- 1) The capacity of the model to represent each individual (not a collection) by means of a dedicated software agent.
- 2) The capacity of the model to endow each individual with distinctive traits to ensure the uniqueness of each entity inside the temporal and spatial context of the simulation (this guarantees that each individual might be different with respect not only to others but with itself at different moments)
- 3) The ability of the model to represent the way the individual utilizes resources by means of its local and direct interaction with its environment.
- 4) The ability of the model to reflect the phases through which the individual goes along its life cycle.
- 5) The size of a population and all other statistical variables of the simulation may be computed at any time from the result of the aggregated accounting of the single individuals and not from inference.
- 6) The population dynamics (of cyclical nature) must emerge from the integration of the life cycle of its individuals (which rather than cyclical is lineal) through generational change.

The above characteristics guarantee individual's heterogeneity and are attained through the computational artifacts of the BioMASS tool. Thus, the agents instantiated by this tool automatically conform to the above criteria. Autonomy is also achieved within the simulation tool by a thought design, providing scheduling mechanisms that allow agents to freely behave without central coordination. Using the BioMASS tool, the development of an IBM for an ecological system would then consist on defining and instantiating a population of agents with some distribution of initial states and during the simulation execution each individual will follow a differentiated and autonomous living path.

### III. AGENTS MODEL: STRUCTURE AND DYNAMICS

One purpose of the BioMASS ecosystem tool is to facilitate biologists to create their specific models. As they may not be

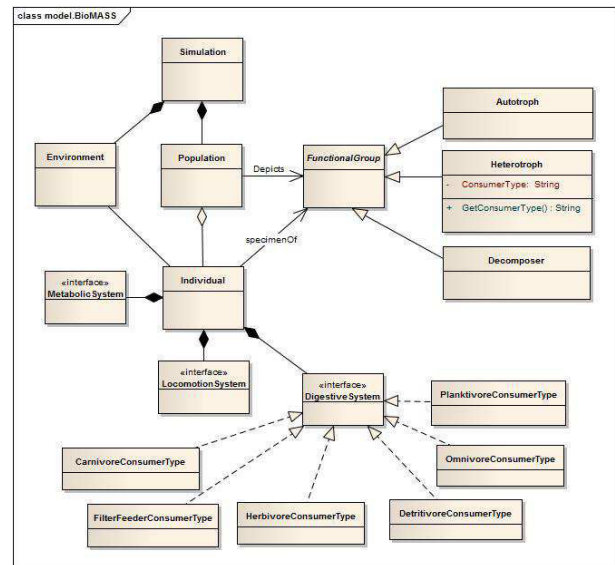


Fig. 1. BioMass class model excerpt

experts on computing programming, the tool provides a set of assistants that should help them to define the species in an ecosystem, their deployment, and other simulation parameters. Therefore, the agent model should be general enough to satisfy the requirements of a wide set of species and well-structured to facilitate its extensibility when needed.

The BioMASS model depicted in Fig. 1 is inspired on previous work by Parrott and Kok [13], but modified in three senses. First, rather than simulating a specific ecosystem, this paper presents a software tool that enables the non programmer to model different types of ecosystems. Second, this paper emphasizes the definition of functional groups (a term commonly used by ecologists) that may represent a group of individuals (from the same or from different species) that share relevant attributes in the context of the simulation. Third, we have turned to an architecture based on a set of interrelated objects to represent a living organism (an individual of a functional group) and a set of interchangeable modules representing the different systems that compose it. To this end, the *Individual* class consists of a set of well defined interfaces such as *DigestiveSystem*, *MetabolicSystem* and *LocomotionSystem* interfaces which will be realized by concrete implementation classes following the special characteristics of each functional group. Thus, an individual organism, as shown in Fig. 1, is made of several components. There is a taxonomy of classes of organisms which is expressed as inheritance relationships with *FunctionalGroup* abstract class on the top and the *Autotroph*, *Heterotroph* and *Decomposer* classes underneath. Each new functional group results as the instantiation of one of these classes and adjusting a set of attributes to characterize all its individuals. During the simulation a *Population* class object will be instantiated as well for each functional group required in the model under study, which in turn will spawn the *Individual* objects representing the organisms of such population.

Microorganisms are considered as part of the environment

modeled as a Repast Context, which provides methods to estimate their density and to account on the effect of these microorganisms, such as the decomposition of death organisms by bacterial activity, or the biomass primary production by phytoplankton activity.

All organisms have a location in the environment, a mass and a volume. They follow a life-cycle from birth to death and then decompose. During their life-cycle they pass through different phases or states following a path determined for their species. The state has influence on different factors such as growth rates, the ability to reproduce, or physical shape.

The agents have been defined taking into account the main activities that are relevant for the biomass flow through an ecosystem. These activities are shown in Fig. 2. As depicted in this activity diagram, the organism may be alive, in which case, at each simulation cycle, it will have to spend a certain amount of energy for living, following the model explained in section III-A. Then it has to check its surrounding and depending on the urgency of avoiding a threat or the need for food, the agent will decide to escape or will attempt to capture some prey. These actions imply energy consumption for moving, and in the case of feeding, also for the metabolic functions. Then individuals can wander around, grow and reproduce depending on their state. These activities also imply some additional energy expenditure, and in consequence, changes of mass. Feeding and reproduction imply as well mass exchanges with the environment as explained in section III-A. These activities have been used to identify the components of the individual agent, which is seen as a set of systems: locomotion, sensorial, digestive, etc. It has been considered as well a decision system for the agent to have some memory of past events and some ability to reasoning and adaptation, although this has not been implemented yet.

As IBM has gained in popularity, software tools have appeared to implement them. There are various ecological-oriented agent-based simulation (ABS) platforms comparable with the one proposed in this work, such is the case of Mimosa [14], CORMAS [15] and VLS [16]. Mimosa is a framework based on ontologies for knowledge representation in a discrete-event system simulation, in this framework ecologists must first describe the concepts or categories they are using to describe the target system. CORMAS is a multi-agent simulation toolkit to model social dynamics in interaction with natural resource dynamics. It provides the framework for building models of interactions between individuals and groups sharing natural resources. Its main purpose is to provide the possibility to manipulate and to incorporate into the same model spatial entities defined at different hierarchical levels. VLS is a software and an API which supports multi-modelling, simulation and analysis of natural complex systems from environmental sciences like physics, biology, ecology and economy for the integration of heterogeneous models.

The previous frameworks are originals and can be used to simulate environmental issues. Nevertheless, the BioMASS model proposed in this paper is built on a strong theoretical basis and it does not aim to be a general ecological ABS tool

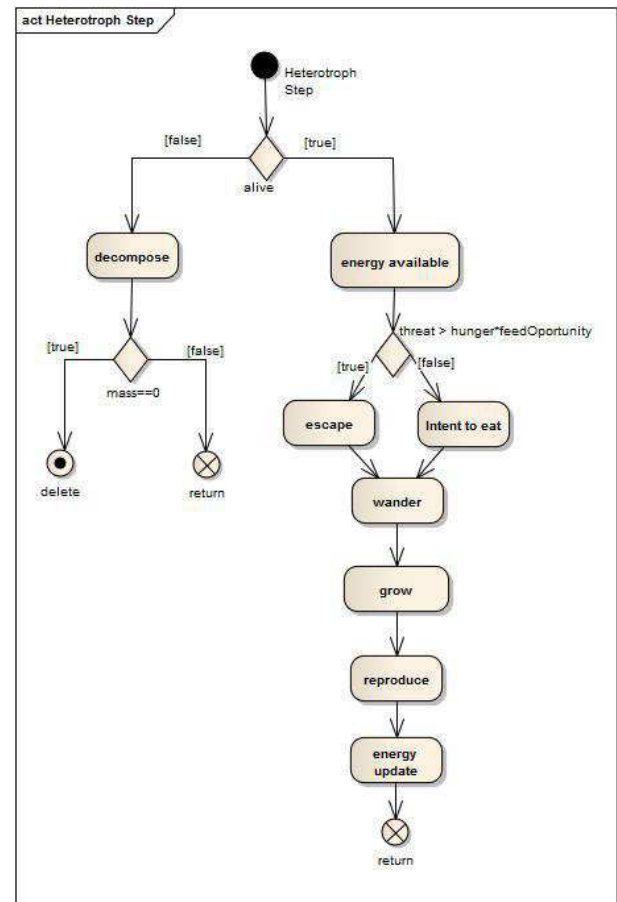


Fig. 2. Individual activities performed at each step of simulation

as the ones cited above, instead it is a domain-oriented easily customizable open tool that provides ecologists with a modeling and simulation tool that leverage the potential of MAS methodologies and architectures to enforce the implementation of true individual models.

#### A. Agents Resource Dynamics

The main purpose of the model is to analyse material and mass flows in the trophic chain. Therefore, simulation of agent resource dynamics becomes a relevant part of the model.

The physical resources employed by a population of living organisms in their interaction with the environment are ultimately defined by the local and independent relationship between each individual and its surroundings. The local resource utilization is a very important factor for diversity among organisms of the same class and age and the only way for energy and mass transfer between the living populations and the physical environment.

Under this approach, the cumulative biomass is the most important distinguishing feature of each individual since it determines the amount of energy available for the organism functions as well as the amount of resources to be taken from the environment. As the individual grows gaining biomass, other attributes such as energy consumption, strength and

speed of displacement might be automatically modified according to a predefined relationship for this kind of organism. In BioMASS, and with the aim of adding realism to the model graphical representation, density and volume spatial attributes are also included in each individual because they are essential for its graphical representation and for the definition of the physical interactions among the individuals or between them and the environment structure.

Following Parrott and Kok's ideas [13], each organism is modelled as being composed of lean mass and fatty mass both of which increase through the ingestion of another organism with similar composition. The fatty mass, composed of lipids and water, is susceptible of being transformed into energy through metabolism and for that reason might increase or decrease according to the rate of biomass ingestion and energy usage. In contrast, the lean mass never decreases because at this stage does not exist any process for protein conversion into lipids, carbohydrates or energy (as it happens in nature).

Every organisms may eat as much food as the physical volume of its stomach and the availability of food allows it. Then, under a periodical basis the ingested biomass, or a portion of it, is incorporated to the animal own biomass, but only enough to satisfy both, the requirements imposed by a programmed growth rate function specific for the functional group, and the fatty mass deficit caused by energy consumption. The excess of biomass ingested is expelled as waste. The proportion of fatty mass to lean mass is kept under certain limits that are specific for each functional group, but being subject to conversion into energy and thus consumed, the fatty mass might decrease beyond a critical limit and the animal starve to death. The system incorporates a mechanism to collect the biomass of death organisms and digestive waste transferring that biomass to the software object that represents the environment for further quantification and decomposition.

### B. Growing as a differential trait (life cycle)

In nature, the organisms go through different development stages or phases as they complete their life cycle. It is common that starting as larvae, organisms evolve into juvenile, mature and senile individuals presenting at each phase important and distinctive physical attributes, like size and form, and physiological characteristics such as their capability to reproduce or their feeding habits. Therefore, an IBM should be able to reflect these changes.

In this case, the differential characteristics related to the individual life cycle involve two aspects: feeding habits (some animals, specially in marine ecosystems, are herbivores during larval stage and carnivores there after) and growth rate (usually younger organisms grow faster). The growth rate,  $r_t$ , for every individual in age  $t$ , is modeled using the proposal of [13]:

$$r_t = k_2 m_t e^{-k_1 t} e^{k_3 (m_{p_t} - m_t)} . \quad (1)$$

Under this model, the growth rate is directly proportional to the lean mass,  $m_t$ , and due to the term  $e^{-k_1 t}$  younger individuals grow faster than older ones. When the lean mass is

below its potential value  $m_{p_t}$  the growth rate is multiplied by the factor  $e^{k_3 (m_{p_t} - m_t)}$  in order to reduce the difference. The constant values  $k_1$ ,  $k_2$  and  $k_3$  are positive and species-specific.

Once the growth rate is evaluated the lean mass is actualized with the next discrete approximation

$$m_{t+1} = m_t + c_t r_t \Delta t , \quad (2)$$

where coefficient  $c_t \in [0, 1]$  measures food availability.

The potential lean mass is obtained setting  $c = 1, \forall t$  (food is completely available at every instant). Determination of the constants  $k_1$ ,  $k_2$ ,  $k_3$  on the base of the growth function for this model is neither straightforward nor intuitive. It is more natural to establish restrictions on the lean mass in order to deduce the values of the constants. Indeed,  $k_1$  and  $k_2$  can be easily evaluated if the growth function is stated as a differential equation of the potential lean mass

$$\frac{d(mp)}{dt} = k_2 m_t e^{-k_1 t} , \quad (3)$$

with solution

$$mp(t) = C e^{-\frac{k_2}{k_1} e^{-k_1 t}} , \quad (4)$$

where  $C$  is a new and arbitrary constant.

Applying the restriction  $mp(t = 0) = m_0$  (lean mass at birth),  $C$  is determined and the lean mass function can be rewritten as

$$mp(t) = m_0 e^{\frac{k_2}{k_1} (1 - e^{-k_1 t})} . \quad (5)$$

Considering that the lean mass attains its maximum value,  $m_f$ , asymptotically

$$\lim_{t \rightarrow \infty} mp(t) = m_f = m_0 e^{\frac{k_2}{k_1}} \quad (6)$$

the next equation is obtained

$$k_2 = k_1 \log \left( \frac{m_f}{m_0} \right) \quad (7)$$

A further restriction of the lean mass at adult age,  $mp(t = A) = \epsilon m_f$ , yields

$$k_1 = \frac{1}{A} \log \left[ \frac{\log \left( \frac{m_0}{m_f} \right)}{\log \epsilon} \right] \quad (8)$$

In the simulation every individual computes its growing rate periodically basing this computation on its present mass and age and then depending on the available resources (ingested biomass) increases its size.

### C. Population Dynamics as Emergence

Section II has discussed the importance for an IBM to be able to produce the population dynamics and statistics as emergent products of the generational succession of individuals. However, in practice it is almost impossible to reach equilibrium for a population in such a way that its age distribution is sustainable through several generations if we do not start with an already adequate population. For this, an initial population must be carefully constructed. One way to do this is to assume a constant death rate to obtain an exponential age distribution (details can be found in the 9th chapter of [17])

$$N_i = M e^{\delta(i\Delta t)} \quad (9)$$

where  $N_i$  is the expected population for age group  $i$  ( $i \in \{0, 1, 2, \dots\}$ ),  $\Delta t$  is the lapse of time separating two consecutive generations,  $M$  is the expected number of individuals incorporated in a new generation and  $\delta$  is the constant death rate. On future developments BioMASS will incorporate not only this but other tools to produce the initial populations for the different functional groups.

### IV. SIMULATION CONFIGURATION

BioMASS tool allows biologists to define and simulate ecological models following a set of steps, which are guided by an assistant. These steps are essentially the following:

- 1) Define the functional groups to be incorporated into the model. It is possible to identify an already defined functional group that is close in properties and behavior to the new class.
- 2) Fill in the parameters that characterize the functional group.
- 3) Define the environment. This is a 3D grid with a set of parameters to configure as well.
- 4) Define the population parameters for each functional group.
- 5) Define simulation execution parameters such as number of iterations, files to keep monitoring data, etc.

Some of the steps are expected to be improved in the future, for instance, to allow defining new behavioral rules for the different functional groups.

### V. EXPERIMENTATION

BioMASS has been tested with a simulation model of an hypothetical marine ecosystem. In this simulation model a new generation of aquatic pelagic individuals (fish) belonging to two different functional groups, herbivores and carnivores, are instantiated periodically as if they were born from randomly dispersed eggs. The herbivores feed on plankton which is uniformly distributed over the space. However the density of plankton is dynamically updated by following the logistic differential equation and considering the total amount of consumption by herbivores:

$$\frac{dP}{dt} = k \left( 1 - \frac{P(t)}{M} \right) P(t) - C(t). \quad (10)$$

In this differential equation,  $P(t)$  is the total amount of plankton biomass at time  $t$ ,  $k$  is a proportionality constant,  $M$  is the maximum plankton biomass that the environment can support and  $C(t)$  is the consumption rate by herbivores.

Herbivores gain biomass rigorously following their potential growing function and the plankton availability. Carnivores predate on herbivores whenever they get close enough, they also gain biomass following their potential growing function, but unlike herbivores, their growing function depends on the herbivores availability. Both carnivores and herbivores move randomly (at this development phase there are not hunting or evasion functions) and the result of close encounters among them may be carnivores eating the herbivores (if the size difference allows it), or simply a direction change for collision avoidance.

Organisms incorporate biomass as they feed following the growth function described above. They also spend biomass (fatty mass) converted to energy as they live and move but might replenish their reserves. If they do not do it, after some time they might starve to death. Fig. 3 shows a screenshot of the running simulation system illustrating the interface used to define new species and their attributes. It also shows a rendering of the simulated world with organisms represented by means of ellipsoids with different size, position and orientation. Fig. 4 depicts the evolution of the amount of biomass of the system and Fig. 5 shows the population distribution by age classes for one moment of the simulation.

Under this preliminary configuration with only two functional groups participating on the simulation, the system showed a tendency to instability either by extinction or explosion of the populations. The determining factors for either of these results were the number of new organisms spawned on each generation and the rate at which they grow. The exploding scenario was preferable since it could be contained by controlling the amount of new biomass introduced into the system in the form of environment's plankton.

Thanks to the user graphical interface of the simulation tool it was easy to manipulate the parameters of the simulation so the experimental cycle of setting, running and feedback could be effortlessly and quickly repeated.

### VI. CONCLUSIONS

Biomass tool offers ecologists the ability to design computational simulation experiments about natural ecosystems using a user-friendly graphical interface and concepts familiar to them. With this interface the researcher describes one or more functional groups which are classes of individuals with common characteristics, then populations from these functional groups and finally the parameters of the simulation itself. Among the adjustable parameters for the definition of the functional groups are those that distinguish a set of individuals such as diet, minimum and maximum size, growth rate, etc. Some of these features are related to the behavior of the individuals of a functional group enabling BioMASS to be used to model relationships among organisms that go beyond the predator-prey relationship.

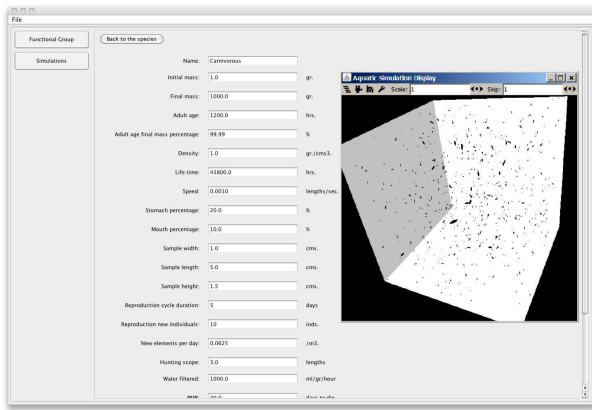


Fig. 3. Graphical user interface

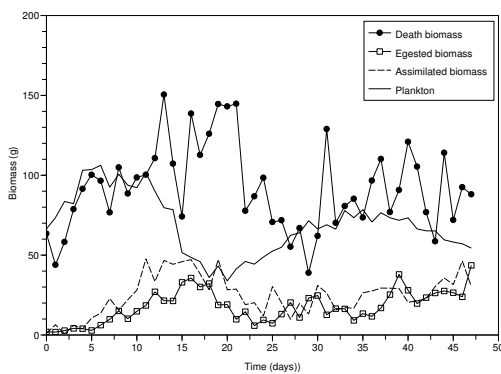


Fig. 4. Biomass dynamics in a simulation run

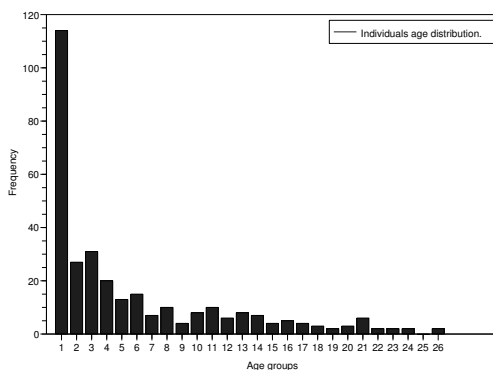


Fig. 5. Population age distribution at a given day

The selection of adjustable parameters for the definition of functional groups is sufficiently broad to permit the configuration of great diversity of simulation scenarios using the GUI provided. If necessary, Biomass provides the ecologist with programming skills and object-oriented architecture that can be extended to incorporate new properties or modify the existing ones.

Based on the preliminary results of the experimentation phase, and as future work we envisage to extend the tool to consider a spatial distribution of primary production of biomass resources (such as plankton). Also, we intend to incorporate some simple hunting and evading functions as well as a more realistic energy consumption model for individuals. An open issue to be addressed is the computing power necessary to simulate the great amount of individuals of some of the most interesting and complex ecosystems such as the tropical reefs.

#### ACKNOWLEDGMENTS

Authors thank to the three anonymous reviewers for their helpful commentaries to the paper.

#### REFERENCES

- [1] V. Christensen and C. J. Walters, "Ecopath with ecosim: methods, capabilities and limitations," *Ecological Modelling*, vol. 172, no. 2–4, pp. 109–139, 2004.
- [2] N. J. Gotelli and G. L. Entsminger. (2011) Ecosim: Null models software for ecology. [Online]. Available: <http://garyentsminger.com/ecosim.htm>
- [3] GoldSim. (2008) Monte carlo simulation software. [Online]. Available: <http://www.goldsim.com>
- [4] V. Grimm and S. F. Railsback, *Individual-Based Modeling and Ecology*. Princeton University Press, 2005.
- [5] J. E. Arias-Gonzalez, E. Nunez-Lara, C. Gonzalez-Salas, and R. Galzin, "Trophic models for investigation of fishing effect on coral reef ecosystems," *Ecological Modelling*, vol. 172, no. 2–4, pp. 197–212, 2004.
- [6] D. L. DeAngelis and W. M. Mooij, "Individual-based modeling of ecological and evolutionary processes," *Annual Review of Ecology, Evolution, and Systematics*, vol. 36, no. 1, pp. 147–168, 2005. [Online]. Available: <http://www.annualreviews.org/doi/abs/10.1146/annurev.ecolsys.36.102003.152644>
- [7] J. Ferber, *Multi-agent systems: an introduction to distributed artificial intelligence*. Harlow: Addison-Wesley, 1999.
- [8] F. Bousquet, O. Barreteau, C. L. Page, C. Mullon, and J. Weber, *An environmental modelling approach: the use of multi-agent simulations*, ser. Advances in environmental and ecological modelling. Elsevier, Paris, 1999, pp. 113–122.
- [9] C. E. Sansores and J. Pavón, "Agent-based simulation replication: A model driven architecture approach," *Lecture Notes in Artificial Intelligence*, vol. 3789, pp. 244–253, 2005.
- [10] F. Bousquet and C. L. Page, "Multi-agent simulations and ecosystem management: a review," *Ecological Modelling*, vol. 176, no. 3–4, pp. 313–332, 2004.
- [11] Repast. (2011) Recursive porous agent simulation toolkit: Symphony. [Online]. Available: [http://repast.sourceforge.net/repast\\_simphony.html](http://repast.sourceforge.net/repast_simphony.html)
- [12] C. E. Sansores, F. Reyes, H. F. Gómez, and O. Molnár, "On the improvements of computational individualism of an ibm," in *Soft Computing Models in Industrial and Environmental Applications, 6th International Conference SOCO 2011*, ser. Advances in Intelligent and Soft Computing, vol. 87. Springer Berlin / Heidelberg, 2011, pp. 533–542.
- [13] L. Parrott and R. Kok, "A generic, individual-based approach to modelling higher trophic levels in simulation of terrestrial ecosystems," *Ecological Modelling*, vol. 154, no. 1–2, pp. 151–178, 2002.



- [14] J. P. Müller, "A framework for integrated modeling using a knowledge-driven approach," in *International Congress on Environmental Modelling and Software Modelling for Environment's Sake*, ser. International Environmental Modelling and Software Society (iEMSs), Fifth Biennial Meeting, Ottawa, Canada, W. Y. D. A. Swayne, A. A. Voinov, A. Rizzoli, and T. Filatova, Eds., 2010, pp. 826–837.
- [15] F. Bousquet, I. Bakam, H. Proton, and C. L. Page, "Cormas: Common-pool resources and multi-agent systems," in *Proceedings of the 11th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems: Tasks and Methods in Applied Artificial Intelligence*, ser. IEA/AIE '98, A. P. del Pobil, J. Mira, and M. Ali, Eds. London, UK: Springer-Verlag, 1998, pp. 826–837. [Online]. Available: <http://portal.acm.org/citation.cfm?id=646866.759746>
- [16] G. Quesnel, R. Duboz, and E. Ramat, "The Virtual Laboratory Environment – An operational framework for multi-modelling, simulation and analysis of complex dynamical systems," *Simulation Modelling Practice and Theory*, vol. 17, pp. 641–653, April 2009.
- [17] J. P. M. de Sa, *Applied Statistics Using SPSS, STATISTICA, MATLAB and R*. Springer Verlag, 2007.



# International Symposium on Multimedia Applications and Processing

**M**ULTIMEDIA information has become ubiquitous on the web, creating new challenges for indexing, access, search and retrieval. Recent advances in pervasive computers, networks, telecommunications, and information technology, along with the proliferation of multimedia mobile devices—such as laptops, iPods, personal digital assistants (PDA), and cellular telephones—have stimulated the development of intelligent pervasive multimedia applications. These key technologies are creating a multimedia revolution that will have significant impact across a wide spectrum of consumer, business, healthcare, and governmental domains. Yet many challenges remain, especially when it comes to efficiently indexing, mining, querying, searching, and retrieving multimedia data.

The Multimedia—Processing and Applications 2011 (MMAP 2011) Symposium addresses several themes related to theory and practice within multimedia domain. The enormous interest in multimedia from many activity areas (medicine, entertainment, education) led researchers and industry to make a continuous effort to create new, innovative multimedia algorithms and application. As a result the conference goal is to bring together researchers, engineers and practitioners in order to communicate their newest and original contributions on topics that have been identified (see below). We are also interested in looking at service architectures, protocols, and standards for multimedia communications—including middleware—along with the related security issues, such as secure multimedia information sharing. Finally, we encourage submissions describing work on novel applications that exploit the unique set of advantages offered by multimedia computing techniques, including home-networked entertainment and games. However, innovative contributions that don't exactly fit into these areas will also be considered because they might be of benefit to conference attendees.

## TOPICS

Topics of interest are related to Multimedia Processing and Applications including, but are not limited to the following areas:

- Image and Video Processing
- Speech, Audio and Music Processing
- 3D and Stereo Imaging
- Distributed Multimedia Systems
- Multimedia Databases, Indexing, Recognition and Retrieval
- Data Mining
- Multimedia in E-Learning, E-Commerce and E-Society Applications
- Multimedia in Medical Applications
- Multimedia Authentication and Watermarking
- Entertainment and games
- Multimedia Interfaces

## GENERAL CHAIR

**Dumitru Dan Burdescu**, University of Craiova, Romania

## STEERING COMMITTEE

**Ioannis Pitas**, University of Thessaloniki, Greece

**Costin Badica**, University of Craiova, Romania

**Borko Furht**, Florida Atlantic University, USA

**Harald Kosch**, University of Passau, Germany

**Vladimir Uskov**, Bradley University, USA

**Thomas M. Deserno**, Aachen University, Germany

**Mohammad S. Obaidat**, Monmouth University, USA

## PROGRAM COMMITTEE

**Carl James Debono**, University of Malta, Republic of Malta

**Michael Lang**, National University of Ireland, Ireland

**David Bustard**, University of Ulster, UK

**Janis Grundspenkis**, Riga Technical University, Latvia

**Rynson Lau**, Shanghai University, P.R. China

**Che-Chern Lin**, National Kaohsiung Normal University, Taiwan

**Bogdan Logofatu**, University of Bucuresti, Romania

**Toshio Okamoto**, University of Electro-Communications, Japan

**Reda Alhadj**, University of Calgary, Canada

**Qi Chun**, Xi'an Jiaotong University, P.R.China

**Enn Öunapuu**, University of Technology, Tallinn, Estonia

**George Tsihrintzis**, University of Piraeus, Greece

**Stefan Trzcielinski**, Poznan University of Technology, Poland

**Wilfried Philips**, Universiteit Gent, Belgium

**Vladimir Cretu**, Politehnica University of Timisoara, Romania

**Igor Kotenko**, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Science, Russian Federation

**Kazuo Ohzeki**, Shibaura Institute of Technology, Japan

**Valery Korzhik**, State University of Telecommunications, St. Petersburg, Russian Federation

**Boris Shishkov**, IICREST / Delft University of Technology, Netherlands

**Daniel Grosu**, Wayne State University, USA

**Laszlo Böszörményi**, Klagenfurt University, Austria

**Vladimir Fomichov**, State University—Higher School of Economics, Moscow, Russian Federation

**Miguel Angel Vega-Rodríguez**, University of Extremadura, Spain

**Dan Popescu**, CSIRO, Sydney, Australia

**Marek Ogiela**, AGH University of Science and Technology, Poland

**Giuseppe Mangioni**, University of Catania, Italy

**Stefan Trausan-Matu**, Politehnica University of Bucharest, Romania

**Reggie Kwan**, Caritas Francis Hsu College, Hong Kong

**Rami Finkler**, Afeka College of Engineering, Tel Aviv, Israel

**Richard Chbeir**, Bourgogne University, France

**Christopher Barry**, National University of Ireland, Ireland

**Mihai Mocanu**, University of Craiova, Romania

**Andrea Molinari**, University of Trento, Italy

**Franz Wotawa**, Technische Universitaet Graz, Austria

**Christos Douligeris**, University of Piraeus, Greece

**Jacek Zurada**, University of Louisville, USA

**Ryszard Choras**, Institute of Telecommunications, Poland

**Shiguo Lian**, France Telecom R&D Beijing, P.R. China

**Rajkumar Kannan**, Bishop Heber College, India

**Abdel-Badeeh M. Salem**, Ain Shams University, Egypt

**Jaime Lloret Mauri**, Polytechnic University of Valencia, Spain

**Yoshimi Teshigawara**, Soka University, Japan

**Christian Timmerer**, Klagenfurt University, Austria

**George Thiruvathukal**, Loyola University, USA

**Voicu Groza**, University of Ottawa, Canada

**Jaime Lloret Mauri**, Polytechnic University of Valencia, Spain

**Asa Smedberg**, Stockholm University, Sweden

#### ORGANIZING COMMITTEE

**Dumitru Dan Burdescu**, University of Craiova, Romania

**Costin Badica**, University of Craiova, Romania

**Liana Stanescu**, University of Craiova, Romania

**Marius Brezovan**, University of Craiova, Romania

**Cristian Mihaescu**, University of Craiova, Romania

#### PUBLICITY CHAIR

**Amelia Badica**, University of Craiova, Romania

# Robust Digital Watermarking System for Still Images

Sergey Anfinogenov and Valery Korzhik (Member, IEEE)  
State University of Telecommunications  
St. Petersburg, Russia  
serganff@gmail.com , korzhik@spb.lanck.net

Guillermo Morales-Luna  
Computer Science, CINVESTAV-IPN  
Mexico City, Mexico  
gmorales@cs.cinvestav.mx

**Abstract**—Fast and wide-scale spreading of the image data on the Internet creates great opportunities for illegal access by the different kinds of intruders. In order to solve the problem of intellectual property protection, digital image watermarking can be successfully used. We describe a new method of digital watermarking based on the embedding of the local maxima into the Fourier transform area of the image. Simulation results are presented, which confirm that the proposed method is resistant to cyclic shifts, row and column removal, cropping, addition of noise, rotation and JPEG transforms.

## I. INTRODUCTION

DIGITAL watermarks (WM) can be effectively used for copyright protection to various products. However, intruders (the so-called *pirates*), who try to copy and spread illegally these products, attempt to remove the WM or to perform some transforms over the watermarked products which, without impairing the product itself, make impossible to extract the WM reliably by legal users. In this paper a new method of 0-bit WM system creation with *blind* decoder is proposed. This means that the system task is to find the fact that the WM is indeed present in the marked object. It is common to use the following criteria for the evaluation of the systems efficiency: the *probability of false detection* of the WM ( $P_{fa}$ ) and the *probability of WM missing* ( $P_m$ ).

There are two main approaches to provide resistance of WM against different deliberate transforms: the use of recovering transforms to reduce the attacked products to their original WM-ed forms, and the use of the invariant domain for such transforms which allow blind detection of WM's. The WM systems here introduced are based on the properties of mathematical conversions that establishes a domain which is invariant to the changes of the WM-ed products. Several such systems are based on the properties of the *Fourier-Mellin Transform*. As an example, we hold up the system introduced at [1], where the informed decoder has been used. However, that system cannot prevent the attacks that partly delete the cover image. There are attempts to refine the system characteristics by introducing a *Logarithmically-Polar Transform* (LPM). In [2] an example of such attempts is described. In theory, this method works properly, however in practice such system appears inapplicable. Even without embedding, the carrying out of direct and reverse LPM impairs seriously the quality of

the image and therewith it demands considerable computing resources.

A good example of a robust watermarking system is constructed by the holographic method, proposed in [3]. Nevertheless, it has one essential drawback: in order to extract the WM, it is necessary to possess the original image. W. Luo *et al.* [4] have proposed a fast and robust JPEG domain image watermarking method but it cannot be used for automatic watermark detection.

Thus, the problem of robust WM systems design resistant to the whole complex of transforms has not been solved completely. Our new method, which is an alternative solution, is described in the following section.

## II. DESCRIPTION OF THE ROBUST WM SYSTEM

The development of a WM system robust against a complex of natural and intentional conversions is not an easy problem, particularly when the *blind* decoder is used. In the previous section, a short description can be found of the basic approaches to fulfill this task. However, the problem has not been yet solved completely. Nevertheless, some progressive ideas have been borrowed by ourselves while developing the introduced new method.

There are some approaches, for instance [1], where the WM is embedded into the area of the Fourier amplitude spectrum, because this area is invariant under the image cyclic shift. Besides, it seems reasonable to embed the identification code of the owner into the position of some maxima, as proposed in [5]. However, the maxima in our method are located directly in the amplitude spectrum and do not undergo preliminary by log-polar conversion. According to the known recommendations, it is also necessary to select, not all frequency coefficients for an embedding, but only those of them which lie in the field of the middle frequencies.

The main idea of the proposed method consists in generating the local areas worked out by the stegokey, and replacing the center of each area by its amplitude spectrum maximum. The number of maxima should be chosen in such a way, on the one hand, to provide an acceptable quality of the image and, on the other hand, it should survive after a number of image deformations. If the position of the local area maximum coincides with the position given by the stegokey, then the maximum is considered to be recognized. The ratio of the

recognized maxima number and the total maxima number is compared with a threshold, and the decision about the presence or absence of a WM in the image can be outperformed.

The task of the local maxima formation can be solved as follows:

First, the matrix  $\mathbf{G}$  of the mutual-independent random values (in the unit real interval  $[0, 1)$ ) is generated. After that, the matrix  $\mathbf{Z}$  of the same order is created according to the following rule:

$$\mathbf{Z}[i, j] = \begin{cases} 1 & \text{if } \mathbf{G}[i, j] > \lambda \\ 0 & \text{if } \mathbf{G}[i, j] \leq \lambda \end{cases}$$

where  $i = 0, \dots, M - 1, j = 0, \dots, N - 1$ , and  $\lambda$  is some real-valued threshold. The parameter  $\lambda$  defines an amount of the units in matrix  $\mathbf{Z}$  and, therefore, the amount of the embedding areas as well. The higher is the value  $\lambda$ , the fewer units are there in  $\mathbf{Z}$ . The remaining positions are filled with zeros.

Now it is necessary to create a new matrix  $\mathbf{L}$  according to the following expression:

$$\mathbf{L}[i, j] = \begin{cases} 1 & \mathbf{Z}[i, j] = 1 \ \& \ S[a, i, j] < 1 \\ 0 & \text{otherwise} \end{cases}$$

where

$$S[a, i, j] = \sum_{m=i-a}^{i+a} \sum_{n=j-a}^{j+a} \mathbf{Z}[m, n]$$

and  $a$  is a predetermined integer which defines the size of the local areas.

The matrix  $\mathbf{L}$  forms a mask with randomly allocated values 1 which lie in the centers of not intersecting areas of size  $(2a + 1) * (2a + 1)$ .

Next, the values 1 at the matrix  $\mathbf{L}$  are replaced with 0's in those frequencies where the embedding is not performed. It is worth to note that in order to make real the image brightness values after the embedding of a WM, it is necessary to preserve the symmetry of the matrix. Therefore the lower half of the amplitude spectrum matrix should be replaced with a mirror display of the upper half of the matrix, that requires to restrict the embedding area with the upper half of the matrix  $\mathbf{L}$  only. Let us denote as  $\mathbf{K}$  the matrix thus obtained from  $\mathbf{L}$ , which is considered as a stegokey, useful in order to detect the WM.

After the performing of all previous steps we should change the amplitude matrix  $\mathbf{A}$  according to the matrix  $\mathbf{K}$ . For this purpose, we select areas  $\delta$  of size  $(2a + 1) * (2a + 1)$  in the amplitude matrix  $\mathbf{A}$ . The centers of these areas are allocated on the same positions as the 1's in the matrix  $\mathbf{K}$ . We replace the values of an amplitude in the center of each area by the maximum amplitude over this area multiplied by some coefficient  $\beta$ , where  $\beta \geq 1$ . The remaining values of the matrix  $\mathbf{A}$  are not changed. The coefficient  $\beta$  is selected in such a way that the new image does not differ visually from the original one. This process can be presented by the equation:

$$\mathbf{A}_w[i, j] = \begin{cases} \beta \max_{(m,n) \in I_a(i,j)} \mathbf{A}[m, n] & \text{if } \mathbf{K}[i, j] = 1 \\ \mathbf{A}[i, j] & \text{if } \mathbf{K}[i, j] = 0 \end{cases}$$

where

$$I_a(i, j) = \{(m, n) \mid \max\{|m - i|, |n - j|\} \leq a\},$$

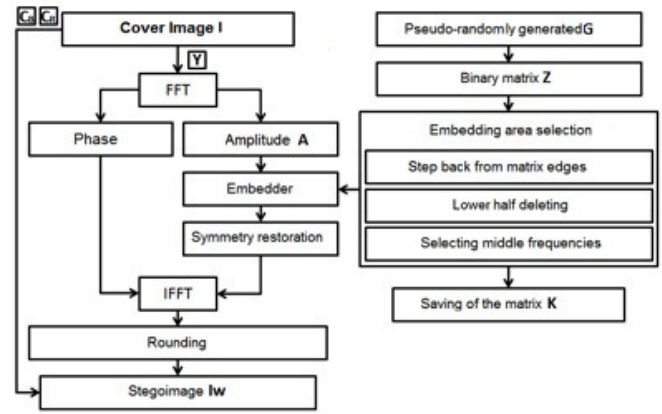


Fig. 1. Diagram of the watermark embedding method.

and  $\mathbf{A}[i, j]$  is the value of the two-dimensional Fourier amplitude at the point with coordinates  $(i, j)$ .

Let us restore the symmetry of the amplitude matrix. For this purpose we transform the first row in the matrix according to the formula:

$$\forall i = 2, \dots, M : \mathbf{A}_w[M - i + 2, 1] = \mathbf{A}_w[i, 1],$$

where  $i$  is the  $x$ -coordinate of the matrix current element. Then, we transform the first column in the matrix according to the formula:

$$\forall j = 2, \dots, N : \mathbf{A}_w[1, N - j + 2] = \mathbf{A}_w[1, j],$$

where  $j$  is the  $y$ -coordinate of the matrix current element. For all remaining elements of the matrix we use the ratio:  $\forall i = 2, \dots, M, j = 2, \dots, N$ :

$$\mathbf{A}_w[M - i + 2, N - j + 2] = \mathbf{A}_w[i, j].$$

After that, let us apply the inverse Fourier transform, connect three color components of the image and save it as the image after embedding.

The scheme of a WM embedding for a color image is presented in Fig. 1.

In order to extract a WM, firstly the fast Fourier Transform (FFT) of the luminance component of the image is performed and the amplitudes  $\mathbf{A}_w$  are calculated. Next, the areas of the size  $(2a + 1) * (2a + 1)$  with the centers in positions of the matrix  $\mathbf{K}$  values 1 are created. After that we count how many areas contain a maximum at their centers. This number is divided by the total number of the areas and compared with some threshold  $\Delta$ . In case the threshold is exceeded, the presence of a WM is detected, otherwise its absence is declared.

Besides, some modification of the extraction algorithm is made, allowing to detect a WM even after an image rotation. For this purpose, the image is sequentially rotated with the step  $0.5^\circ$  up to  $180^\circ$ , and the detection process described above is repeated each time. If the threshold is not exceeded at any step, then no WM is detected (see Fig. 2).

From the algorithms of the WM embedding and extraction given above we can see how the local maxima principle works.

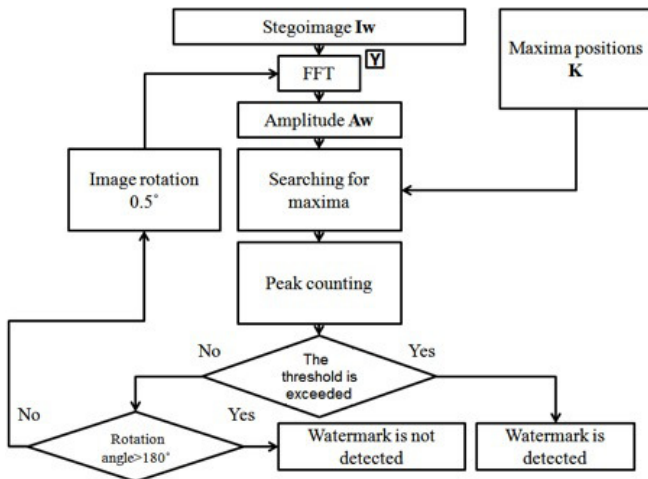


Fig. 2. Diagram of the watermark detection method.

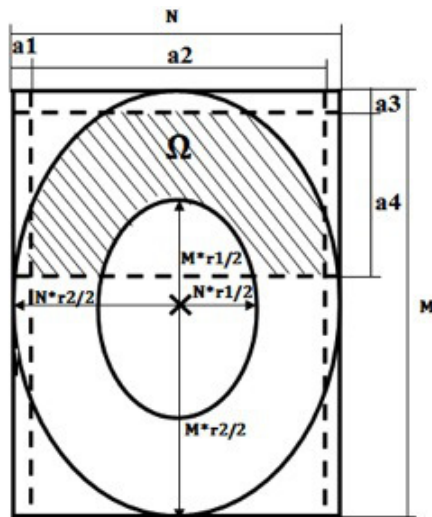


Fig. 3. The shape of the area  $\Omega$ .

On the one hand, the substitution of brightness amplitude in the centers of the areas for maxima with some coefficient ( $\beta \geq 1$ ) should not considerably worsen the image quality. And on the other hand, it is expected that various conversions of the image which do not worsen it considerably, will not change the position of maxima in local areas. However, in order to provide this expectations it is necessary to correctly select the main parameters of the proposed method. These parameters are the coefficient  $\beta$ , the area size  $(2a + 1) * (2a + 1)$  and the threshold value  $\Delta$ .

The experimental research of the method for various images has shown that the optimal size of the local areas is  $5 * 5$ . The parameters of the embedding area were selected experimentally as well. The geometry of the optimally selected area of embedding is shown in Fig. 3. Here,  $a_1$  is the number of rows from the left boundary of the matrix to the beginning of the area  $\Omega$ ,  $a_2$  is the width of the area  $\Omega$ ,  $a_3$  is the



Fig. 4. Cover images (left column) and stegoimages (right column) ( $a = 5, \beta = 1.2$ ).

number of columns from the upper boundary of the matrix to the beginning of the area  $\Omega$ ,  $a_4$  is the height of the area,  $r_1$  is the coefficient defining the sizes of an internal oval,  $r_2$  is the coefficient defining the sizes of an external oval. As far to the choice of the parameter  $\beta$ , its best value (by the results of many experiments with various images) has appeared to be equal to 1.2. The choice of the optimal detectability threshold is made in such a way to minimize the probability of false detection of a WM, and for each specific image the parameters of embedding can be chosen individually, just before the embedding of a WM.

In Fig. 4 the examples of two images without embedding and with embedding of a WM are shown. We can see that the images before and after embedding of a WM do not differ visually, which testifies indeed a high quality saving of the image after the WM embedding.

On the other hand, the choice of the threshold  $\Delta = 0.076$  allows us to make the decision about the presence or absence of a WM with an absolute reliability (i.e.  $P_m = 0, P_{fa} = 0$ ).

### III. INVESTIGATION OF ALGORITHM ROBUSTNESS

The results of the experiments presented in Table I show that for the choice of the threshold value  $\Delta = 0.076$ , the probability of false detection appears equal to zero. The probability of successful detection of a WM is equal to one also after the cyclic shift on 50% on a vertical and a horizontal, by the removal of 50% rows or columns, after rotation of the image on the angle up to  $50^\circ$ .

TABLE I  
EXPERIMENTAL RESULTS.

(1)	(2)	With embedding of a WM						
		(3)	(4)	(5)	(6)	(7)	(8)	(9)
max $\Delta$	0.07242	1	1	0.67864	0.53232	0.40438	0.25889	0.42827
min $\Delta$	0.03508	0.175	0.15009	0.04403	0.05653	0.05487	0.07753	0.076459
$\Delta$	0.058344	0.92828	0.92675	0.28575	0.24798	0.22337	0.11703	0.12692
Threshold $\Delta_D$	0.076	0.076	0.076	0.076	0.076	0.076	0.076	0.076
Probability of successful WM detection	0	1	1	0.95	0.94	0.98	1	1

- (1): Parameters  
 (2): No embedding  
 (3): Without distortions  
 (4): Cyclic shift of 50% on a vertical and a horizontal axis  
 (5): Noise adding 5%  
 (6): Removal of 10% of rows and columns  
 (7): Cropping  
 (8): Rotation on 5°  
 (9): Rotation on 50°

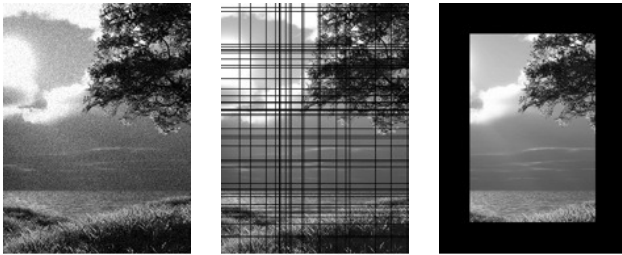


Fig. 5. Pictures after adding of the 10% noise, rows and columns removal and cropping.



Fig. 6. Pictures before and after adding false maxima.

In the Table I the normalized values  $\Delta$  of the recognized maxima number ratio to their total number of values 1 are presented, calculated as a result of 100 various images testing. For all experiments the parameters  $a = 5$ ,  $\beta = 1.2$  have been selected.

The probability is less than one, but it remains still acceptable after adding as noise 5% of the image brightness range. However, looking at the images after such strong conversions (Fig. 5) we can see that their commercial value is low, and it is very unlikely to be applied to the images by pirates.

Let us now recall that the Kerkhoffs principle with respect to steganography means that the attacker knows everything except the stegokey, i.e., first of all, the algorithm of embedding and extraction of a WM. Therefore, the attacker can apply more sophisticated attacks with the purpose of making impossible the reliable detection of a WM by the owner, but simultaneously, with saving its high quality.

In particular, knowing of the offered WM method, the attacker can try to add false maxima in Fourier amplitude spectrum in a hope that in this case a threshold  $\Delta_D$  will not be exceeded and, hence, the legal embedding will be not detected.

However, such attack appears unsuccessful because, although even if a WM is not detected the image is distorted significantly (Fig. 6).

Of course if the attacker gets a stegokey, he could have changed the position of local maxima and thereby he could provide the impossibility of the WM detection without distortion of the image. Although the stegokey is secure, pirates can try to find it by an analysis of the Fourier amplitude spectrum

with an embedded WM. However, in this way they face with insuperable difficulties because maxima are formed only in local areas and do not exceed the other maxima connected with the peculiarities of the original image.

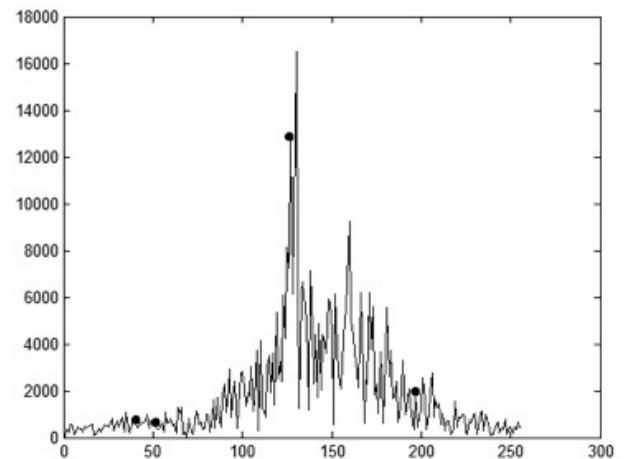


Fig. 7. One line of the amplitude coefficients of the Fourier transform.

In Fig. 7 one line of the amplitude matrix of the image with embedding is given as an example. Coefficients which form

the local maxima are marked with a filled small circle. We can see that even for a single line it is impossible to recognize the positions of maxima.

Simulation shows that after saving the WM-ed image in JPEG format, it is still possible to extract the WM for not very low quality (not worse than 30%). When the JPEG quality is worsened, the possibility of a WM extraction decreases and depends on a particular type of the image. It is possible to improve the robustness of the system by introducing an adaptive algorithm that will embed the maxima only into the frequency areas that will not change their values even after a strong JPEG compression. We can highlight the three main steps of such an adaptive algorithm:

- 1) Compare the FFT for the original image and the FFT of the same image after the JPEG compression at 5% quality factor.
- 2) Find the areas where amplitude changes are minimal.
- 3) Embed the local maxima only in those areas.

The performed experiments have shown that such an algorithm made it possible to extract the watermark after a 5% quality JPEG compression.

It is worth to note that if a WM is embedded in the luminance channel of the image, then it survives even after a stronger JPEG attack.

#### IV. CONCLUSIONS

A new method of WM embedding and extraction is proposed, that occurs to be robust against such transforms of

an image as cyclic shifting, rotation, removal of rows and columns, noise addition and cropping. The important advantage of this method is that it does not require the original image for WM detection.

It is reasonable to improve the method by conducting further research of the given method in order to provide better WM extraction after strong JPEG compression and collusion attacks [6]. One way to overcome the collusion attacks is to embed some additional maxima that would be the same for the same images and would not be erased after making an averaged copy.

#### REFERENCES

- [1] J. J. O. Ruanaidh, T. Pun, and J. J. K., "Rotation, scale and translation invariant digital image watermarking," in *IEEE Int. Conf. on Image Processing ICIP1997*, 1997, pp. 536–539.
- [2] C. Woo, J. Du, and B. Pham, "Geometric invariant domain for image watermarking," in *Proceedings of the International Workshop on Digital Watermarking, IWDW '06*. Springer LNCS Vol. 4283, 2006, pp. 294–307.
- [3] A. Bruckstein and T. Richardson, "A holographic transform domain image watermarking method," *CSSP Journal Special Issue*, vol. 17, no. 3, pp. 361–389, 1998.
- [4] W. Luo, G. L. Heileman, and C. E. Pizano, "Fast and robust watermarking of JPEG files," in *Proceedings of the Fifth IEEE Southwest Symposium on Image Analysis and Interpretation*. Washington, DC, USA: IEEE Computer Society, 2002, pp. 158–. [Online]. Available: <http://portal.acm.org/citation.cfm?id=882499.884595>
- [5] R. Ridzon and D. Levicky, "Robust digital watermarking based on the log-polar mapping," *Radioengineering*, vol. 16, no. 4, pp. 76–81, 2007.
- [6] K. J. R. Liu, W. Trappe, Z. J. Wang, M. Wu, and H. Zhao, "Multimedia fingerprinting forensics for traitor tracing," in *EURASIP on Signal Processing and Communications*. Hindawi, 2005.





# Estimating Topographic Heights with the StickGrip Haptic Device

Tatiana V. Evreinova, Grigori Evreinov and Roope Raisamo

School of Information Sciences, University of Tampere

Kanslerinrinne 1, FIN-33014 Finland

Email: {etv, grse, rr}@cs.uta.fi

**Abstract**—This paper presents an experimental study aimed to investigate the impact of haptic feedback when trying to evaluate quantitatively the topographic heights depicted by height tints. In particular, the accuracy of detecting the heights has been evaluated visually and instrumentally by using the new StickGrip haptic device. The participants were able to discriminate the required heights specified in the scale bar palette and to detect these values within an assigned map region. It was demonstrated that the complementary haptic feedback increased the accuracy of visual estimation of the topographic heights by about 32%.

## I. INTRODUCTION

TO EXTEND imaging capabilities, visualization of multi-dimensional data using two-dimensional printing techniques intended for a regular paper and flat screens requires to use the conditional pictorial means such as gray tones and color palettes. Nevertheless, scale bar palettes have to be optimized for converting changes of various physical values into the color gradient of intensity with a predefined step [1]–[4]. Such a function of transformation should rely on non-linear perceptual sensitivity of the human vision. However, the color-dependent sensitivity of the human eye is often neglected in processing and presentation of geographical information.

Moreover, some of people have perceptual problems related to color discrimination. Therefore, it is often impractical with an acceptable error rate to assess visually measurable topographic parameters, such as depth and elevation, being originally coded by intensity of gray tones or color gradient. Consequently the significance of scale bar palettes as a measuring tool (based on colorimetric matching) degrades. Variations in lighting conditions and perceptual interpreting of the shades and color parameters of images can significantly modify the true physical values. To compensate for a perceptual error, the landmarks on a map are usually accompanied with the labels of the true values being roughly transformed into brightness, contrast and saturation regarding the scale

bar palette. Discreteness of labeling depends on the map scale and display constraints. But, labeling cannot solve the problem to accurately display the variation in landscape metrics.

On the other hand, there is an increasing interest in geographical maps for traveling and navigation. At that, the information depicted in digital maps should be presented in a way that is easily accessed and understood. Multisensory integration of geoscientific data has been examined in a number of studies: for geophysical exploration of deeper geological structures on the seafloor [5] and complex geographical areas [6], [7]; for haptic exploration of climate maps [8], [9], and improving visualization data from the oil and gas domain [10]–[12]; for cartographic software creation and navigation of blind sailors [13], [14] and for the purposes of personal safety travel around the city and neighborhoods [15], [16], planning and hiking in a national park [17], and so on.

Among other visualization techniques, such as 3D rendering with autostereoscopic and multi-projectors (edge-blended digital dome displays) or multi-touch spherical displays, the haptic component can complement visual information for deeper understanding of traditional geographical maps and exploration of satellite images. Simultaneous activity of vision and touch creates a coherent and robust percept of the virtual objects and a sense of immersion into geographical environment [18]–[20]. Let us consider several examples of relevant studies.

Faeth, Oren and Harding [19] implemented and evaluated the multimodal mesh manipulation system for 3D visualization of geospatial data. Pushing and pulling the tip of Phantom stylus provided haptic information about deformation of the inspected virtual surface.

Haptic Tabletop Puck-device [21], [22] for haptic exploration of geographical maps was implemented and tested in the Interactions Lab at the University of Calgary. The authors presented various types of the terrain by simulating various textures and properties of digital objects such as the height, malleability, and friction. They also displayed ocean temperature through different vibration frequencies.

Chang at MIT Media Lab [23] presented Formchaser device – a single point finger-held mechanism that raised and

The authors gratefully acknowledge the support of Finnish Academy Grant 127774.

lowered index fingertip when the color intensity of the image pixels was changed. The prototype implemented a series of interfaces which allowed the map observers to get a feel of ascent over mountains and immersion into valleys, to sense the waves and ripples on the water surface in a video.

However, both Tabletop Puck and Formchaser had a very limited range of elevation (less than 10 mm) of the prominent part (rod, tip or lever) that should raise and lower the finger, on which it was mounted. The earlier prototypes suffered from technical and usability problems such as bulkiness, residual friction and visual misalignments.

Simonnet with colleagues [13], [14] studied another aspect of alternative visualization of navigation parameters for blind sailors. They developed and evaluated the haptic-auditory navigational instrument "SeaTouch". Haptic exploration of different textures helped to the blind sailors to discriminate the sea, the land, the coastline and the silent objects in virtual maritime environment in the absence of visual feedback.

It is worthwhile to note that most researchers distinguish information related to cutaneous, kinesthetic, and haptic sensory systems [24]. To provide an awareness in peripersonal space, a cutaneous system implies physical contact between objects of interest and the outer surface of the observer's body. The kinesthetic sensory system integrates afferent information originating from the muscles, joints, and skin and efference copy, which enables the brain to evaluate sensory discrepancy resulting from the comparison between the predicted and actual feedback [25-26]. Thus, the kinesthetic sense contributes to the body self-awareness providing information related to the static and dynamic body postures (relative positioning of the head, torso, limbs, and digits) [25], [27]. The haptic system combines both cutaneous, kinesthetic and proprioceptive signals.

The research presented in this paper was addressing a practical question: is it possible to increase the accuracy of the subjective assessment of the local topographic elevation coded by shades of gray and/or color intensity when haptic

feedback, presented as a function of the terrain height, could be associated with values of the light intensity? The accuracy of detecting the topographic heights had been evaluated visually and instrumentally with the new StickGrip haptic device. The topographic heights were randomly selected from the gray scale palette and detected within an assigned geographical region. At that, we hypothesized and sought to confirm that a kinesthetic sense of distance to tablet or/and perception of the finger joint-angle positions [28], can enhance visual accuracy in estimating topographic heights encoded by light intensity.

## II. EXPERIMENTAL SETUP AND PROCEDURE

According to Castleman [29], the human eye can distinguish hundreds of different colors and about 40 shades of gray in a monochrome image. The present experimental research was conducted to evaluate the impact of the haptic component on visual discrimination of the topographic heights associated with an intensity of the grayscale palette.

Eleven different regions of the Earth have been collected from the Google Maps satellite images. After sliding averaging on 5 by 5 pixels, the original color screenshots were processed to convert them into grayscale images having the color reduced to only 20 tones of grayscale, as shown in Fig. 1. Herewith, in order to avoid side effects, such as learning and participants familiarity with a given map, the selected topographic objects had different geographical scale, the palettes comprised of different sets of the shades of gray and each of images was presented only once in two experimental conditions (StickGrip vs. Visual).

Moreover, the map exploration is greatly affected by background information and visual attention distribution. Therefore, before actual exploration of the map elevations we aimed to evaluate the individual baseline sensitivity in more strict conditions using a matching-sorting task.

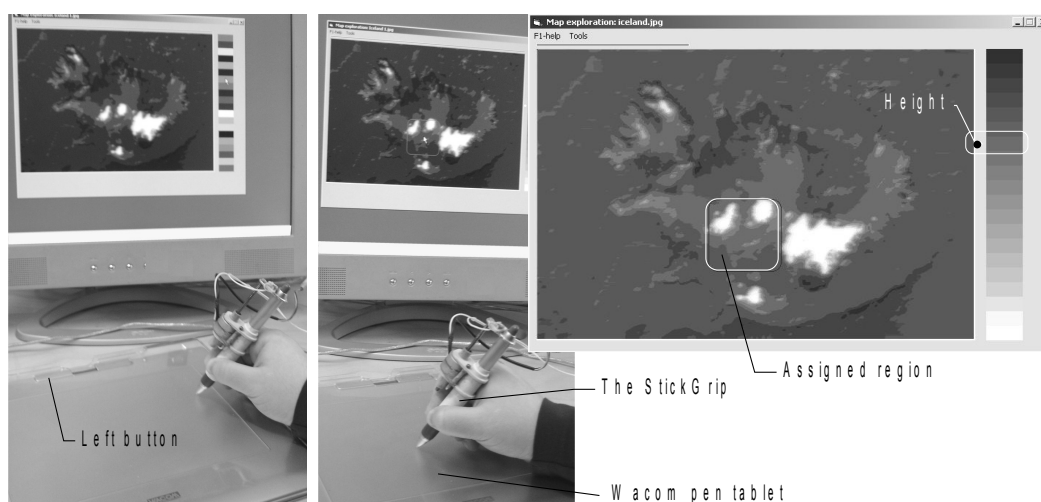


Fig 1. Experimental setup. Testing of the intensity discrimination ability (on the left) and detecting the topographic height within an assigned map region (on the right)

### A. The StickGrip haptic device

The StickGrip haptic device was used to evaluate the topographic elevation haptically. It comprises of the Wacom pen input device added with a motorized penholder as shown in Fig. 1. A point of grasp of the penholder is sliding up and down the shaft of the Wacom pen. When the participants explored the map, they felt as their hand was displaced towards and away from the physical surface of the pen tablet. Distance and direction of the grip displacements were coordinated with visual parameters encoding height of the map regions under inspection [30]. Functionality of the StickGrip was controlled using the pen tablet buttons to activate displacements continuously (the right button) or to complete the task (the left button).

The StickGrip has a range of 40 mm of the total displacement with an accuracy of ( $\pm 0.8$  mm) for the Wacom pen having a length of 140mm coordinated with the intensity of gray levels ranging from 0 to 255. The use of the Portescap 20DAM20D18-L linear stepper motor did not require any additional gears, led to a low noise and equal torque with no differences in directionality that could confound the user. The displacements of the point of grasp in this range ( $\pm 20$  mm) with an average speed of about 15 mm/s give a true feedback about the distance and direction (closer and further) regarding the surface of the pen tablet (or pen tip) and, consequently, such a feedback is a part of the afferent information regarding the local topographical heterogeneity. In our experiments, we also aimed to examine how the new technique for exploration of the topographical maps is accurate and robust.

### B. Pretesting the intensity discrimination ability

During this part of research, we examined the visual ability of the participants to differentiate only 20 levels of gray

which have been used to encode the heights of the satellite images preprocessed with the reduced number of gray tones.

During the first session, the order of intensity levels of the grayscale palette grid was randomized as shown in Fig. 1 (on the left). By placing the darkest row of the palette in the upper position and the lightest one in the bottom position, the participants were asked to rearrange the grid to have a smoother transition between gray tones, as they perceived it.

The task has required from the participants of visual sensitivity to light, self-perception of the finger joint-angle positions [28], attention concentration and patience. Of course, perceptual abilities in such a task cannot be separated from cognitive and behavioral components such as the sorting optimization strategy. In order to reduce the cognitive load and to reveal the perceptual problems, the participants were neither required to minimize the number of permutations nor the time to complete the task. Nevertheless, the perceptual performance was evaluated in terms of the total numbers of permutations, the task completion time and error rate. It was expected that the error rate could indicate the problematic areas of the scale bar palette where the person could not differentiate two or more neighbor intensities.

### C. Detection of the local topographic heights

Immediately after testing the intensity discrimination ability, the participants were asked to perform an exploration of the topographic heights within the map. During this session, each participant had to discover 20 topographic heights randomly selected from the palette grid. Each of height was repeatedly ascertained within 10 randomly assigned regions of the map (the white quadrangle in Fig. 1, on the right). At that, the center of the assigned region was displaced in a random direction from the original height location. An exploration of images was carried out inside the restrained region. In this way we aimed to reduce and align the difficulty of the height detection task in different geographical regions. The

TABLE I.  
BASE-LINE PERCEPTUAL PERFORMANCE IN THE MATCHING-SORTING TASK. THE DATA WERE AVERAGED OVER TEN PARTICIPANTS.

Map of region	Permutations, (SD)		Time, s (SD)		Errors, (SD)	
	StickGrip	Visual	StickGrip	Visual	StickGrip	Visual
Africa	18.6 (1.6)	19.6 (3.2)	52.3 (3.9)	46.5 (9.6)	0.03 (0.1)	2.5 (2.7)
Baycal, RF	20.4 (3.4)	19.7 (2.9)	76.6 (17.4)	47.4 (9.0)	0.2 (0.1)	2.1 (1.7)
Iceland	18.2 (3.3)	18.8 (2.8)	51.3 (3.8)	43.3 (8.7)	0.01 (0.1)	1.7 (2.1)
Japan	21.4 (3.3)	20.7 (2.8)	71.5 (13.3)	46.6 (6.4)	0.02 (0.2)	2.7 (1.0)
Kamchatka, RF	18.2 (2.2)	20.7 (3.2)	50.6 (5.9)	45.8 (6.7)	0.2 (0.1)	2.1 (1.3)
Malaysia	21.7 (1.9)	21.4 (3.1)	78.9 (9.6)	45.9 (6.3)	0.4 (0.1)	2.8 (1.5)
New Zealand	20.0 (1.3)	19.5 (2.0)	60.7 (7.3)	45.2 (4.4)	0.2 (0.1)	2.1 (1.4)
Norway	19.8 (2.6)	21.5 (4.3)	53.7 (6.8)	46.7 (7.9)	0.01 (0.1)	1.9 (1.5)
Panama	19.8 (3.0)	22.0 (5.6)	62.0 (6.2)	45.4 (11.3)	0.01 (0.2)	1.9 (1.1)
Swiss Alps	18.4 (1.8)	20.3 (2.1)	49.5 (7.6)	43.4 (4.4)	0.01 (0.1)	2.5 (5.2)
Turkey	18.4 (1.8)	20.0 (2.8)	53.4 (5.1)	40.3 (5.9)	0.02 (0.1)	0.4 (0.8)
Mean (SD)	19.5 (1.3)	20.4 (1.0)	60.1 (10.9)	45.1 (2.1)	0.1 (0.1)	2.1 (0.7)

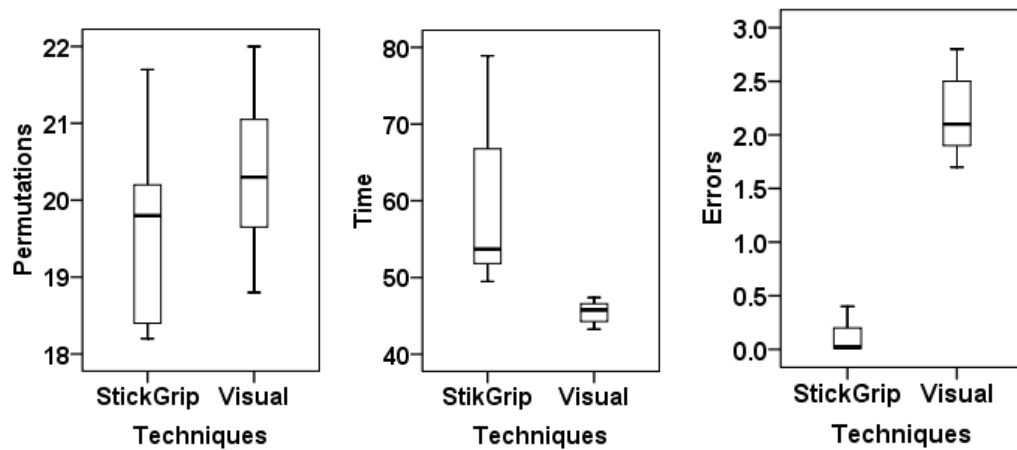


Fig 2. Comparative box plots of the base-line perceptual performance in the matching-sorting task with two techniques of exploration and rearranging of 11 scale bar palettes. The data were averaged over ten participants.

perceptual performance was evaluated in terms of the task completion time and deviation of the local elevation detected from the height assigned within the palette.

#### D. Procedure

In one block of trials (marked as “Visual”), both for testing of the intensity discrimination ability and during an exploration of the topographic heights, the participants relied only on the visual observation of the images on the computer screen and used a regular optical mouse to point at the exact location and to complete the task.

In another block of trials, the participants were asked to use the StickGrip haptic device to have an ability to assess haptically (on demand) the height (intensity) specified in the scale bar palette and the local elevation of the map at selected locations.

During the matching-sorting task, they swapped the corresponding rows of the palette grid by clicking with the left mouse button. During haptic exploration of maps it was required to examine different locations on the tablet without input of any command. Therefore, the participants pressed the left button of the tablet to indicate their decision. With the StickGrip device, the participants perceived haptically the level of shade intensity and could evaluate the difference between neighbor intensity levels of rows of the palette grid or neighbor map locations.

Both conditions (StickGrip vs. Visual) were randomly presented throughout the experiment. The entire test was repeated for 11 different regions of the Earth with no more than three sessions per day.

Detailed verbal instructions were given to the participants regarding the procedure of the experiment. Each of the participants was given an opportunity to refuse the continuation of the experiment at any point without any explanation of the reason. Then an informed consent from each participant about the procedure of the experiment was obtained.

#### E. Participants

Ten volunteers participated in the study (7 males and 3 females). They were unpaid, only beverages were provided. The age of the participants ranged from 21 to 36 years, with a mean age of 26.5. They had normal or corrected-to-normal visual accuracy, and none of them reported sensitive dysfunction in fingers. The participants were right-handed regular computer users and during the test, they used their right hand under both conditions: Visual vs. StickGrip. None of the participants were familiar with experimental setup or were involved earlier in the experiments with haptic feedback.

### III. RESULTS

The results were collected under two conditions: visual observation only, and a situation when the visual observation of the map and the height in the scale bar palette was accompanied with the complementary haptic sense of elevation of the point of grasp of the StickGrip device. The statistical analysis was performed using SPSS 18 for Windows (Chicago, IL).

#### A. Pretesting the intensity discrimination ability

The number of permutations needed to rearrange palettes was averaged over ten participants for each geographical region and presented in Table I. The comparative box plots of the overall data averaged over ten participants are presented in Fig. 2. As can be seen from Table I and Fig. 2, the number of permutations was close to the number of shades within the palette grid, indicating that the participants used a sub-optimal strategy. The comparative box plots demonstrated that the participants spent significantly more time when they used the StickGrip device but there were few accidental errors. In the absence of haptic feedback, a significant increase of errors was systematically recorded.

When the participants used the StickGrip device, the average number of permutations was of about 19.5 with a standard deviation (SD) of 1.3, varying from 18.2 (SD=2.2) for

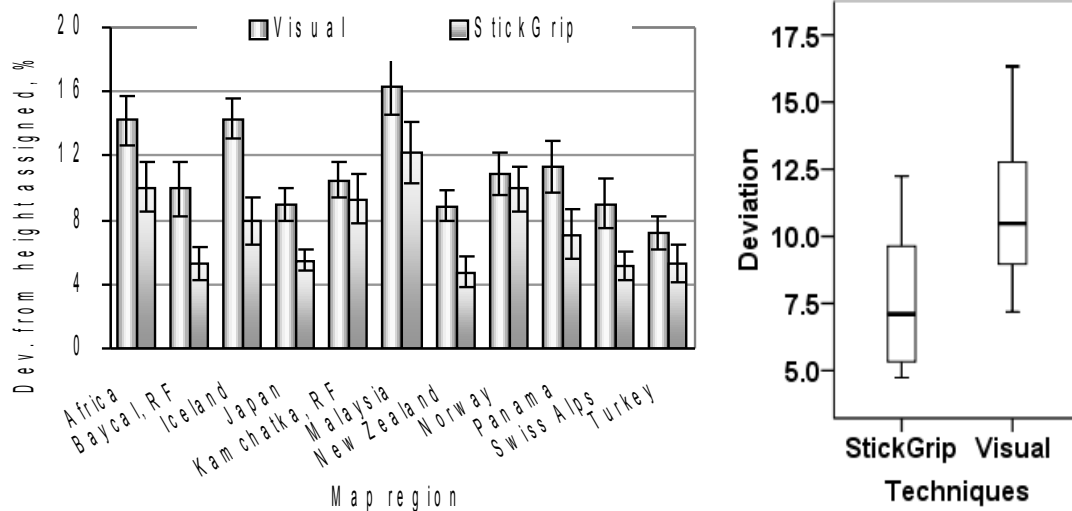


Fig 3. The deviation of the local elevation detected from the height assigned within the palette. The data were averaged over ten participants under two conditions of exploration of eleven map regions.

the palette of shades collected from the Kamchatka region to 21.7 (SD=1.9) related to the palette of Malaysia. The number of permutations required to rearrange palettes relying only on the visual information and using a regular mouse to swap the rows of the palette varied from 18.8 (SD=2.8) associated with the palette of Iceland region to 22.0 (SD=5.6) representative for the palette of Panama region with a mean of about 20.4 (SD=1.0). The paired samples t-test revealed a small but statistically significant difference between the number of permutations performed under two conditions (StickGrip vs. Visual):  $t(10)=2.292$  ( $p<0.05$ ), at that, the correlation of this parameter was about 0.437 and not significant ( $p>0.01$ ).

When the participants used the StickGrip device, the average task completion time changed from 49.5 s (SD=7.6 s) in a case of palette associated with the Swiss Alps region to 78.9 s (SD=9.6 s) for the palette related to the Malaysia re-

gion, with a mean of about 60.1 s (SD=10.9 s). The visual condition of rearranging rows of the maps' palettes (Table I) demonstrated that an average task completion time ranged from 40.3 s (SD=5.9 s) for the map of Turkey region to 47.4s (SD=9.0 s) for the palette of the Baycal map, with a mean of about 45.1 s (SD=2.1 s).

The results of the paired-sample t-test indicated that the difference in perceptual performance assessed by the parameter of the task completion time with two different techniques (StickGrip vs. Visual) was significant:  $t(10) = 4.917$  ( $p< 0.05$ ), while the correlation index was low and not significant 0.482 ( $p>0.01$ ).

The analysis of error rates under two conditions (StickGrip vs. Visual) of rearranging the color palettes showed that with the use of the StickGrip the mean of the number of errors ranged from 0.01 to 0.4 with a mean of about 0.1 (SD=0.1).

TABLE II.  
 PAIRED DIFFERENCES BETWEEN ASSIGNED AND SELECTED TWENTY HEIGHTS AND THE TASK COMPLETION TIMES UNDER TWO CONDITIONS (STICKGRIP VS. VISUAL) THE DATA WERE AVERAGED OVER TEN PARTICIPANTS

Map of region	Heights (intensity levels)		Time, s	
	Corr., Sig.	t(df=19), Sig.	Corr., Sig.	t(df=19), Sig.
Africa	0.981 p<0.001	9.0 p<0.005	0.133 p>0.5	8.92 p<0.0001
Baycal, RF	0.893 p<0.001	4.27 p<0.001	0.121 p>0.5	11.84 p<0.0001
Iceland	0.928 p<0.001	4.13 p<0.001	0.019 p>0.5	11.71 p<0.0001
Japan	0.976 p<0.001	0.03 p>0.5	0.291 p>0.5	10.92 p<0.0001
Kamchatka, RF	0.880 p<0.001	2.75 p>0.01	0.391 p>0.5	11.98 p<0.0001
Malaysia	0.962 p<0.001	2.06 p<0.05	0.435 p>0.5	10.36 p<0.0001
New Zealand	0.956 p<0.001	3.68 p<0.05	0.025 p>0.5	15.70 p<0.0001
Norway	0.940 p<0.001	3.94 p<0.001	0.482 p>0.5	12.27 p<0.0001
Panama	0.935 p<0.001	3.99 p<0.001	0.127 p>0.5	20.50 p<0.0001
Swiss Alps	0.893 p<0.001	2.64 p>0.01	0.562 p>0.5	14.38 p<0.0001
Turkey	0.937 p<0.001	3.44 p<0.005	0.162 p>0.5	13.19 p<0.0001

The average number of error rate for the visual condition varied from 0.4 (SD=0.8) (the map of Turkey region) to 2.8 (SD=1.5) instances (the Malaysia region) with a mean of about 2.1 (SD=0.7). In most cases, the errors committed were recorded when the participants had to compare the darkest rows of the palette.

The results of the paired samples t-test revealed that the participants committed significantly more errors when the matching-sorting task was performed in the absence of haptic feedback,  $t(10) = 10.556$  ( $p < 0.001$ ). Correlation of error rates under two conditions over different regions was very low and not significant 0.351 ( $p > 0.01$ ).

#### B. Detection of the local topographic heights

Providing the visual matching task with relevant haptic information demonstrated that deviation of the detected local elevation from the height assigned varied in different geographical maps (Fig. 3) from a minimum of 4.7% with a standard deviation (SD) of 1.9% to a maximum of 12.2% (SD=3.8%) with an *average of about 7.5%* (SD=2.6%).

The visual condition of observing the map region and the height assigned demonstrated that the deviation of elevation values detected ranged from a minimum of 7.2 % (SD = 2.0%) for the map of Turkey region to a maximum of 16.3% (SD= 3.4%) for the map of Malaysia with a *mean of about 11.0%* (SD=2.7%).

The results of the paired-sample t-test of deviation of the local elevation from the height specified in the scale bar palette indicated that adding the complementary haptic sense significantly increased an accuracy of estimating the height coded by shades of gray, by about 32% [in particular,  $(11.04-7.51)/11.04 \cdot 100\% = 31.97\%$ ],  $t(10) = 7.31$  ( $p < 0.000$ ). The index of correlation between the data collected under two conditions (StickGrip vs. Visual) was positive 0.824 and significant  $p < 0.005$ . The correlation indicated that the differences in human performance were mostly observed due to the different exploration conditions and, in a less extent, due to differences in the satellite images.

It was also reported by the participants that matching task was clear and helped them to estimate the benefits of the StickGrip device in distinguishing two gray tones with vanishing difference of intensity. After acquiring some experience in the use of the StickGrip device during pretesting, detection of the local topographic heights did not cause any problems. The experimental data indicated that relying on complementary haptic feedback the participants were able to assess a subtle difference between the assigned heights and elevations of the selected locations within a specified region of the map.

As can be seen from Table II, the elevations detected in all geographical maps under two conditions (StickGrip vs. Visual) were highly and significantly correlated with the heights assigned within palettes. The correlation varied from a minimum of 0.880 ( $p < 0.001$ ) to a maximum of 0.981 ( $p < 0.001$ ). The paired-sample t-test for the data averaged over ten participants revealed a significant difference in accuracy of de-

tection of the assigned values under two experimental conditions (StickGrip vs. Visual) in eight of eleven map regions. The differences in accuracy varied significantly from a minimum of  $t(19) = 2.06$  ( $p < 0.05$ ) to a maximum of  $t(19) = 9.0$  ( $p < 0.005$ ). Only in three regions (Japan, Kamchatka and Swiss Alps) the differences in accuracy of height detection were low and not significant.

The index of correlation of the time spent to complete the perceptual matching task under two conditions (StickGrip vs. Visual) was low and not significant varying from a minimum of 0.019 ( $p > 0.5$ ) to a maximum of 0.562 ( $p > 0.5$ ). The paired-sample t-test for the data averaged over ten participants (for 20 heights) revealed that the differences in completion time under two conditions (StickGrip vs. Visual) varied from a minimum of  $t(19) = 8.92$  ( $p < 0.0001$ ) to a maximum of  $t(19) = 20.50$  ( $p < 0.0001$ ).

#### IV. CONCLUSION

The goal of the research discussed in this paper was to evaluate the accuracy of detecting the topographic heights visually and instrumentally with the StickGrip haptic device. Eleven different regions of the Earth have been collected from the Google Maps satellite images. The original color screenshots were preprocessed to convert them into grayscale images having the limited number of intensity levels of twenty tones.

By performing the matching-sorting task, during the baseline experiments the participants were examined for their ability of the light intensity discrimination. In the second session, the participants explored the map heterogeneity to detect the height randomly selected from the scale bar palette. Both experiments have required from the participants of visual sensitivity to light, self-perception of the finger joint-angle positions, attention concentration and patience.

The results of the paired-sample t-test revealed that the participants committed significantly more errors in the baseline experiments when they performed the task in the absence of haptic feedback, the difference was high and significant  $t(10) = 10.556$  ( $p < 0.001$ ). The experimental data collected in the second session of the map exploration indicated that relying on the complementary haptic feedback the participants were able to assess a subtle difference between the assigned heights and elevations of the selected locations in different map regions. The results of the paired samples t-test of deviation of the local elevation from the height specified indicated that adding the complementary haptic sense increased an accuracy of estimating the height by about 32%. The difference between two conditions was high and significant  $t(10) = 7.31$  ( $p < 0.000$ ). The benefits of instrumental support for human performance are evident.

We hypothesized and confirmed that a kinesthetic sense of distance to tablet or/and perception of the finger joint-angle positions, can enhance visual accuracy in estimating topographic heights coded by shades of gray. It was also demonstrated that untrained participants can accurately detect geographic locations with necessary height when values of the



grayscale intensity were complemented with haptic feedback presented as a function of elevation.

In further research, we plan to examine more complex scenario of interaction with cartographic information by exploring topographic surfaces with different types of discontinuity and textures. The StickGrip device can be considered as a robust tool having the potential for everyday work with graphic editors to engineers, architects, interior designers and ordinary users.

#### REFERENCES

- [1] J. T. Bjorke, K. Saeheim, "Investigation of the Channel Capacity of Seafloor Maps with Colored Depth Intervals," in *Proc. 11th Scandinavian Research Conf. on Geographical Information Science*, Norway, 2007, pp. 61–73.
- [2] E. Chesneau, "Improvement of Color Contrasts in Maps: Application to Risk Maps," in *Proc. 10th AGILE Int. Conf. on Geographic Information Science*, Denmark, 2007, pp.1-14.
- [3] K. Moreland, "Diverging Color Maps for Scientific Visualization," in *Proc. 5th Int. Symposium on Visual Computing, ISVC 2009*, Part II, LNCS 5876, Springer-Verlag, 2009, pp 92-103.
- [4] B. E. Rogowitz, L. A. Treinish, S. Bryson, "How Not to Lie with Visualization," *Computers in Physics J.*, vol. 10, pp. 268–273, 1996.
- [5] C. Harding, I. A. Kakadiaris, J. F. Casey, R. B. Loftin, "A Multisensory System for the Investigation of Geoscientific Data," *Computers & Graphics J.*, vol. 26, no.2, pp. 259–269, 2002.
- [6] De F. Felice, F. Renna, G. Attolico, A. Distante, "A Haptic/Acoustic Application to Allow Blind the Access to Spatial Information," in *Proc. 2th Joint EuroHaptics Conf. and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems WHC'07*, Washington, DC, USA, 2007, pp. 310 – 315.
- [7] K. S. Papadopoulos, "On the Theoretical Basis of Tactile Cartography for the Haptic Transformation of Historic Maps," *e-Perimtron J*, vol. 1, no. 1, pp. 81-87, 2005.
- [8] C. Lee, B. D. Adelstein, S. Choi, "Haptic Weather," in *Proc. 16th IEEE Symp. Haptic Interfaces for Virtual Environment and Teleoperator Systems, HAPTICS'08*, Reno, NV, USA, 2008, pp. 473-474.
- [9] N. Yannier, C. Basdogan, S. Tasiran, O.L. Sen, "Using Haptics to Convey Cause-And-Effect Relations in Climate Visualization," *IEEE Transactions on Haptics '08*, vol. 1,no. 2, pp. 130–141, 2008.
- [10] W. A. Aviles, J. F. Ranta, "Haptic Interaction with Geoscientific Data." in: The Fourth PHANTOM Users Group Workshop, AI Lab Technical Report No. 1675 and RLE Technical Report N 633, MIT, 1999, pp. 78-81.
- [11] B. Fröhlich, S. Barass, B. Zehner, J. Plate, M. Göbel, "Exploring Geoscientific Data in Virtual Environments," in *Proc. IEEE Visualization '99*, Washington, DC, USA, 1999, pp. 169–173.
- [12] C. Harding, B. Loftin, A. Anderson, "Visualization and Modeling of Geoscientific Data on the Interactive Workbench," *The Leading Edge J.*, vol. 19, no. 5, pp. 506–511, 2000.
- [13] M. Simonnet, J-Y. Guinard, J. Tisseau, "Preliminary Work for Vocal and Haptic Navigation Software for Blind Sailors," in *Proc. 6th ICDVRAT'06*, University of Reading, UK, 2006, pp. 255–262.
- [14] M. Simonnet, R. D. Jacobson, S. Vieilledent, J. Tisseau, "Can Virtual Reality Provide Digital Maps to Blind Sailors? A Case Study," in *Proc. Int. Conf. of Cartography ICC'09*, Santiago, 2009, 10 p.
- [15] S. Landau, E. Bourquin, J. Miele, A. J. Van Schaack, "Demonstration of a Universally Accessible Audio-Haptic Transit Map Built on a Digital Pen-Based Platform," in *Proc. 3-d Int. Workshop on Haptic and Audio Interaction Design*, Jyväskylä, Finland, 2008, pp. 23-24.
- [16] Y. Murai, H. Tatsumi, N. Nagai, M. Miyakawa, "A Haptic Interface for an Indoor-Walk-Guide Simulator," *ICCHP'06*, Springer-Verlag, 2006, pp. 1287-1293.
- [17] C. Magnusson, K. Tollmar, S. Brewster, T. Sarjakoski, L. T. Sarjakoski, S. Roselier, "Exploring Future Challenges for Haptic, Audio and Visual Interfaces for Mobile Maps and Location Based Services," in *Proc. 2nd Int. Workshop on Location and the Web CHI2009*, vol. 370, no. 8, New York, 2009, pp. 1-4.
- [18] M. Ernst, H. Bulthoff, "Merging the Senses into a Robust Percept," *Trends in Cognitive Science J.*, vol. 8, no.4, pp. 162–169, 2004.
- [19] A. Faeth, M. Oren, C. Harding, "Combining 3-D Geovisualization with Force Feedback Driven User Interaction," in *Proc. 16th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems*, Irvine, CA, 2008, Art. 25.
- [20] S. Kibria, "Functionalities of Geo-Virtual Environments to Visualize Urban Projects," M.Sc. Thesis GIMA 2008, Utrecht Univ., TU Delft, Wageningen Univ., ITC, 2008.
- [21] N. Marquardt, M. A. Nacenta, J. E. Young, S. Carpendale, S. Greenberg, E. Sharlin, "The Haptic Tabletop Puck: Tactile Feedback for Interactive Tabletops," report 2009-936-15, Dept. of Computer Science, Univ. of Calgary, Calgary, AB, Canada T2N 1N4. 2009.
- [22] N. Marquardt, M. A. Nacenta, J. E. Young, S. Carpendale, S. Greenberg, E. Sharlin, "The Haptic Tabletop Puck: Tactile Feedback for Interactive Tabletops," in *Proc. ACM Int. Conf. on Interactive Tabletops and Surfaces, ITS '09*, Nov. 23-25, 2009, Banff, Alberta, Canada, 2009.
- [23] A. Chang, J. Gouldstone, J. Zigelbaum, H. Ishii, "Pragmatic haptics," in *Proc. TEI '08*. ACM, 2008, pp. 251-254.
- [24] R. S. Dahiya, G. Metta, M. Valle and G. Sandini, "Tactile Sensing—From Humans to Humanoids," in *IEEE Transactions on Robotics*, vol. 26, no 1, pp. 1-20, Feb. 2010.
- [25] J. M. Loomis, S. J. Lederman, "Cognitive processes performances," in *Tactual Perception* vol. 2, in K. R. Boff, L. Kaufman, & J. P. Thomas (Eds.) Handbook of Perception and Human Performances Series. New York: Wiley, 1986, pp 31/1-31/41.
- [26] M. S. A. Graziano and M. M. Botvinick, "How the brain represents the body: Insights from neurophysiology and psychology," in *Common Mechanisms in Perception and Action: Attention and Performance*, W. Prinz and B. Hommel (Eds.) London, U.K.: Oxford Univ. Press, 2002, pp. 136–157.
- [27] R. L. Klatzky and S. J. Lederman, "Touch," in *Experimental Psychology* vol. 4, in A. F. Healy and R. W. Proctor (Eds.) Handbook of Psychology Series, New York: Wiley, 2003, pp. 147–176.
- [28] H.Z. Tan, M.A. Srinivasan, C.M. Reed, N.I. Durlach, "Discrimination and Identification of Finger Joint-Angle Position Using Active Motion," *ACM Transactions on Applied Perception J.*, vol.4, no 2, Article no 10, July 2007, ACM New York, 2007.
- [29] K. R. Castleman, "Digital Image Processing," Prentice-Hall, Englewood Cliffs NJ, 1996.
- [30] G. Evreinov, T.V. Evreinova, R.Raisamo, "Method, Computer Program and Device for Interacting with a Computer," Finland Patent Application, G06F ID 20090434, 2009.



# Image Indexing by Spatial Relationships between Salient Objects

Eugen Ganea  
University of Craiova  
Bd. Decebal 107, Romania,  
Email: eganea@software.ucv.ro

Marius Brezovan  
University of Craiova  
Bd. Decebal 107, Romania,  
Email: mbrezovan@software.ucv.ro

**Abstract**—In this paper, we presented our technique to extract and to use the spatial relationships between two or more salient objects. Using an object oriented hypergraph data structure, the spatial relationships are determined and stored in an object-oriented database. This work aims to unified the phases of processing, indexing and retrieval of images. The proposed model can be applicable to other types of data (video) and to semantic relations hidden in an image. The structure of the database used for image storage allow the construction of the indexes classes hierarchy in order to improve the results of image retrieval. Our method requires more experiments for datasets which come from different areas and for images which contain more salient objects.

## I. INTRODUCTION

GROWTH rate for multimedia information (image, audio, video, text) involves new approaches to the problem of information retrieval. This paper describes research done to build an object-oriented database that allows storage and content-based querying of images. The main goal of this work is spatial relationships-based indexing and querying of images using hypergraph data structure. The main steps needed to achieve this goal are (I) object-oriented approach to image processing; (II) use the syntactic features of the salient objects for spatial relationships determination; (III) indexing techniques for efficient content-based image retrieval with an object-oriented database. The technique proposed uses hypergraph data structures to determine and to store the spatial relationships between two or more salient objects. In the graph approach, the edges stores information about spatial relationships between only two neighboring objects. Using a hypergraph, we represent relationships among several salient objects detected in the image processing phase. In [1] was described a hypergraph-based image representation that considered Image Adaptive Neighborhood Hypergraph (*IANH*) model. Our object oriented model is based on an initial hexagonal representation of an image and hypergraph structure is constructed on this representation. This method allows to assign semantics to an image and the developing of content-based query language for image retrieval. The object-oriented database is implemented on top of *HyperGraphDB* (*HGDB*) [2]. The presentation will focus on salient objects detection, object-oriented image modeling of spatial relationships, query language and image indexing. Section II presents the technique for salient objects detection and the algorithms

for determination spatial relationships between objects. Section III describes our proposed method for representation, storing, indexing and image retrieval. Section IV gives our experimental results and Section V concludes the paper.

### A. Related Work

The concepts of spatial relationships can be grouped into three categories [3]: orientation relationships, distance relationships and topological relationships. The orientation relationships describe the relations between two spatial positions of visual objects. These relationships are established according to two basic concepts: salient object and reference object. Cohn et al. [4] provides a spatial distribution in terms of space used entities (regions) and ways of describing the spatial relationships between these entities (topological distance or direction). In [5] was presented a method to identify relations between two salient objects in images. They used spatial relationships to homogenize, reduce and optimize the representation of relationships with the 9-Intersection model proposed by [6]. An other approach,  $\Delta$ -*TSR* (Triangle Spatial Relationships) [7] is used for similarity search in image database. In this research, image descriptions is based on co-occurrences of triplets of objects whose geometric relationships are encoded using the angles of the triangle formed by the objects. All these descriptions are invariant to rotation in 2D, translation or scaling of the image. An extension of  $\Delta$ -*TSR* is given in [8], where a  $B+$  tree is constructed for all images by using the low level color feature and create multidimensional  $B+$  tree by combining it with the  $B+$  tree that we made it for symbolic images by using *TSR*. In *DISIMA* project [9], the spatial relationships are modeled using for each salient object, the minimum bounding rectangle (*MBR*) whose projections are taken on the  $ox$  and  $oy$  axes. As indexing technique, are used  $2D-h$  tree [10], which is based also on  $B+$  tree. In [11] are presented the main fuzzy approaches to define spatial relationships and comparative discussions. Such approaches are useful to express the linguistic terms as: *close*, *far*, *very close* and *very far*. The problem of all these researches which are based on  $2-D$  string representation is failure to take into account the all spatial relationships between objects (for example inclusion) and reference to pairs of neighboring salient objects. In this paper, these shortcomings have been overcome by using structures such as object oriented hypergraph.

## II. OBJECT ORIENTED MODEL FOR REPRESENTATION OF SPATIAL RELATIONSHIPS

The introduction of the hypergraph structure was in [12] and is a generalization of graph theory. The main idea refers to consideration of sets of edges and then a hypergraph is the family of these edges (called hyperedges). For the segmentation phase of the image processing process, we use the hypergraph structure and a dual representation of images, as in Figure 1, hypergraph and forest tree.

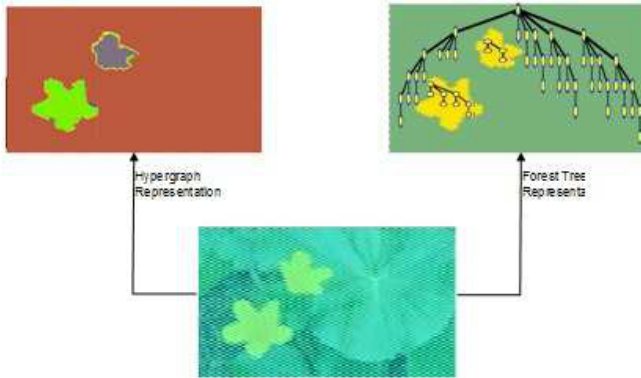


Fig. 1. Dual representation (hypergraph/forest tree) of image

This approach allows the implementation of the algorithm for minimum spanning tree for image segmentation [13] and construction, as a result of segmentation, of hypergraph which is used for all steps of image processing. The hypergraph data model and object-oriented model make join to define the spatial relations between regions from images and their features; the composite model is called hypergraph object-oriented model [14]. Depending on hypergraph structure the complex spatial relations can be described easily and the attributive data can also be integrated more efficiently. The object-oriented model allows to define the methods by which messages are exchanged between objects and to implement the inheritance mechanism which offers classes which have new definitions based on existing definitions.

### A. Image Segmentation and Image Annotation

Useful information available after segmentation phase (syntactic information) are correlated with semantic data. These category of information are stored using object-oriented hypergraph (*HOOG*), that results from the process of segmentation and annotation. Semantic information of each segmented region are assigned through a semi-automatic procedure for images that represent the set of training images, and for the rest of the images processed using an inference algorithm which is based on decision tree [15]. The structure of the *HOOG* is based on instances of a set of classes that described the syntax and the semantic of image. Each class has three specific parts: a description of the list of attributes for each object, a semantic of object, and spatial relationships which give links between objects. The set of classes consists of two hierarchies: the first

corresponds to data extracted from the training set of images resulted after the segmentation and manual annotation, and it is specific to the domain from which images came. The second hierarchy of class is generated automatically using attribute graph grammars results of the inference algorithm that takes as input the corresponding hypergraphs to the representative images of the domain. In both hierarchies, classes are divided into two categories: classes which refer to images syntax, respectively classes that relate to their semantics.

### B. Spatial Relationships Determination

The first category of spatial relationships is used to describe the position of an object in image; representation often used to describe the absolute positions which are the cardinal points: north (*N*), south (*S*), east (*E*), west (*W*), north-east (*NE*), south-east (*SE*), north-west (*NW*) and south-west (*SW*). The space allocation of a 2-D image is divided it into nine zones, each zone corresponds to a cardinal point and middle zone is the center of the image. Each salient object is represented by the minimum bounding rectangle and center of gravity. Absolute position is determined by establishing membership of gravity center to the nine areas described above. The relative positions are calculated by comparing the coordinates of centers and frontier of the bounding rectangles of two neighbors salient objects. In a similar mode, that shown in Figure 2 for a spatial relationship of type *left/right* are determined all relative spatial relationships.

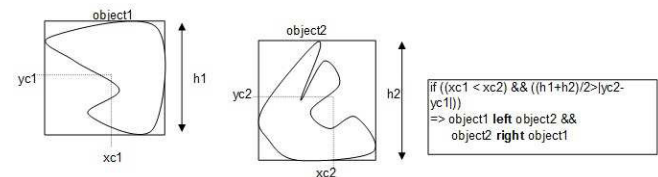


Fig. 2. Spatial relationship type left/right

Distance spatial relationships imply distance concepts between objects. As in the orientation relationships, in determining distance relationships are necessary the three basic concepts: the reference, the primary object and the reference object. Spatial relationships concepts specified above can be put in correspondence with pairs of salient objects, to determine whether or not they check those concepts. For this step, geometric inference, we use object-oriented hypergraph structure because can be model for semantic of each spatial relationship. For object-oriented hypergraph we propose the implementation of dynamic semantics of their through an algorithm similar to the *RETE* algorithm from expert systems. After the processing phase (segmentation and annotation), each salient object resulted, *O* has the following information attached: dominant color,  $color(O)$ ; hexagons list that forms the area of the object,  $la(O)$ ; ordered list of hexagons that forms the border of the object,  $lc(O)$  and  $semantic(O)$ . On the basis of lists  $lc(O)$  and  $la(O)$  are determined other attributes: the area of object,  $area(O)$ ; the perimeter of the

object contour,  $perimeter(O)$ ; the center of gravity,  $g(O)$ ; eccentricity,  $e(O)$ ; compactness,  $comp(O)$ ; the minimum and the maximum value of  $X$  coordinate of the pixel of the object,  $x-min(O)$  and  $x-max(O)$ ; the minimum and the maximum value of  $Y$  coordinate of the pixel of the object,  $y-min(O)$  and  $y-max(O)$ . In addition, for two each pair of neighbors objects,  $O1$  and  $O2$ , the algorithm of segmentation determines the common part of the contour,  $cb(O1, O2)$ . All these salient geometric features are represented by the group of local geometric descriptors that correspond to neighbors objects. In this stage of extraction of spatial characteristics is generated a dual hypergraph by creating hyperedge that store the relationships by neighbors objects. The result of processing algorithm is stored as a hypergraph whose nodes are represented detected areas and through the hyperedges are represented neighborhood relationships. The geometric properties specific to each detected object are available from the segmentation phase and are used as input for the algorithm of the dual hypergraph construction. Dual hypergraph structure can be defined as:  $HGs = (HNs, HEs)$ , where  $HNs$  is the set of nodes created to show the spatial relationships by neighbors objects, and  $HEs$  is the set of hyperedges which makes connection between nodes with spatial information. The number of newly added nodes (spatial nodes) is equal to the number of hyperedges from the hypergraph obtained after segmentation image. The algorithm 1 is used to determine dual hypergraph.

The link between the initial hypergraph and the hypergraph corresponding to the spatial relationship is done through  $HE$  hyperedges that are referenced by  $HNs$  nodes. Dual hypergraph thus determined is used to complete the structure of indexes used in the indexing information in the database. Based on the information stored at each node of initial hypergraph ( $HG$ ), the edges of dual hypergraph ( $HGs$ ) are decorated with labels that are specifying the spatial relationship. For each topological relationship is using a rule based on a salient feature that enables the geometric relationship. Rules have been written using the *CLIPS* language, which is a forward chaining rules-based system, built on the *Rete* algorithm and which also supports the object-oriented programming paradigm - *COOL* (CLIPS Object-Oriented Language). To determine the relations *Inside/Outside*, *Above/Bellow* and *Left/Right* were defined the following rules that are using geometric feature value corresponding to the common border of two objects  $o1$  and  $o2$ :

### III. IMAGE STORING AND REPRESENTATION USING OBJECT ORIENTED DATABASE

This section presents the structure of object-oriented database that is used to store and query semantic information obtained by *2D* image processing. Object-oriented approach for representing information related to image processing provides a direct binding with object-oriented database scheme that will store combinations by the objects corresponding to the processed images. The choice of object oriented model for representing images is based on two arguments: first

---

**Algorithm 1:** The algorithm for constructing dual hypergraph

---

**Input:** Segmented and annotated image hypergraph  
 $HG=(HN, HE)$

**Output:** Dual hypergraph  $HGs$

```

1 Procedure dualHGConstruction ( $HG; HGs$ );
2 * initialize  $HNs$ ; initialize  $HEs$ ;
3 for  $i \leftarrow 1$  to  $sizeof(HN)$  do
4    $hn\_i \leftarrow$  node  $i$  from  $HN$ 
5   * determine the list of hyperedges  $HE\_i$  for  $hn\_i$ 
   from  $HE$ 
6   * reset the list of nodes  $HN\_ij$ 
7   for  $j \leftarrow 1$  to  $sizeof(HE\_i)$  do
8      $he\_j \leftarrow$  hyperedge  $j$  from  $HE\_i$ 
9     if  $!find(he\_j, HNs)$  then
10      * create node  $hn\_ij$ 
11      * add  $hn\_ij$  to  $HNs$ 
12    end
13    else
14       $hn\_ij \leftarrow$  find ( $he\_j, HNs$ )
15    end
16    * add  $hn\_ij$  to  $HN\_ij$ 
17  end
18  for  $j \leftarrow 1$  to  $sizeof(HN\_ij)$  do
19    for  $k \leftarrow j + 1$  to  $sizeof(HN\_ij)$  do
20      * add hyperedge ( $hn\_ij, hn\_ik$ ) to  $HEs$ 
21    end
22  end
23 end
```

---

argument relates to the separation of structure of a class by its content, and the second refers to the specificity of each image that does not allow an implicit modeling using predefined data structures such as those used in relational databases. Relational model has several limitations in the representation of complex objects corresponding to an image. Looking on representation of data in relational databases, links between the two relationships are represented through foreign key type attributes in a relationship that refer to primary key type attributes in another relationship. Tuples that have the same values for foreign keys, respectively primary, those are logically linked, although they are not physically associated (logical references). In case of object model, relationships are represented by reference through an object identifier (*OID* - Object Identifier) which provides a structural association of tuples. On the other hand, object-oriented model, unlike the relational model, supports complex object structures with the possibility of using sets, lists or other complex data structures [16]. It also allows defining the methods by which messages are exchanged between objects and implements the inheritance mechanism that provides the possibility to have new classes definitions based on existing definitions. Given the complexity and variety of objects that may be finding within the scene of an image processed and the properties of object model above

```

(defrule Inside_r1_r2
  (= perim (r1) common_perim)
  =>
  ins (r1, r2)
)
(defrule Outside_r1_r2
  (<> perim (r1) common_perim)
  =>
  ots (r1, r2)
)
(defrule BelowAbove_r1_r2
  (< yc1 yc2)
  (> (w1+w2) |xc1 -xc2|)
  =>
  bhv (r1, r2)
  abv (r2, r1)
)
(defrule LeftRight_r1_r2
  (< xc1 xc2)
  (> (h1+h2) |yc1 -yc2|)
  =>
  lft (r1, r2)
  rgt (r2, r1)
)

(defrule Inside_r2_r1
  (= perim (r2) common_perim)
  =>
  ins (r2, r1)
)
(defrule Outside_r2_r1
  (<> perim (r2) common_perim)
  =>
  ots (r2, r1)
)
(defrule BelowAbove_r2_r1
  (< yc2 yc1)
  (> (w1+w2) |xc1 -xc2|)
  =>
  bhv (r2, r1)
  abv (r1, r2)
)
(defrule LeftRight_r2_r1
  (< xc2 xc1)
  (> (h1+h2) |yc1 -yc2|)
  =>
  lft (r2, r1)
  rgt (r1, r2)
)
    
```

Fig. 3. Rules for determining spatial relationships

mentioned, in the paper was chosen the object-oriented model. Object-oriented modeling involves using instances of classes that are defined above and whose properties are defined by attributes and the communication between them is done through exchanging messages implemented by methods of classes. This approach of problems, based on algebraic, for collections of objects and relationships between them was taken by many researchers in studies and conducted implementations. The implementation from [17],  $\Lambda$ -DB is a database management system based on object-oriented database (OODB) based on standard ODMG3.0 [18]. In [19] was introduces a model for OODB representation (GOOD - Graph-Oriented Object Database) that is based on a graph data structure, operations on database objects translate into the transformation of the graph. Taking the approach used for image processing (segmentation and annotating), we use as kernel the database *HyperGraphDB*. The set of definitions of classes whose instances are intended to be serialized by the database compose the schema of OODB. In our case, the schema is divided in two hierarchy: the classes which refer the syntactic characteristics of images, respectively classes that relate to their semantics. The syntactic schema are independent from the image domain and is presented in Figure 4.

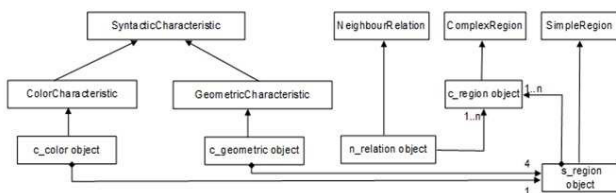


Fig. 4. Syntactic Schema for Object-Oriented Database

In the database there are two types of groups: at image level, respectively, at salient objects level. The first group contains information extracted from the whole image and allows queries that can be expressed as follows: "Find all images that have the same color with the specified image". The grouping of relevant objects level allows queries of the form: "Find all images that contain objects  $O_1$ ,  $O_2$  and between  $O_1$  and  $O_2$  exists spatial relationship  $R$ ". For both types of queries we need a semantic schema. An example of semantic schema to represent the information of images which came from the soccer domain is presented in Figure 5.

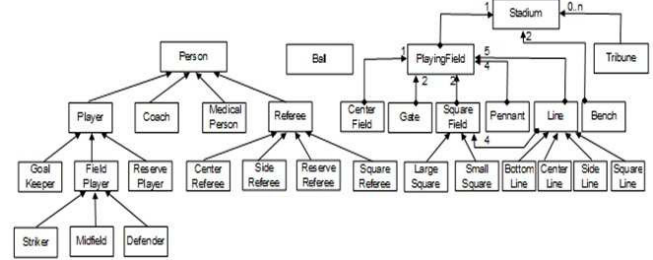


Fig. 5. Example of Semantic Schema for Object-Oriented Database

A. Image Indexing and Image Retrieval

Indexing problem is approached using graph theory, the indexing relationship is represented by indexes allocated within classes and forming a directed graph. Other approaches [20] refer to the database schema, thus the necessary time for the selection of optimum index is of high complexity. Based on the database scheme, developed a new approach to the problem of indexing by exploiting the graphical structure. Index information are used as the *OID* related to the objects in the database; we use as index information, the spatial relationships of salient objects. In figure 6 is presented the structure of node which corresponds to a region. The field *active* is a

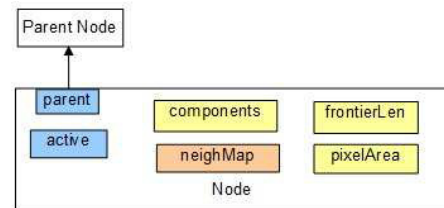


Fig. 6. Indexing Node Structure

boolean value specifying if the corresponding region is the root of a tree representing a salient object. The rest of fields are used only if the field *active* is true. The attribute *parent* represents the index of the salient object which is the parent of the current region. The field *components* is a list of indices of the pixels belonging to the associated region. The attributes *frontierLen* and *pixelArea* represent the length of contour region and respectively the value of area region. The field



*neighMap* is a *HashMap* instance which store all the spatial relationships between current region and all neighbors regions. The elements of the attribute *neighMap* are extracted from dual hypergraph determined with the algorithm 1. The nodes are organized in a tree with links *up* and *spatialRel*; the reference for *up* link is given by *parent* attribute and the *spatialRel* references are given by instances of the hyperedges objects from dual hypergraph. This dual representation of data, through a tree and through an hypergraph, allows the construction of dual indexes. The first type of indexes refers to the runtime necessary query indexes and is based on tree structure. The second group of indexes are the indexes used by the *OODB* and is based on hypergraph structure. Choosing hypergraph to represent indexes was made because this type of structure is a good framework for query processing and the information retrieval corresponding to processed images.

Image retrieval systems have been developed using a variety of technologies based on various disciplines of computer science. The development of new technologies has emerged the possibility of improving existing retrieval systems. Thus, is the case of using concepts of object-oriented programming for recognition objects. Using the object model for storing images is based on complex and different structure of each image that does not allow a simple data model that uses predefined data structures such as those used in relational databases. We develops an interpreter that translate the semantic queries based on symbolic language in *SPARQL* (Protocol and *RDF* Query Language) query language [21]. To define the symbolic language we use as the lexical atoms, the concepts of domain corresponding to the annotation process and the elements which corresponds to the spatial relationships. The interpreter generate *SPARQL* corresponding to the symbolic query, which are specific only to the *SELECT* operation.

#### IV. EXPERIMENTS

The experimental results demonstrates that the method produces a good image processing, an indexing of image and an optimal retrieval of the visual objects from different images. We used for experiments the *jpeg*s files of *TRICTRAC* dataset [22]. The images of the dataset refers to progressive image in *jpeg* format for synthetic video sequence of soccer. The specification of query assumes a simple graphical interface enabling the introduction of symbolic language query and view images obtained as a result ordered by the metric value that determines the distance between the query and response. The metric value determines intrinsic information in accordance with [23] and involves determination of the distance between hypergraph-query and each hypergraph corresponding to each image obtained. The translation from semantic query to a format accepted by the database, it requires a complex transformation. The first part of query processing done in the format switch *HGOQL* symbolic language; in phase two, *HGOQL* query is transformed into a *SPARQL* query, which allows implementation of structural matching algorithms graph type. A *SPARQL* query that corresponds to the initial query

"*player inside red square*" (search for all images where a player in red is inside square) is shown in Figure 7.

```
SELECT ?hglImage
FROM hgSoccerDb
WHERE {
  ?p Player (?c ColorAttribute);
  ?s Square;
  ?p INSIDE ?s;
}
FILTER eq(?c,"red");
```

Fig. 7. Example of SPARQL query

The query in *SPARQL* format has three components: selection operator *SELECT* clause corresponds to the specific database, the *WHERE* clause specifies criteria for selecting the hypergraph objects within the database and allows the results restriction *FILTER*. In the example, in the set of results should be included only images that contain a player in red and is in the square. A subset of the images obtained are shown in Figure 8.



Fig. 8. Image results of semantic query

#### V. CONCLUSIONS

This paper presents the algorithms for image processing, indexing and retrieval which are based on hypergraph data structure. The image processing implies the segmentation and annotation of the salient objects from the image. Using an object oriented hypergraph data structure, the spatial relations are determined and stored in an object-oriented database. This work aims to unified the phases of processing, indexing and retrieval of images. The kernel database used for experiments is the *Hypergraph.DB*, an object-oriented database, which allows to add modules for indexing and retrieval visual information. The results of the experiments on the *TRICTRAC*



dataset show that using object-oriented database allows storing and retrieving of complex objects within images. The future work implies the using of the hypergraph theory with the goal of searching and retrieving complex images based on the complex query formulated in a symbolic language.

#### ACKNOWLEDGMENT

The support of the The National University Research Council under Grant CNCISIS IDEI 535 is gratefully acknowledged.

#### REFERENCES

- [1] Bretto, A. and Gillibert, L.: Hypergraph Based Image Representation. Graph Based Representations in Pattern Recognition, 1–11 (2005)
- [2] HyperGraphDb, <http://www.hypergraphdb.org/> (consulted 15/11/2010).
- [3] Kuipers, B.: A hierarchy of qualitative representations for space. *Spatial Cognition, An Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, Springer-Verlag, 337–350, (1996)
- [4] Cohn, A.G. and Hazarika, S.M.: Qualitative spatial representation and reasoning: An overview. *Fundamentae Informaticae*, 46, 2–32, (2001)
- [5] Chbeir, R., Amghar, Y., Flory, A.: Novel Indexing Method of Relations Between Salient Objects, *Effective Databases for Text Document Management*, 174–182, (2003)
- [6] Egenhofer, M. and Herring, J.: Categorising Binary Topological Relationships Between Regions, Lines, and Points in Geographic Databases, A Framework for the Definition of Topological Relationships and an Algebraic Approach to Spatial Reasoning Within this Framework. Technical Report 91–7, National Center for Geographic Information and Analysis, University of Maine, Orono (1991)
- [7] Hoang, N., Gouet-Brunet, V., Manouvrier, M., Rukoz, M.: Delta-TSR: Une approche de description des relations spatiales entre objets pour la recherche d'images. *MajecSTIC'09*, Avignon, 1–8, (2009)
- [8] Rezaei Kalantari, K., Shirgahi, H., Ranjbar, A.: Symbolic Image Indexing and Retrieval by Spatial Similarity, *American Journal of Scientific Research*, 13, 99–112, (2011)
- [9] Oria, V., zsu, M.T., Liu, L., Li, X., Li, J.Z., Niu, Y., Iglinski, P.: Modeling Images for Content-Based Queries: The DISIMA Approach, *Second International Conference on Visual Information Systems*, 339–346, (1997)
- [10] Niu, Y., zsu, M.T., Li, X.: 2D-h Trees: An Index Scheme for Content-Based Retrieval of Images in Multimedia Systems, *IEEE International Conference On Intelligent Processing Systems*, 1710–1715, (1997)
- [11] Bloch, I.: Fuzzy spatial relationships for image processing and interpretation: a review, *Image and Vision Computing, Discrete Geometry for Computer Imagery*, 23, 89–110, (2005)
- [12] Berge, C.: *Hypergraphs*, North-Holland Mathematical Library, (1989)
- [13] Burdescu, D.D., Brezovan, M., Ganea, E., Stanescu, L.: New Algorithm for Segmentation of Images Represented as Hypergraph Hexagonal-Grid, *Iberian Conference on Pattern Recognition and Image Analysis*, (2011)
- [14] Ganea, E., Brezovan, M.: An Hypergraph Object-Oriented Model for Image Segmentation and Annotation, *Proceedings of the International Multiconference on Computer Science and Information Technology*, 5, 695–701 (2010)
- [15] Ganea, E., Burdescu, D.D., Brezovan, M., Stanescu, L., Stoica, C.: A System for Image Processing to Automatic Annotation, *Proceedings of the Fifth International Multi-Conference on Computing in the Global Information Technology*, 87–92, (2010)
- [16] Petrescu, M.: An object-oriented data model for pattern recognition systems, *Proceedings, Development and Application Systems Conference*, (1992)
- [17] Fegaras, L., Srinivasan, C., Rajendran, A., Maier, D.: lambda-db: An odmg based object-oriented dbms. In W. Chen, J. F. Naughton, and P. A. Bernstein, editors, *SIGMOD Conference*, page 583, (2000)
- [18] Berler, M., Eastman, J., Jordan, D., Russell, C., Schadow, O., Stanienda, T., Velez, F.: *The object data standard: ODMG 3.0*, Morgan Kaufmann Publishers Inc., (2000)
- [19] Gyssens, M., Paredaens, J., J. V. den Bussche, Gucht, D. V.: A graph-oriented object database model, *IEEE Transaction on Knowledge and Data Engineering*, 6(4):572, (1994)
- [20] Gagliardi, R. and Zezula, P.: An indexing model for object-oriented database systems, *Proceedings of Advanced Computer Technology, Reliable Systems and Applications*, 287–289, (1991)
- [21] Rapoza, J.: SPARQL Will Make the Web Shine, <http://www.eweek.com/>, (2006)
- [22] Desurmont, X., Hayet, J.-B., Delaigle, J.-F., Piater, J., Macq, B.: TRICTRAC Video Dataset: Public HDTV Synthetic Soccer Video Sequences With Ground Truth, *Workshop on Computer Vision Based Analysis in Sport Environments (CVBASE)*, 92–100, (2006)
- [23] Seco, N.: *Computational Models of Similarity in Lexical Ontologies*, Master's Thesis, University College Dublin (2005)

## From icons perception to mobile interaction

Chrysoula Gatsou  
School of Applied Arts  
Hellenic Open University  
Patra , Greece  
Email: cgatsou@teiath.gr

Anastasios Politis  
Graphic Arts Technology  
Faculty of Fine Arts and Design,  
TEI of Athens  
Athens, Greece  
Email: politisresearch@techlink.gr

Dimitrios Zevgolis  
School of Applied Arts  
Hellenic Open University  
Patra , Greece  
Email: zevgolis@eap.gr

**Abstract**—This study deals with the vital issue of whether a mobile phone interface icon effectively expresses the function related to it. The subject of the effectiveness of icons used in mobile phone interfaces deserves examination. Icons are an integral part of most mobile interfaces, for they are the bridge enabling interaction. We also examine how far any icon represents the meaning of the function for which it has been designed, chosen and installed by the mobile phone manufacturer and designer. Among the chief findings are (1) graphical representation affects the recognition rate of icons and influences user perception and (2) there are significant differences in performance in recognizing icons among different age groups.

**Keywords**—icons recognition; human factors; interface design; mobile interaction.

### I. INTRODUCTION

MOBILE phone interaction is nowadays part of everyday human behavior and an activity which involves speaking, listening, touching and performing other tasks, in order to communicate. Interactivity converts a system into a communication medium by eliciting user interaction with the interface. One of the main goals of a mobile phone interface is to relate phone functions and operations to elements of interaction that are performed well (e.g. sounds and visual elements). Mobile interfaces use icons to represent the functionality required by users in performing their tasks. Since visual aspects, such as graphics and icons, are essential elements of user-device interaction, are used extensively in interface design on the assumption that visual icons are capable of transcending language barriers and of presenting meaning in condensed form [1], [2], [3]. With the increase in the use of new technologies and of the internet at home, there is an exponential growth in numbers of novice users, that is, ordinary people who lack skills in computer science and are drawn from a wide range of backgrounds, they face difficulties in operating their computers. Ordinary people are now the main target of the market, which produces new applications very rapidly. Consequently, there is a need for new tools with particular features to assist such users. Yet there has been little investigation of the influence of graphical icons on the perception of ordinary mobile phone users.

An icon can be defined as a graphical representation of concepts that symbolize computer actions [4]. Exponents of icons argue that iconic interfaces enjoy many advantages [5].

One such suggested advantage is that icons are easily recognized [6]. Also, it is suggested, that graphic images help users memorize and recognize functions available within an application [7]. In addition, iconic interfaces are especially important for novice users who only infrequently use interactive systems. To be effective, an icon must fulfill several criteria, such as whether it is visible, legible, and comprehensible. Studies have found that the visual and cognitive features of icons significantly influence an icon's effectiveness [8], [9], [10]. Recently designers of mobile interfaces have been using icons to represent the functionality required by users to perform their tasks. Icons are a popular method for visually representing functionality, because they provide direct access, allow direct manipulation and can economise on valuable space in interfaces. A key concern in the design of iconic interfaces is the effective depiction of the meaning of the icon. Potentially speaking, an icon can represent both the referent and its attributes, associations, and state [2].

The proper use of iconic mobile interfaces reduces system complexity and helps users interact with mobile phones more easily.

Given this discussion, present study seeks answers to the following questions:

1. Are mobile phone function icons easily recognizable by a wider audience?
2. Is there any difference in recognition rate among different age groups?
3. Are there any differences in the recognition rate between the genders in each of the age groups?

### II. ICON CLASSIFICATION, SEMIOTICS AND INTERACTION

Icons can be divided into broad categories that rest on Pierce's early explanation of semiotics. Pierce classified signs in three categories that is, **icon**, **index** and **symbol** [11]. For a sign to exist, it must consist of all three parts (the object, the representamen and the interpretant) and the interaction between them is a process Peirce termed *semiosis* (from Greek '*sēmeiōsis*').

**Icon.** An icon is the simplest of these types of representation, since it consists of a pattern of lines that physically resembles what it 'stands for'. Icons display features that resemble the object they signify.

**Index.** An index correlates in space and time to its meaning and relates indirectly to the concept of its referent.













		Type of Representation				
		Pictographic	Concrete	←	→	Abstract
Iconic						
Indexical						
Symbolic						

Fig. 1 Types of icon representations (adapted from Nadin)

**Symbol.** A symbol is a sign whose relation to what is signified is conventional or arbitrary.

Wileman states that symbols can be assigned in three groups. Pictorial, graphic and verbal symbols range from concrete to abstract representations [12]. Fig. 1, illustrates several representations of a “camera” based on Nadin’s idea [13]. Different amount of interpretation from concrete to abstract and different types of icons (iconic, indexical, symbolic) plays an important role in user performance. The interpretation becomes easier, as the representation becomes more schematic. As the level of abstraction increases, the sign becomes progressively more generic and less complex. From a semiotic point of view, the design of an interface for a mobile application consists of various signs. By means of these, the designer tries to convey the meaning he intends to convert [14]. When the user interacts with the screen of the mobile phone, he or she is required to guess the object of the sign, since the sign is designed to convey specific meanings. When the user’s interpretation (interpretant) matches the intended object of the sign, the designer has achieved his aim of producing a successful icon [13]. Ideally, the link between the representamen and object should be obvious to all the users of the interface and result in only one interpretant. This should activate the correct mental model, which allows the user both to understand the action and to interact appropriately [15].

### III. DIFFERENT SYMBOLS, DIFFERENT MEANINGS, IDENTICAL FUNCTIONS

There is an increasing range of existing iconography in mobile phones, together with a number of interesting graphics. Not all users though can transfer their skills from one model to another, because of differences in the interface and the icons between the two models. Different individuals interpret the same icon in different ways and one icon may be capable of more than one interpretation, this phenomenon is being labeled the ‘ambiguity’ of the icon.

Rossi and Querrioux - Coulombier suggest that “the relationship between an icon and its meaning should be automatic and consequently independent of any learning” [16]. This means that for an icon to work more effectively

than some other means of representation, such as a textual description, it needs to draw on the understanding of the implicit meaning of the icon. Various icons on various handsets, differing among themselves in appearance, but representative of the same function, may complicate the intellectual model applied by the user and so cause problems in the perception on the part of the user.

The most important role of an icon is to convey, without the use of text, the meaning of the function it represents, thereby making icons more efficient than text in the operation of mobile phones and in function implementation. The effectiveness of an icon in relation to its intended meaning also depends on the degree of mapping between physical form and function, this being known as the “articulatory distance”[17]. The closer the visual representation is to the intended meaning, the shorter the articulatory distance becomes.

Ideally, the icons used in the interface for representing information will activate the appropriate mental models in the users. How the user interprets the sign will depend on the user’s mental models. Likewise, how the designer chooses to represent the object may also depend on his own set of mental models [18]. It is important to note that the function assigned to an icon by those designing it may be quite different to the meaning actually attributed to it by users.

The correct interpretation of icons also depends on other factors, such as the context in which the icon is used. Any text labels that might be displayed together with an icon and the user’s familiarity with the icon and with its application context [19]. The elderly are likely to have less experience than other younger age-groups with contemporary handset devices and to be less familiar with icons displayed by a device and with applications, which thus makes such icons more difficult to interpret.

	iPhone 2007-2010 (4 x iOS)	NOKIA 2006 (N73) 2008 (5320) 2009 (N9500)	MOTOROLA 2003 (V600) 2009 (evolve Q4)	SAMSUNG 2008 (D780) 2009 (bada OS)	SONY ERICSSON 2005 (K750) 2009 (C903)
PHONE BOOK					
PHONE CALL					
MESSAGE					
SETTINGS					
CAMERA					
CLOCK					
INTERNET					
GAMES					

Fig. 2 Types of icon representations from different handsets

Previous studies have shown that mobile phone icons make for faster, more direct access to a mobile function [16]. This then leads to the inevitable question, "What makes an effective comprehensive interface mobile icon"?

In Fig. 2, we see eight of the most frequent functions from five different popular brands of mobile phones, namely, Iphone, Nokia, Motorola, Samsung and Sony Erikson. We present icons from two models per brand (Nokia, Motorola, Samsung and Sony Erikson ) and one model from Iphone. It

is to be observed that even the same company is not consistent in its choice of symbols to depict functions. In Nielsen’s view, “the latest mobile devices are agonizingly close to being practical, but still lack key usability features required for mainstream use” [20].

IV. RESEARCH METHODOLOGY

There are several criteria that an icon must satisfy, if it is to be effective. Among these are legibility, distinctiveness, comprehension, the reaction time [21].

The main problem in evaluating icons is the proper construction and modification of them [22]. Several methods have been utilized to evaluate graphic symbols and icons. The method used most often is a comprehension test, also termed a ‘recognition test’ [23]. Howell and Fuchs were the first to devise criteria for the correct recognition of symbols, grouping them into the following categories: identifiable (60-100%), medium (30-60%) and vague (0-30%) [24]. Many researchers have employed procedures involving “matching tests” to evaluate graphic symbols [25],[26]. In the “matching” method, the suitability of an icon is evaluated in relation to other icon variables. Yet another method is the icon intuitiveness test, created by Nielsen and Sano, in which an icon is shown without any label to a small number of users, typically five [27]. The users are asked to guess what the icon is intended to represent. Sanders and McCormick have also shown that the criteria for selecting symbols generally include a degree of recognition, a matching degree and a subjective preference and opinion [28].

A. The selection of the sample for the evaluation.

After choosing handsets from five different manufacturers on the basis of brand popularity, we selected icons for our study from the main menu functions. It was impossible to represent each function by a standard number of icons, since the icons in question are extremely diverse in appearance. Some were selected on the grounds that, although they were drawn from different brands, they converged and we were eager to investigate whether such convergence aided user perception. Our goal was to determine whether or not the visual representations offered by icons do indeed help users to understand the functionality of the icon in question.

B. Participants

We employed a sample of 60 participants, all volunteers. They possessed mobile phones and came from various backgrounds. They were roughly equal in terms of gender and their age distribution is given in Table 1. All participants have normal vision, though some wore glasses or contact lenses. The majority had owned a mobile phone for more than one year. Each subject was given a brief overview of the experiment and briefed as to the purpose and procedure of the study.

C. Icon recognition questionnaire

Before answering the icon recognition questionnaire, all participants completed a pre-experiment questionnaire which collected personal details and data relating to technology skills and mobile phone experience. A paper-based icon

TABLE I. AGE, GENDER AND NUMBER OF PARTICIPANTS.

Age group	No. of participants	Participant Gender	
		Male	Female
20-29	10	5	5
30-39	11	7	4
40-49	12	5	7
50-59	14	8	6
60-69	8	2	6
70-79	5	5	0
$\Sigma$	<b>60</b>	<b>28</b>	<b>32</b>

recognition questionnaire was prepared, which involved 54 mobile phone function icon .The questionnaire was designed to examine icon recognition and perception performance over different age groups. According to the Organization for International Standardization (ISO3864), icon recognition rates should be at least 66.7%, to be acceptable [29]. With a view to making the procedure of presenting the participants with the icons they were to interpret as efficient as possible, a table was constructed in Adobe InDesign with numbered rows, placed an icon next to each number, and left the space to the right for a set of referents from eight functions that participants had to select the proper one. Since the test required that the icons be clearly visible, they were printed at high resolution. The recognition rate was computed as follows:

$$(Number\ of\ correct\ choices / Number\ of\ respondents) \times 100 = Recognition\ rate(\%)$$

No	A1	A2	A3	A4	A5	A6	A7	A7	
Phone book									
Recognition	56.7	63.3	78.3	30.3	56.7	56.7	40.7	70.0	
No	B1	B2	B3	B4	B5	B6	B7	B8	B9
Phone call									
Recognition	75.0	50.0	60.0	20.0	53.3	21.7	55.0	60.0	60.0
No	C1	C2	C3	C4	C5	C6			
Message									
Recognition	98.3	93.2	88.3	86.7	95.0	96.7			
No	D1	D2	D3	D4	D5	D6	D7	D8	
Setting									
Recognition	65.0	91.7	90.0	68.3	95.0	58.3	58.3	91.7	
No	E1	E2	E3	E4					
Camera									
Recognition	78.3	96.7	96.7	100.0					
No	F1	F2	F3	F4					
Clock									
Recognition	60.0	100.0	61.7	61.7					
No	G1	G2	G3	G4	G5	G6	G7	G8	G9
Internet									
Recognition	40.0	68.3	93.3	45.0	95.0	91.7	95.0	85.0	65.0
No	H1	H2	H3	H4	H5	H6			
Games									
Recognition	78.3	40.0	30.0	81.7	85.0	93.0			

Fig. 3 Recognition rate of icons.

In this study, the 54 icons were graded according to their recognition rate.

V. ANALYSIS OF RESULTS

The summary of the test results is shown in Fig. 3. The recognition rate for 29 icons was over 66.7%, a fact which provides an overall answer to questions we posed ourselves. In view of the ISO standard mentioned above, we award the icons we tested one of two grades: 'good', with a correct answer rate of above 66.7%, and 'low', with a correct answer rate below 66.7%. On this basis, 29 of the icons tested are to be considered 'good' and so are suitable for mobile phone use, the remaining 25 icons achieving only a recognition rate below this level.

Other facts emerge from our analysis. Six mobile icons were easily recognized and associated with their correct functions, thus fulfilling Howell's criteria. Icons E4 and F2 enjoyed the highest recognition rate of all, 100%. However, the analysis of our test results relates to our research questions to a greater degree than this. It is clear that the icon recognition rate differs over age groups, some icons enjoying a high recognition rate and some others a lower rate. Older participants were less accurate in recognizing and interpreting the meaning of the icons. The findings shown in Fig.4 and Table II illustrate this point.

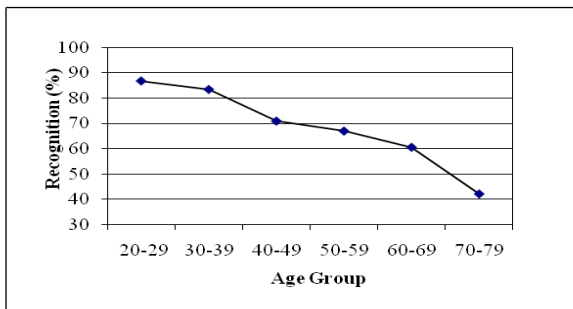


Fig. 4 Recognition rate and different age group

TABLE II.  
MEAN RECOGNITION RATE IN AGE GROUPS

Age group	No. of participants	Mean Recognition rate
20-29	10	86.9%
30-39	11	83.5%
40-49	12	71.0%
50-59	14	67.3%
60-69	8	60.6%
70-79	5	42.2%
<b>Σ</b>	<b>60</b>	<b>68.5%</b>

If we regard a recognition rate of 66.7% as indicating success, the most effective icons are:

- F2 and E4, with a recognition rate of 100%,
- C1, with a recognition rate of 98.3%,
- E3, with a recognition rate of 96.7%,
- F2, with a recognition rate of 96.7%,
- F3, with a recognition rate of 96.7% and
- D5, with a recognition rate of 95.0%.

Information regarding matters of experience with technology and of gender was derived from the pro-experiment questionnaire. An analysis of the results is given in Fig. 5 and 6. As for the six icons whose recognition rate fell between 20%-40%, various suggested factors, which are summarised in Table III, may be responsible for this poor performance.

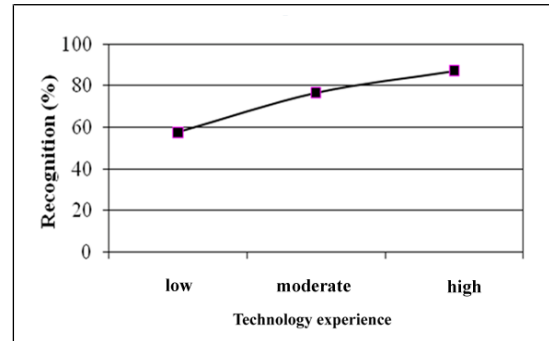


Fig. 5 Recognition rate and experience with technology.

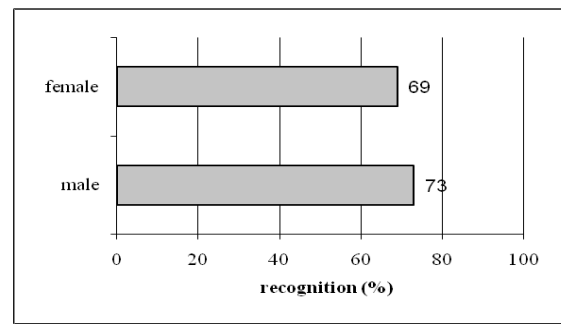








Fig. 6 Recognition rate and gender.

TABLE III.  
ICONS WITH LOW RECOGNITION RATE 20%-40%

Icons	R/r*	Original meaning	Possible reasons for misunderstanding
	40%	phone book	The graphic, intended to indicate a note book, is visually unclear.
	40%	internet	The use of a compass to denote the internet is unfamiliar to some users. Generally an icon of a globe is more effective.
	30%	phone book	The combination of head phone and notebook is confusing.
	30%	games	The addition of the word 'games' would probably add to clarity and effectiveness
	21.7%	phone call	The combination of handset and an individual is ambiguous.
	20%	phone call	The resemblance to a notebook may cause confusion with the phone book icon.

\*R/r = recognition rate

Interestingly, the recognition rate for the various icons denoting a very basic function, "phone call" (Fig. 7), is surprisingly low, with the exception of the icon employed by the Iphone, which consists of a head phone. This enjoyed the



highest rate of recognition (75%), whilst that of all the other icons fell beneath 66.7%






	IPHONE	NOKIA	MOTOROLA	SAMSUNG	SONY ERICSSON
Phone call icon					
Recognition rate	75%	50%	60%	20%	53.3%
				21.7%	55.0%
					60.0%
					60.0%

Fig. 7 Recognition rates for various icons representing ‘phone call’ function.

VI. DISCUSSION

In this particular study, we found that:

- 29 of our 54 icons enjoyed a recognition rate of more than 66.7% ,
- is a significant difference in recognition rates among age groups, with recognition rate decreasing as age increases and that
- there is no importance difference between genders, with the recognition rate displayed by males being only 4% higher than that displayed by females.

Our study suffers from limitations that may have given rise to inaccuracies in our results. As we have pointed out, we settled upon a paper-based form of test, as some of our participants were unfamiliar with computer technology. Such participants, ignorant of computer technology, were unable to compare the icon they were requested to evaluate with other icons from the application from which the test icon was drawn. This does not reflect reality, where the users of an application may be in a better position to guess the meaning of the icon in question by comparing it with other icons in the same application.

For reasons of legibility, comparatively large depictions of icons were used in our recognition test. In reality, of course, icons are becoming ever smaller and less visible [22]. Our focus was upon ordinary people. Although they were not necessarily experienced users of technology and were drawn from various age groups, they were called upon to evaluate icons, no easy task. Among the factors of which account needs to be taken is the medium in which the icons were presented and examined and, above all, age differences.

In general, however, iconic signs are more easily recognized than symbolic signs. It is thus extremely important that the appropriate design style be selected at the initial stage of the icon design process. If either information or function has strong ties with an object, a pictorial icon is the best choice. Examples of this are our icons F2 and E4, which enjoyed a recognition rate of 100%.

VII. CONCLUSION

We have dealt in this study only with the representation and recognition of icons, yet our findings can contribute to the improvement in how a larger number of users experience interfaces. Other issues, however, such as the structure of menus and colour combinations employed in icons, also require in-depth study.

Since the amount of information in our lives continues to increase, information designers must design solutions that match users’ requirements as much as possible. The proper selection of graphical elements are one way to optimize

communication with users, but requires designers to be aware of how users interact with graphical elements.

The use of an appropriate icon is a vital factor in ensuring the correct functioning of mobile phone applications. In order for icons to evoke the intended meaning in the viewer’s consciousness, or even subconsciousness and for them to achieve communication between designer and user, a symbol should display a strong, direct association with the desired meaning, in the mind of both designer and user. During the icon formulation process, a design whose aim is to produce functional results makes such functions comprehensible. Furthermore, in order to help new or ordinary users interpret icons correctly, some form of comprehensive test or test of recognition should precede any attempt at improving performance.

We hope that the results of our study will offer a deeper understanding of how a wider audience uses mobile phones and icons, in particular.

ACKNOWLEDGMENT

This research was supported by the European program ESPA and by HERAKLEITOS II in Greece. We are also grateful to the participants in the study.

REFERENCES

- [1] S. Caplin, *Icon design: Graphic icons in computer interface design*. London: Cassell, 2001.
- [2] D. Gittins, “Icon-based human-computer interaction”. *International Journal of Man Machine Studies*, 24, (1986), pp.519-543.
- [3] S. J. P. McDougall, M. B. Curry, and O. de Bruijn, “Exploring the effects of icon characteristics on user performance: The role of icon concreteness, complexity, and distinctiveness,” *Journal of Experimental Psychology: Applied*, vol. 6, no. 3, 2000, pp. 291-306.
- [4] C. Ware, *Information Visualization*. Morgan Kaufmann, 2000.
- [5] G. Shank and P. Darke. "Understanding Corporate Data Models," *Information and Management*, 35:1, 1999, pp. 19-30.
- [6] B. Shneiderman, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 3rd ed. Reading, MA: Addison-Wesley, 1997
- [7] K. Siau, "Human-computer interaction: The effect of application domain knowledge on icon visualization," *J. of Computer Information Systems*, 45(3), 2005 p.53-62.
- [8] M. A Blattner, D. A. Sumikawa, and R. A. Greenberg, “Earcons and icons: their structure and common design principles”. *Human-Computer Interaction*. 1989
- [9] M. E. Familant, and M. C. Detweiler, “Iconic reference: evolving perspectives and an organizing framework”, *International Journal of Man± Machine Studies*, 39, 705± 728, 1993
- [10] Y. Rogers, "Icons at the Interface: Their Usefulness," *Interacting with Computers*, 1:1, 1989.
- [11] C. S Peirce. *Logic as semiotic: the theory of signs*. In: Innis, R. E. (Ed.) *Semiotics: an introductory reader*. Bloomington, Indiana: Indiana University Press. 1985.
- [12] M. S. Wileman *Visual Communicating*. Educational Technology Publications, New Jersey, 1993.
- [13] M . Nadin, “Interface design: A semiotic paradigm”. *Semiotica*. 69: 269–302, 1988.
- [14] R. Buchanan, “Declaration by Design: Rhetoric, Argument, and Demonstration in Design Practice”, *Design Issues*, 2(1), 1985, pp. 4-22.
- [15] S. Isherwood, “ Graphics and Semantics: The Relationship between What Is Seen and What Is Meant in Icon Design”. *HCI (17) 2009*, p.197-205
- [16] J. Rossi, & G. Querrioux-Coulombier, “Picture Icon and Word Icon”. *From Human and Machine Perception*. New York, NY: Plenum Press, 1997.
- [17] E.L.Hutchins, D.J.Hollan, and D.L. Norman,. “Direct manipulation interfaces”, in Norman, D.A. and Draper, S. (eds) *user-centered system design* Lawrence Earlbaum Associates, Hillsdale, NJ, USA, 1986

- [18] D. Norman, *Things That Make Us Smart*, Addison-Wesley Publishing Co., Reading, MA 1993
- [19] W. Horton, *The Icon Book – Visual symbols for computer systems and documentation*, New York: Wiley and Sons, 1994.
- [20] J. Nielsen, Useit.com [Online]. Available: <http://www.useit.com/alertbox/20030818.html>
- [21] R. Dewar, "Design and evaluation of public information symbols," in *Visual Information for Everyday Use: Design and Research Perspectives*, H. J. G. Zwaga, T. Boersema, and H. C. M. Hoonhout, Eds. London: Taylor & Francis, 1999, pp. 285-304.
- [22] S. Blankenberger and K. Hainj. Effects of icon design on human-computer interaction. *Int. J. Man-Machine Studies* 1991, 35, 363-377
- [23] W. C. Howell, & A. H. Fuchs, Population stereotypy in code design. *Organisational Behavior in Human Performance* 3, 1968, pp 310-339.
- [24] H. I. Cheng, P. E. Patterson, "Iconic hyperlinks on e-commerce websites. *Applied Ergonomics* " 38 (1), 2007, pp 65-69.
- [25] E. Heard, "A symbol study-1972," Paper No 740304, Society of Automotive Engineers, New York, 1974.
- [26] Easterby and H. Zwaga (Eds.), *Information design, "The design and evaluation of signs and printed material"*(pp. 277-297). New York: J. Wiley & Sons
- [27] J. Nielsen, and D. Sano, SunWeb: User Interface Design for Sun Microsystem's Internal Web. In *Proc. 2nd World Wide Web Conf. '94: Mosaic and the Web*. Chicago, IL, pp. 547-557.
- [28] Sanders, M. S. and McCormick, E. J "Human factors in engineering and design," 7th ed, New York: McGraw-Hill, 1993.
- [29] D. P. T. Piamonte, J. D. A. Abeysekera, and K. Ohlsson, Understanding small graphical symbols: a cross-cultural study. *International Journal of Industrial Ergonomics*, 27 (6), 2000, pp. 399-404



# Automatic Speech Recognition for Polish in a Computer Game Interface

Artur Janicki and Dariusz Wawer

Institute of Telecommunications, Warsaw University of Technology  
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland  
Email: A.Janicki@tele.pw.edu.pl, Dariusz.Wawer@gmail.com

**Abstract**—The paper describes the process of designing a task-oriented continuous speech recognition system for Polish, based on CMU Sphinx4, to be used in the voice interface of a computer game called *Rally Navigator*. The concept of the game is presented, the stages of creating the acoustic model and the language model are described in details, taking into account the specificity of the Polish language. Results of initial experiments show that as little as 15 minutes of audio material is enough to produce a highly effective single-speaker command-and-control ASR system for the computer game, providing the sentence recognition accuracy of 97.6%. Results of the system adaptation for a new speaker are presented. It is also showed that the statistic trigram-based language model with negative trigrams yields the best recognition results.

## I. INTRODUCTION

**A**UTOMATIC speech recognition (ASR) systems gradually replace keyboards and touch pads in various applications - so it happens in word processors where dictation software is being introduced. But there are also trials to replace joysticks and buttons in computer games, thus making the games more interesting and enabling multimodal input. ASR systems can be successful in computer games, on condition that they provide high recognition accuracy and short processing time.

To ensure realistic conditions, an ASR system in the computer game should be able to recognize continuous speech, which is how people usually talk. Continuous speech recognition is much more difficult than recognition of isolated words. What is more, apart from a few examples ([1], [2]), such systems barely exist for the Polish language, which is highly inflective and thus hard to recognize.

This paper describes experiments with designing a small-vocabulary task-oriented automatic continuous speech recognition system for Polish, which can be used in the voice interface of a computer game.

## II. AUTOMATIC SPEECH RECOGNITION

### A. Methods for Continuous Speech Recognition

Early works on ASR systems, starting in the 1950s, concerned recognition of isolated phonemes, or at best - a few words. One example is the isolated digits recognizer constructed at Bell Labs in 1952 [3]. The invention of Dynamic Time Warping (DTW) in the late 60s allowed for projects on larger-vocabulary word recognition and for processing of connected words [4]. In 1971 the ARPA SUR (Speech

Understanding Research) program started, aiming at creating a reliable large vocabulary ASR for continuous speech. One of its results was HARPY - an ASR system developed at the Carnegie Mellon University, working with semantic accuracy of 95% at the processing speed of 80 times real-time [4]. Studies on continuous speech recognition continued intensively in the 1980s, when the usage of statistic acoustic modeling and statistic language modeling advanced, and they have continued till nowadays.

The key to successful continuous speech recognition is a combination of a highly accurate *acoustic modeling* (AM) and a proper *language modeling* (LM).

1) *Acoustic modeling*: its aim is to recognize as accurately as possible the phonetic content of the input speech signal, by comparing parameters of the speech signal (usually MFCC - mel-frequency cepstral coefficients or PLP - perceptual linear prediction parameters) with acoustic models stored in the ASR system. There were successful trials of using artificial neural networks for phonetic modeling, but contemporary systems almost exclusively use statistical acoustic modeling based on Hidden Markov Models (HMM). Context-dependent phonemes, called also triphones, are usually the base speech units used in acoustic modeling. Efficient acoustic models are trained on multi-speaker speech corpora containing hours of transcribed recordings, therefore their preparation for a new language is a very demanding and time-consuming task.

2) *Language modeling*: it is very important, because for continuous speech the word boundaries are difficult or impossible to detect. The language model enables the ASR system to decode the sequence of phonemes, recognized during acoustic recognition into the correct sequence of words. A proper language model makes it very likely that the recognized sequence will have correct syntax and will be semantically correct. One of the language modeling techniques is the use of  $N$ -grams [5]. In this statistical method sequences of  $N$  words are assigned various likelihoods. Such a language model is created based on statistical analysis of a given language or a given domain, depending on the ASR type (e.g. if it is a large-vocabulary one or task-oriented).  $N$  ranges usually between 2 to 4. Some probability mass is left for 2-grams (*bigrams*) and 3-grams (*trigrams*) unseen during the training procedure - such an operation is called *model smoothing*.

Effective way of decoding the text is to perform a Viterbi search on a recognition network (it can be in the form of

a tree or word lattice), built out of the lexicon. Because of high calculation complexity, often the acoustic recognition (phoneme-based) and lexical one (word-based) are performed sequentially: as we proceed with the signal analysis, word sequence hypotheses are created, and at the same time acoustic and language scores are calculated, so that only the paths with highest scores are continued. The sequence with the highest score which reaches the end of the signal is the decoded sentence.

Another method of approaching language modeling is to build a grammar describing all possible phrases. To describe a grammar one can use Java Speech Grammar Format (*JSGF*) [6], which is a part of Java Speech API [7] and is designed for strict command and control systems. The grammar is defined by declaring rules which can contain either words, operators, or other previously declared objects. Such a language model based on grammar leaves no margin for any unforeseen word sequence.

### B. The Sphinx Framework

CMU Sphinx framework, partly funded by DARPA, was created and has been continuously developed at Carnegie Mellon University. It consists of several subprojects. In our work we used *SphinxTrain* [8] and *Sphinx4* recognizer. *SphinxTrain* consists of tools written in C which can be used to train and adapt acoustic models. It also provides scripts which simplify acoustic model generation. *Sphinx4* is written in Java and is a complete recognition system with modular architecture. It communicates with the application in two ways: the first one is the input, which acquires the audio data, and the output, which returns the recognition - the best match for the data. The other communication input is a recognizer control mechanism and the output are system state notifications.

Internally, *Sphinx4* consists of the frontend, the decoder, the linguist and the configuration manager. The frontend receives audio data and may perform early signal modifications like removing long silences or amplifying the signal. After the signal leaves the frontend, the feature extraction occurs and the data reaches the decoder. Simultaneously the linguist generates the possible word sequences (possibly as word lattices or trees) based on the language model and sends them to the decoder. With the voice features and data from the acoustic model the decoder scores the possible sequences keeping only the most probable. After the signal finishes or a final state is reached, the result of the recognition is returned to the application. This corresponds to the speech recognition process described earlier in Section II.A.

Many of the above mentioned elements have several implementations, and each implementation can be separately configured. E.g. there are implementations for both  $N$ -gram and *JSGF* language models. The Sphinx framework was originally designed for English, but nowadays it supports also, among others, Spanish, French and Mandarin. However, there are no acoustic nor language models available for Polish.

To use it with Polish, the package required some modifications to work properly, e.g. several internal classes needed

to be adapted to correctly read names containing non-ASCII characters, which were found in the language models and dictionaries.

### C. Speech Recognition in Computer Games

Nowadays computer games are a large and well developing industry, generating high revenues. To be successful in the market, a game has to be easy to play, intuitive, must be fun and interesting and, last but not least, must not irritate the player.

While the first three of the requirements apply rather to the game design, the last one refers as well to the ASR system, and it places severe restrictions on the system. First of all, the system must recognize the command in a short time. The amount depends on how fast is the action in a game, but times greater than one second are definitely too long. Secondly, the accuracy of recognition must be really high. Should the system make an error and issue a wrong command, the player would be unhappy. Should such situation occur repeatedly, he or she would quit the game and never play it again.

An ideal game using speech recognition should work robustly for every speaker, including people with various accents, non-native speakers and even people with speech deficiencies.

There are many factors that can have negative impact on speech recognition robustness:

- usually a fairly low quality microphones being used;
- games are often played in noisy environments;
- highly-effective, fast ASRs consume large amounts of computing power, which may limit their usability on consoles and older PCs.

It is, however, advantageous that games usually require command-and-control (task-oriented) systems, which are easier to implement.

A computer game is a specific environment for a speech recognition system. It can be treated as a special type of a dialog system, in which in some cases the game may actually predict what the user might want to say, knowing the game scenario. This way the system may modify dynamically the language model to reflect that. Care must be taken while implementing such features and changes to the model must remain moderate.

The usage of speech recognition systems in computer games is yet unexplored. There have been a few attempts to utilize ASR systems, particularly in strategy games [9] and flight simulators [10]. ASR systems implemented in these games did work properly, but have not been enthusiastically accepted, mostly because they were tedious to use.

## III. DESIGNING THE RECOGNITION SYSTEM FOR POLISH

In this section we will describe specificity of the Polish language, we will present the concept of the computer game *Rally Navigator* which is going to use the ASR interface and we will describe the consecutive steps of designing the speech recognition system for this game.

TABLE I  
POLISH PHONEMES INVENTORY, BASED ON [11]

phoneme type	voiceness	phonemes (in SAMPA)
vowels	voiced	a e o u i ɪ
nasals	voiced	m n ŋ N
plosives	unvoiced	p t k k'
	voiced	b d g g'
fricatives	unvoiced	f s s' S x
	voiced	v z z' Z
affricates	unvoiced	ts ts' tS
	voiced	dz dz' dZ
laterals	voiced	l
liquids	voiced	j w
trills	voiced	r

### A. The Polish Language

The Polish language belongs to West-Slavic language family. Spoken language contains 38 phonemes: 6 vowels and 32 consonants [11], out of which fricatives are the most numerous (9 phonemes). Most of the phonemes (plosives, fricatives and affricates) exist in pairs: unvoiced-voiced, e.g. [ts'] - [dz']. Some of these voiced phonemes become unvoiced in an unvoiced context (so called devoicing), and opposite: unvoiced phonemes can become voiced in certain circumstances. This results in pronunciation variation. As for the prosodic features: a melody (pitch contour) of the word is irrelevant to the word's meaning, however the sentence melody sometimes carries semantic information (e.g. can make a question or add an emotional flavor).

From the point of view of grammar, the Polish language is complex. It is highly inflective - nouns are inflected according to 7 cases and 2 numbers, verbs are inflected according to gender, tense and number, adjectives and numerals are inflected, too [12]. There are 3 genders in singular and 2 genders in plural. Thanks to the high inflection, the word order in a Polish sentence is rather free, as the function of the word (e.g. whether a noun is a subject or an object) is determined by the form of the word, and not by the position of the word within the sentence. Subjects are often dropped, because they can be deducted from the form of the verb. There are no articles preceding nouns or any other parts of speech.

These features make continuous speech recognition for Polish quite a demanding challenge. Lack of articles makes detecting nouns difficult. High inflection sometimes causes a single word to have dozens of forms, which often sound similarly. Loose word order makes it very difficult to create a good language model. Pronunciation variation, which can be helpful in speech synthesis [13], disturbs speech recognition. Luckily these problems are less severe if we consider a small vocabulary recognition, which is usually the case for a computer game.

### B. Concept of the Computer Game

Our main aim while inventing the game was to make the ASR system its integral part, not an additional or alternative way of controlling it. We also wanted the game to be original and innovative, possibly giving the player an opportunity

to experience something he has not experienced before. We settled on a game we called *Rally Navigator* in which the player would compete in races - not as a driver, but as a navigator. The player's task would be to provide the driver with information about the route and track elements like curves and straights. To make the game more difficult (and the ASR system more complex) we also decided to include speed control and gear switching. The aim of the game is to win the rally. The more precise and well-timed the information supplied by the player, the quicker the car reaches the finish line.

### C. Developing the Acoustic Model

The following elements were required to train a new acoustic model:

- audio data with recorded speech;
- transcription of each audio file;
- dictionary with phonetic representations of all words appearing in the transcriptions;
- list of phonemes (and sounds) appearing in the transcriptions.

The amount of audio data required to properly train the model depends on the type of the model. For a simple command-and-control one-speaker system (the one we began from) the amount of data can be fairly low. For multi-speaker systems the amount of required audio increases, and increases even further for dictation purposes.

To reflect the conditions in which the system will be used, the audio signal was recorded in a home environment, using 16 kHz sampling. The speaker read the following:

- 114 specially designed, phonetically balanced CORPORA [14] sentences, which contain all phonemes and all diphones (pairs of phonemes) appearing in the Polish language;
- the same set of sentences, but spoken faster. The reason behind the faster set of data is that we predict that in our game the players will sometimes speak hastily, for example for a sequence of tight curves. The Corpora sentences in total formed 11 minutes of our training data;
- sample commands, which will be used in game, and numbers, which must be correctly recognized for the game to work (curve angles are described with numbers, so are gears and lengths of straight road).

In total, we prepared 25 minutes of audio.

The audio files were then transcribed, including special silence marks for all silent moments appearing in the file. Such marks allow the training algorithm to better align the speech and, in turn, produce better models.

The phonetic dictionary was prepared in such a way that it contained all words with all possible variants of their pronunciation, to take into account pronunciation variability, caused by various speaking manners and the specificity of Polish, described earlier. Careful preparation of phonetic dictionary prevents from incorrect association of a phoneme with audio parameters of a different phoneme which would effect in decreasing the model's accuracy.

The list of sounds contained mostly phonemes, but also sounds like clicking a mouse button or breathing. It is important to include such sounds in the transcriptions (if they occur in audio files) for two reasons: they will not be mistaken with other phonemes during training and the working ASR system may successfully recognize and omit them.

To train our acoustic model we used applications and scripts from SphinxTrain [8], which is a part of CMU Sphinx.

#### D. Training the Language Model

We decided to use a tool supplied by CMU Sphinx called *Sphinx Knowledge Base Tool* [15], which generates the language model from a list of sentences. The quality of the model therefore depends on how well the list is prepared or, in other words, how well it reflects the commands which will be issued in the system. The generation process itself is simple, the tool notes and counts all appearing  $n$ -grams and then converts the results to  $n$ -gram language model format compatible with Sphinx. The harder part is creating a good set of sentences. For example, creating a file with all possible commands is actually not a good idea. Longer commands with more parameters would dominate the data making the less complex commands less probable and the model would then not reflect the real probabilities of  $N$ -grams in the system. Furthermore, the amount of possible commands is very large, especially if we want to make the system flexible and allow different variations of the same command. One solution would be to repeat the less complex commands in the training file, so that their amount would be similar to the more complex ones. This would make the file enormously big. But what if we wanted to include two commands in a single sentence? We could then make all possible combinations of these commands, but that would cause the file to grow exponentially.

For our model we have decided to take a different approach. We split our commands into at least three word long parts, each part with at most two parameters and generated all possible variations of each of such fragments. Some of these fragments overlapped, so that all possible bigrams and trigrams were included. This way also allowed us to repeat the less-parametrized fragments, since the amount of repetitions did not have to be large. This whole operation caused our training file to become very small, simple and quick to generate. After using the Sphinx Knowledge Base Tool on this file, the resulting model required some fine-tuning. Most importantly all impossible silence-starting and silence-ending  $N$ -grams were removed.

The last stage of building the model was the inclusion of, as we called them, *negative  $n$ -grams*, which are artificial  $n$ -grams with near-zero probability. All sequences built from unigrams, even if they are not included in bi- or trigrams, have a greater than zero probability of being recognized. Negative  $N$ -grams can be used to disallow word sequences which we consider invalid in our systems, but are sometimes incorrectly recognized by the system. We have identified several such sequences using our tests and included appropriate negative  $N$ -grams in our language model.

Our *JSGF* grammar was created manually. First we defined groups of words describing parameters in our commands, like distances, directions and angles. Then we defined constant fragments of commands using these groups, some of these in a few variations, each correct from polish languages point of view. Next step was grouping these fragments into commands, and some commands into sequences of commands. It is important to note, that we have tried to relax the strict commands by making some words optional, and providing alternatives to some of the words. After the grammar was complete we ran the tests, and then repeated the whole process for sentences from the tests which did not match any of the grammars rules.

## IV. RESULTS

This section presents the most significant results of the carried out experiments.

The ASR system was tested using a set of 85 recordings, containing commands which are likely to occur in the *Rally Navigator* game. It contained phrases 2-16 words long, total number of tested words was 422. 2 male speakers were recorded, speaker *A* whose voice was used the acoustic model training, and speaker *B* used only for testing. Total time of test recordings was 489 s.

Usually when testing ASR system tests, the following metrics are evaluated:

- *substitutions* (Sub) - the number of words which were recognized as other words
- *insertions* (Ins) - the number of words which were wrongly added to the recognized words
- *deletions* (Del) - the number of words which omitted in recognition
- *word error rate* (WER) - the ratio of word recognition errors (such as substitutions, insertions and deletions) against the total number of words;
- *word accuracy* (WA) - the ratio of correctly recognized words against the total number of words; it is strongly related to WER;
- *sentence accuracy* (SA) - the ratio of correctly recognized sentences against the total number of sentences.

Figure 1 shows the recognition performance for various language models. The JSGF grammar yielded the worst results ( $WA = 79.15\%$ ,  $SA = 61.2\%$  for speaker *A*). More detailed analysis of the recognition logs showed that it worked very well only for short sentences. However, it had trouble correctly interpreting long word sequences, especially if they consisted of more than one short command. The recognizer would in such cases completely ignore the data after the first command, in spite of the used grammar definition which allowed compound commands.

After switching to the  $N$ -gram language models, the recognition improved. Even unigram models enabled accuracy better than the one for JSFG, but after adding information about probabilities of bigrams and trigrams, the results improved significantly, yielding word accuracy of 98.58% and sentence accuracy of 92.9%. The use of negative trigrams turned out to

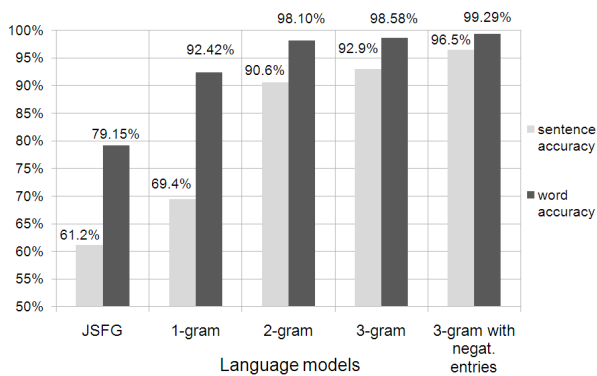


Fig. 1. Word accuracy and sentence accuracy for recognition using various language modeling.

be a successful move, giving for speaker *A* the final result of  $WA = 99.29\%$  and  $SA = 96.5\%$ .

Figure 2 displays the performance of recognition when the acoustic models were trained with different size of audio data. The word error rate for speaker *A* after training the acoustic model with 114 CORPORA sentences (i.e. 7 minutes of recording) was almost 4%, what is considered high, taking into account that it is a small-vocabulary task-oriented recognition. After adding 4 minutes more, containing the same sentences, but uttered faster, WER became slightly below 3% and the sentence accuracy increased from 88.2% to 90.6%. When adding recordings containing numerals and control commands, the performance continued to improve until the training set contained 15 minutes of recordings, where WER equaled 0.7%. Sentence accuracy at this moment reached 97.6%, what was considered a satisfactory result. Further enlargement of audio data resulted in worsening of WER up to 2.1%, due to slight increase of substitutions and deletions. Training using 25 minutes of recordings yielded again a low value of WER - 0.9%.

It is noteworthy that not every word deletion, substitution or insertion resulted in a wrong command. E.g. omitting *w* (here meaning 'to') in a sentence *skrec w prawo* ('turn to the right') caused the command change into 'turn right', being actually the same. So the semantic accuracy was even higher than the sentence accuracy.

Table II gives information about the recognition performance both for speakers *A* and *B*. When challenging the system with the voice of speaker *B*, who was not used to create the acoustic model, the ASR system was able to recognize correctly 72.9% sentences at the WER rate of 8.3%. After adapting the models with 10 CORPORA sentences of the speaker *B*, WER even slightly increased, but adaptation using 20 sentences decreased WER down to 6.62%. Further enlargement of the adaptation session was steadily improving the recognition performance, but 80 sentences were required to make WER as low as 1.42%, with sentence accuracy of almost 93%.

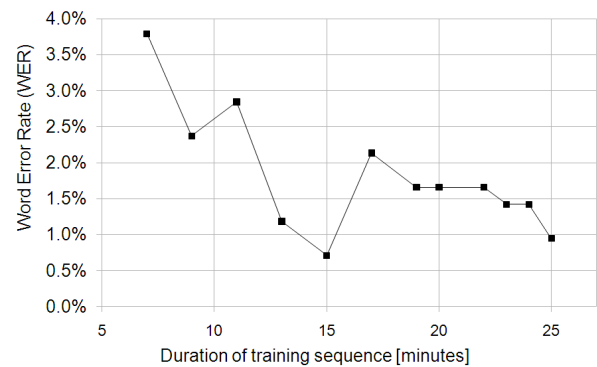


Fig. 2. Word error rate against the duration of audio signal used in the training process.

TABLE II  
RECOGNITION RESULTS FOR SPEAKERS A AND B WITH VARIOUS NUMBERS OF SENTENCES USED FOR ACOUSTIC MODEL ADAPTATION

speaker / adaptation	WER	WA	SA	Sub	Ins	Del
A	0.9%	99.29%	96.5%	2	1	1
B, no adapt.	8.3%	92.91%	72.9%	20	5	10
B, 10 sentences	9.69%	90.78%	69.4%	28	2	11
B, 20 sentences	6.62%	94.09%	77.6%	18	3	7
B, 30 sentences	5.44%	95.27%	81.2%	13	3	7
B, 40 sentences	4.49%	95.98%	82.4%	9	2	8
B, 60 sentences	3.55%	96.93%	84.7%	6	2	7
B, 80 sentences	1.42%	98.58%	92.9%	3	0	3

## V. CONCLUSION AND FUTURE WORKS

This paper described the process of designing a task-oriented continuous speech recognition system for Polish, based on CMU Sphinx4, to be used in a computer game called *Rally Navigator*. We presented the steps undertaken to create the acoustic model and the language model, using both the grammar and the statistic *N*-gram model.

As for the language model, we showed that the best results were achieved if the statistic trigram model was used. We improved it by adding negative trigrams, what decreased the number of misrecognized words.

Initial experiments showed that the audio material as short as 15 minutes is enough to produce a highly effective single-speaker command-and-control ASR system, providing the sentence recognition accuracy of 97.6%. What was expected, such a model required adaptation for another speaker. 20 sentences of the new speaker enabled partial adaptation of ASR, so that it reached word accuracy of 94.09%, but better results (WER below 4%) were obtained if the model was adapted with 60 or 80 sentences. Obviously using such a long audio material for adaptation of each new user would be impractical, so the acoustic model needs to be improved. Training the acoustic model on a large multi-speaker speech corpus of the Polish language is planned as the next step.

## REFERENCES

- [1] B. Ziolkowski, S. Manandhar, R. C. Wilson, M. Ziolkowski, J. Galka, *Application of HTK to the Polish Language*, In Proceedings of IEEE International Conference on Audio, Language and Image Processing ICALIP2008, Shanghai, 2008, pp.1759-1764.
- [2] M. Szymanski, J. Ogorkiewicz, M. Lange, K. Klessa, S. Grochowski, G. Demenko, *First evaluation of Polish LVCSR acoustic models obtained from the JURISDIC database*, Speech and Language Technology, vol. 11, 2008.
- [3] S. Furui, *Selected topics from 40 years of research in speech and speaker recognition*, Interspeech 2009, Brighton UK, 2009.
- [4] L. Rabiner and B.-H. Juang, *Historical Perspective of the Field of ASR/NLU*, Springer Handbook of Speech Processing, ed. J. Benesty et al., Berlin Heidelberg: Springer-Verlag, 2008.
- [5] S. Young, *HMMs and Related Speech Recognition Technologies*, Springer Handbook of Speech Processing, ed. J. Benesty et al., Berlin Heidelberg: Springer-Verlag, 2008.
- [6] Sun Microsystems, *Java Speech Grammar Format*, version 1, 1998, <http://java.sun.com/products/java-media/speech/forDevelopers/JSGF/>
- [7] Sun Microsystems, *Java Speech API*, 1998, <http://java.sun.com/products/java-media/speech/>
- [8] Carnegie Mellon University, *SphinxTrain*, <http://cmusphinx.sourceforge.net/wiki/sphinxtrainwalkthrough>
- [9] Ubisoft Shanghai, *Tom Clancy's Endwar*, 2009, <http://endwargame.us.ubi.com/>
- [10] Ubisoft Romania, *Tom Clancy's H.A.W.X.*, 2009, <http://www.hawxgame.com/>
- [11] W. Jassem, *Podstawy fonetyki akustycznej* (in Polish), Warsaw, Poland: PWN, 1973.
- [12] Ron F. Feldstein, *A Concise Polish Grammar*, Duke University, Durham, USA: Slavic and Eurasian Language Resource Center, 2001.
- [13] A. Janicki, P. Meus, M. Topczewski, *Taking Advantage of Pronunciation Variation in Unit Selection Speech Synthesis for Polish*, 3rd International Symposium on Communications, Control and Signal Processing (ISCCSP 2008), St. Julians, Malta, 2008.
- [14] S. Grochowski, *CORPORA—Speech Database for Polish Diphones*, 5th European Conference on Speech Communication and Technology Eurospeech '97, Rhodes, Greece, 1997.
- [15] Alex Rudnicky, *Sphinx Knowledge Base Tool*, 2010, <http://www.speech.cs.cmu.edu/tools/lmtool.html>

# Classification of Learners Using Linear Regression

Marian Cristian Mihăescu  
Software Engineering Department  
University of Craiova  
Craiova, Romania  
Email: mihaescu@software.ucv.ro

**Abstract**—Proper classification of learners is one of the key aspects in e-Learning environments. This paper uses linear regression for modeling the quantity of accumulated knowledge in relationship with variables representing the performed activity. Within the modeling process there are used the experiences performed by students for which it is known the level of accumulated knowledge. The classification of learners is performed at concept level. The outcome is computed as a percentage representing the concept covering in knowledge

**Keywords**—e-learning, linear regression, learner classification

## I. INTRODUCTION

THIS paper addresses the problem of classifying learners according with performed activity. Each learner is described by a set of six parameters. The parameters are of two types. One regards the quality of answers to the test questions and one regards the time in which the answers were provided. The input dataset consists of the data provided by learners who already finished the courses.

The infrastructure on which the analysis process is performed is of hierarchical nature. A discipline is considered to be an aggregate of chapters. Each chapter has an associated concept map [1]. For each concept within the concept map there is associated a set of test questions. During the usage of the e-Learning environment there are recorded necessary actions performed by learners such that a set of parameters may be obtained.

Once a learner obtains a final result for a discipline his experience may be used for building the linear regression classifier. When there are enough learners with complete data regarding their activity we may start using the classifier on new learners.

The classifier may be used for recommendations purposes. Once a student is classified there may be determined a class with better knowledge coverage and thus there may be determined the activities that need more attention such that the student “jumps” into the destination class. The classifier may be also used to predict the knowledge level of a learner at concept level or at discipline level based on learner’s activity and on current classification model.

The presented analysis process enables an e-Learning system to be give advice to learners regarding activities that need to be performed or an estimation of the knowledge level at a certain moment in time.

Activity data has been obtained from Tesys [2] e-Learning platform. The data is offline processed by Weka [3] data mining software which is a collection of machine learning algorithms for data mining tasks.

The second section presents the state of the art regarding presented issues. The third section presents the infrastructure and methods used in analysis process. The first subsection presents the e-Learning infrastructure that has been used for obtaining activity data. The second subsection presents Concept Maps. The third subsection presents the linear regression technique. Section four presents the analysis process. Section five presents a sample experiment with real data. Finally, conclusions and future works are presented.

## II. RELATED WORK

Linear regression is a statistical approach for modeling the relationship between a scalar variable  $y$  and one or more variables denoted  $X$  [4]. There are many domains in which linear regression is used. Trend line, epidemiology and finance are some of the domains in which linear regression is used.

A trend line represents a trend, the long-term movement in time series data after other components have been accounted for. It tells whether a particular data set (say GDP, oil prices or stock prices) have increased or decreased over the period of time. The term “Trend Analysis Report” is found in many domains and provide important knowledge regarding the analyzed data.

In epidemiology there are studies trying to relate tobacco smoking to mortality. There were performed regression analysis studies in order to reduce spurious correlations when analyzing observational data. The simplest approach builds a regression model in which cigarette smoking is the independent variable of interest, and the dependent variable is lifespan measured in years. Of course, other dependent variables such as socio-economic status may be added to show that the effect of smoking on lifespan is not due to any effect of education or income.

Linear regression is not very much used in e-Learning domain. Educational Data Mining [5] is an emerging discipline, concerned with developing methods for exploring the unique types of data that come from educational settings, and using those methods to better understand students, and the settings which they learn in.



There were performed many studies that used statistical and machine learning algorithms on data provided by e-Learning environments. Some of used algorithms are association rules [6], clustering [7], Bayesian networks [8].

Statistical and machine learning algorithms are used to solve important issues of on-line learning environments. Some of the most addressed issues are simulation and modeling learner's interaction [9, 10], prediction of future performance of learners [11], clustering and classification of learners [7].

### III. EMPLOYED INFRASTRUCTURE AND METHODS

#### A. The e-Learning Environment

E-Learning systems are mainly concerned with delivery and management of content (e.g., courses, quizzes, exams, etc.). Since we are speaking about a web platform the client is represented by the browser, more exactly by the learner that performs the actions.

Defining the e-Learning infrastructure or the presented purpose represents the first and the most important step. In this phase, all the possible actions that may be performed by a learner need to be presented. There are also identified the resources that are delivered by the e-Learning system. Finally, there are identified the highly complex business logic components that are used when actions are performed by learners.

Each implemented action needs to have an assigned weight. In the prototyping phase, the assignment of weights is performed manually according with a specific setup. This assumes that we have an e-Learning system that is already set up. The main characteristics regard the number of learners, the number of disciplines, the number of chapters per discipline, the number of test/exam questions per chapter and the dimension of the document that is assigned to a chapter. The data that is obtained from analyzing a certain setup will represent the input data for the simulation procedure.

Another type of activities regarding learners are represented by the communication that take place among parties. Each sending or reading of a message is assigned a computed average weight.

A sample e-Learning setup infrastructure may consist of 500 students, 5 disciplines, 5 to 10 chapters per discipline, 10 to 20 test/exam questions.

For this infrastructure here may be established a list of costs for all needed actions that may be performed by learners. The weight assigned to an action takes into consideration the complexity of the action and the dimension of the data that is obtained as response after the query is sent.

For obtaining reasonable weight, a pre-assessment procedure is performed. The simulation tool performs this procedure from a computer that resides in the same network as the server such that response times are minimal. Each request that is composed and issued to the e-Learning platform is measured in terms of time and space complexity. A scaling factor will assign each action a certain weight such that the scenarios that will be created when real time testing starts will have a sound basis.

The pre-assessment procedure firstly loads all the data regarding the analyzed e-Learning platform. This means the data about all managed resources (e.g. disciplines, chapters, quizzes, etc.) are loaded such that the simulation tool may build valid requests for the e-Learning environment.

#### B. Concept Maps

Concept mapping may be used as a tool for understanding, collaborating, validating, and integrating curriculum content that is designed to develop specific competencies. Concept mapping, a tool originally developed to facilitate student learning by organizing key and supporting concepts into visual frameworks, can also facilitate communication among faculty and administrators about curricular structures, complex cognitive frameworks, and competency-based learning outcomes. To validate the relationships among the competencies articulated by specialized accrediting agencies, certification boards, and professional associations, faculty may find the concept mapping tool beneficial in illustrating relationships among, approaches to, and compliance with competencies [12].

Recent decades have seen an increasing awareness that the adoption of refined procedures of evaluation contributes to the enhancement of the teaching/learning process. In the past, the teacher's evaluation of the pupil was expressed in the form of a final mark given on the basis of a scale of values determined both by the culture of the institution and by the subjective opinion of the examiner. This practice was rationalised by the idea that the principal function of school was selection - i.e. only the most fully equipped (outstanding) pupils were worthy of continuing their studies and going on to occupy the most important positions in society.

According to this approach, the responsibility for failure at school was to be attributed exclusively to the innate (and, therefore, unalterable) intellectual capacities of the pupil. The learning/ teaching process was, then, looked upon in a simplistic, linear way: the teacher transmits (and is the repository of) knowledge, while the learner is required to comply with the teacher and store the ideas being imparted [13].

Usage of concept maps may be very useful for students when starting to learn about a subject. The concept map may bring valuable general overlook of the subject for the whole period of study. It may be advisable that at the very first meeting of students with the subject to include a concept map of the subject.

#### C. Linear Regression and Weka

Linear regression is the most popular regression model [14]. The goal of this model, is to predict the response to  $n$  data points  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  by a regression model given by

$$y = a_0 + a_1x$$

where  $a_0$  and  $a_1$  are the constants of the regression model.

A measure of goodness of fit, that is, how well  $a_0 + a_1x$  predicts the response variable  $y$  is the magnitude of the residual  $\varepsilon_i$  at each of the  $n$  data points.

$$E_i = y_i - (a_0 + a_1x_i)$$

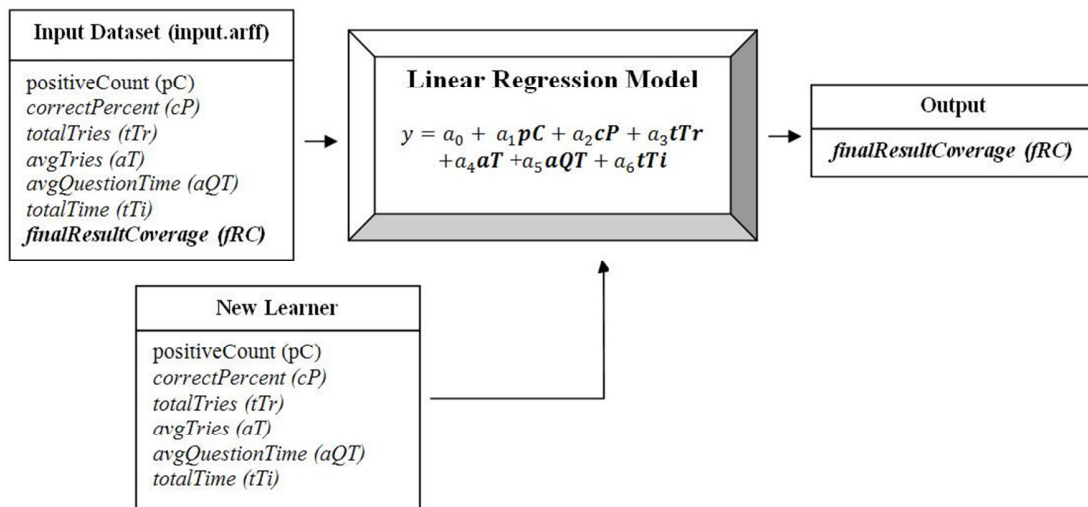


Figure 1. The analysis process

Ideally, if all the residuals  $\epsilon_i$  are zero, one may have found an equation in which all the points lie on the model. Thus, minimization of the residual is an objective of obtaining regression coefficients.

The most popular method to minimize the residual is the least squares methods, where the estimates of the constants of the models are chosen such that the sum of the squared residuals is minimized, that is minimize  $\sum_{i=1}^n E_i^2$ .

Linear regression may be performed on models that have one input variable or multiple input variables. For the multiple input variables the equation has the form:

$$y = a_0 + a_1x_1 + a_2x_2 + \dots + a_nx_n$$

Linear regression finds the parameter values (for the weights  $a_1, \dots, a_n$  and constant  $a_0$  that minimize the sum of the squares of the differences between the actual and predicted  $y$  values.

Weka [15] is a collection of machine learning algorithms. It includes schemes for classification, numeric prediction, meta-schemes and clustering. Linear regression is one of the implemented numeric prediction schemes. Weka uses *arff* file format which require declaration of @RELATION (associates a name with the dataset), @ATTRIBUTE (specifies the name and attribute of an attribute) and @DATA (denotes the start of data segment).

The preprocessing phase in Weka is represented by the necessary actions that load the data. Once the data is loaded there may be performed a linear regression on the dataset. In order to perform this analysis the *LinearRegression* must be chosen. It may be found under *Classify* tab right at *functions* leaf. Finally, the last step to creating our model is to choose the dependent variable (the column we are looking to predict).

#### IV. ANALYSIS PROCESS

The analysis process has four phases. Firstly, the overall procedure is described. This means that the goal of the process is clearly defined. Than, there are defined the

parameters that characterize each instance that build up the dataset. The third step is represented by the effective running of the analysis procedure and obtaining results. Finally, the results are interpreted.

The goal of the analysis process is to predict the knowledge coverage at concept level. A discipline is supposed to be composed of chapters and each chapter has an associated concept map. Each concept has an associated set of quiz questions. We suppose that we have a dataset consisting of past experience of learners regarding quizzes answered related to analyzed concept. For these learners there is known the final result coverage of the concept. Having this data modeled our goal is to estimate the concept coverage using the activity performed by the analyzed learner.

The parameters that characterize each instance are:

- positiveCount* – represents the number of correctly answered questions;
- correctPercent* – represents the percentage of correctly answered questions from the total number of questions;
- totalTries* – represents the total number of tries (answered questions);
- avgTries* – represents the medium number of tries per question;
- avgQuestionTime* – represents, on average, how long (in minutes) it takes for a student to answer a question;
- totalTime* – represents the total time spent on testing;
- finalResultCoverage* – represents the final coverage of the concept. This value is obtained from the final examination data and represents the dependent variable. The value of this variable is known for all learners that participate in building the model. The value of this parameter will be predicted for new learners that provide values only for first six variables.

Figure 1 presents the analysis process. It may be observed that the input dataset is represented by *input.arff* file. In this file resides the data regarding the activity

performed by learners that is used for building the linear regression model. When the input data is fed to the linear regression model builder the final result coverage variable is set as dependent variable. Once the model is created, it may be used to predict the value of the dependent variable provided that values for all other parameters are given. This constitutes the input provided by the learner whose final result coverage needs to be predicted.

This setup uses only normalized and continuous type parameters. That is why the linear regression is chosen from the area of supervised learning algorithms. An important aspect regards the fact that the output variable *finalResultCoverage* is not computed as a formula that takes into consideration the other parameters. The output or predicted variable is obtained from real life examples and thus there is no clear (mathematical) prior dependency between this variable and the rest of variables. This approach makes the experiment to have real consistency regarding the learning process.

## V. SAMPLE EXPERIMENT

The goal of the experiment has already been presented in the analysis process section. The structure of the data is presented in the first section of the *input.arff* file where the attribute names and types are presented. All attributes are of numeric type. This is a constraint imposed by the *LinearRegression* procedure implemented by Weka.

```
@RELATION activity

@ATTRIBUTE positivCount NUMERIC
@ATTRIBUTE correctPercent NUMERIC
@ATTRIBUTE totalTries NUMERIC
@ATTRIBUTE avgTries NUMERIC
@ATTRIBUTE avgQuestionTime NUMERIC
@ATTRIBUTE totalTime NUMERIC
@ATTRIBUTE finalResultCoverage NUMERIC
```

The @data section provides the actual data values. The actual number of learners that are used for building the model is 40. Here is a sample of the dataset corresponding to four learners.

```
@DATA
120,90,133,12.3,45.6,100,71
110,65,71.5,10.3,25.6,180,52
331,27,67,15.8,31.6,56,43
63,92,80,22.6,15.6,36,34
93,31,126,41.6,75.6,60,10
87,62,12,33.8,41.3,41,76
...
```

The first line from @DATA section represents a learner who answered correctly to 120 questions, with a correct percentage of 90%, a total number of tries of 130 and an average of 12.3 number of tries per question. Regarding the time values, the student has an average of 45.6 seconds per question with a total time spent on line of 100 hours. This time represents the whole time

spent on-line within the e-Learning environment. It consists of time spent for testing and for study. At the final examination the discussed concept had a coverage of 71%.

The file with data is set as input data in the preprocessing phase. Then, the *LinearRegression* algorithm is chosen from the functions implemented under *Classify* tab. The *finalResultCoverage* is set as dependent variable. Running in Weka is performed by clicking the *Start* button. Finally, the classifier output is obtained. The output of this analysis process is presented below and represents the linear regression model.

```
==== Run information ====
Scheme: weka.classifiers.functions.LinearRegression
Relation: activity
Instances: 40
Attributes: 7
    positivCount
    correctPercent
    totalTries
    avgTries
    avgQuestionTime
    totalTime
    finalResultCoverage
Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===
Linear Regression Model
finalResultCoverage =
    0.2556 * correctPercent +
   -0.401 * totalTries +
    0.267 * totalTime +
   40.1076

Time taken to build model: 0.03 seconds
==== Cross-validation ====
==== Summary ====
Correlation coefficient      0.2336
Mean absolute error        22.6838
Root mean squared error     28.1816
Relative absolute error     103.337 %
Root relative squared error 109.6054 %
Total Number of Instances   40
```

The above result presents the obtained coefficients of the variables representing the regression output. The interpretation of the pattern regarding the obtained model is the of great importance. Firstly, *positivCount*, *avgTries* and *avgQuestionTime* parameters do not matter. WEKA uses only columns that statistically contribute to the accuracy of the model. It will throw out and ignore columns that don't help in creating a good model. So this regression model is telling us that the number of correctly answered questions doesn't affect the final coverage of the concept.

The total time spent and the correct percentage of correctly answered questions are the parameters that matter.

High number of tries reduce the final coverage of the concept. WEKA is telling us that if learner has a large number of tries his final coverage will be lower. This can be seen by the negative coefficient in front of the *totalTries*

parameter. The model is telling us that every additional try of answering questions reduces the final coverage of the concept by 0.4 percent.

Now, that we have a model we can use it. Let us suppose we have a learner that just had some time spent reading and answering test questions regarding a concept. At this point he may want to know his knowledge coverage of this concept. All we have to do is to feed his data into the model and the the concept coverage will be determined.

The values for the classified learner are: 70 in correctPercent, 60 in totalTries and 50 in totalTime.

Applying the formula we obtain:

$$70*0.2556 + 60*(-0.401) + 50*0.267 + 40.1076 = \mathbf{47.2896}$$

The obtained result needs interpretation. It means that the discussed concept is covered 47.2896 percent by the learner. This is a predicted value obtained by the linear regression model taking into account the previous experiences offered by 40 learners and the current activity performed by the learner for which the prediction is performed.

## VI. CONCLUSIONS AND FUTURE WORK

This paper strives to use a simple data mining technique to obtain knowledge regarding a learner. The goal is to create a model that may be used to predict the knowledge coverage for a learner at concept level. The main outcome of this procedure is that it produces important information for the learner. The created procedure may be adapted for virtually any e-Learning environment with the proper adjustment of the parameters.

Performing such an analysis process needs several things. Firstly, an e-Learning environment is needed. There is also needed data regarding learner's performed activities in a structured manner. This dataset represents the input data for the modeling technique.

A modeling technique is needed. In this paper it is used Linear Regression Modeling implemented by Weka. The outcome of the process is a set of parameters (the coefficients of the linear equation) that may be used to compute a dependent variable.

Once the model is obtained it may be used to compute the value of the dependent variable for a learner that provides values all other parameters. An interpretation of the parameters is needed.

Future works may improve and generalize this procedure. The coverage level may be taken into consideration at chapter or even discipline level.

Another important approach may take into consideration shifting towards discrete values for output variable and even for the other parameters. This may open the window for us-

ing other supervised learning algorithms from the area of classifiers.

Obtained results may be used for building a recommender system that runs along the e-Learning system.

## VII. ACKNOWLEDGMENT

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007 – 2013.

## REFERENCES

- [1] J. D. Novak and A. J. Cañas, "The Theory Underlying Concept Maps and How to Construct and Use Them", *Technical Report IHMC CmapTools*, 2006.
- [2] D. D. Burdescu and M.C. Mihăescu, "Tesy: e-Learning Application Built on a Web Platform", *Proceedings of International Joint Conference on e-Business and Telecommunications*, Setubal, Portugal, pp. 315-318, 2006.
- [3] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA Data Mining Software: An Update", *SIGKDD Explorations*, Volume 11, Issue 1, 2009.
- [4] A. K. Kaw and E. E. Kalu, *Numerical Methods with Applications*, <http://www.autarkaw.com>, second edition, 2010.
- [5] C. Romero and S. Ventura, "Educational Data Mining: A Survey from 1995 to 2005", *Expert Systems with Applications*, 33(1), pp. 135-146, 2007.
- [6] E. García, C. Romero, S. Ventura, T. Calders, "Drawbacks and solutions of applying association rule mining in learning management systems", *Proceedings of the International Workshop on Applying Data Mining in e-Learning*, pp. 1-10, 2008.
- [7] R. Nugent, N. Dean, E. Ayers, "Skill Set Profile Clustering: The Empty K-Means Algorithm with Automatic Specification of Starting Cluster Centers", *Proceedings of The 3rd International Conference on Educational Data Mining*, pp. 151-160, 2010.
- [8] N. Khodeir, N. M. Wanas, N. M. Darwish, N. Hegazy, "Inferring the Differential Student Model in a Probabilistic Domain Using Abduction inference in Bayesian networks", *The 3rd International Conference on Educational Data Mining*, pp. 299-300, 2010.
- [9] M. Mavrikis, "Data-driven modelling of students' interactions in an ILE", *The 1st International Conference on Educational Data Mining*, pp. 87-96, 2008.
- [10] H. Jeong and G. Biswas, "Mining Student Behavior Models in Learning-by-Teaching Environments", *First International Conference on Educational Data Mining*, Montreal, pp. 127-136, 2008.
- [11] H. F. Yu et. al., Feature Engineering and Classifier Ensemble for KDD Cup 2010, *JMLR Workshop and Conference Proceedings*, Invited Paper of KDD Cup 2010 Winner, 2010.
- [12] E. McDaniel, B. Roth, M. Miller, "Concept Mapping as a Tool for Curriculum Design", *The Journal of Issues in Informing Science and Information Technology*, Volume 2, pp. 505-313, 2005.
- [13] L. Vecchia, M. Pedroni, "Concept Maps as a Learning Assessment Tool", *The Journal of Issues in Informing Science and Information Technology*, Volume 4, pp. 307-312, 2007.
- [14] A. Kaw, E. Kalu, *Numerical Methods with Applications*, 2010.
- [15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA Data Mining Software: An Update", *SIGKDD Explorations*, Volume 11, Issue 1, 2009.



## Data Centered Collaboration in a Mobile Environment

Maciej Pańka

Nicolaus Copernicus University  
University Center for Modern  
Teaching Technologies  
ul. Gagarina 17, 87-100 Toruń, Poland  
Email: maciej.panka@umk.pl

Piotr Bała

Nicolaus Copernicus University  
Faculty of Mathematics and Computer Science  
ul. Chopina 12/18, 87-100 Toruń, Poland  
and  
ICM, University of Warsaw  
ul. Pawinskiego 5a 02-106 Warszawa, Poland  
Email: bala@mat.umk.pl

**Abstract**—In this paper we present a system we have developed for a mobile audio-video collaboration that is centered around the distributed datasets. In our approach all the data are processed remotely on dedicated servers, where they are successively rendered off-the-screen and compressed using a video codec. The signals captured from the users' cameras are transferred to the server in real time, where they are combined with the data frames into single video streams. Dependent on the device's capabilities and current network bandwidth every session participant receives individually customized stream, which presents both the remote data and the camera view of currently chosen presenter alternately. At the end of this paper we also present the results of the system's performance test that we have obtained during the collaborative visualization of a remote, multidimensional dataset using different kind of modern mobile devices, including tablets and cell phones.

### I. INTRODUCTION

WITH the dynamic advancement of a broadband Internet access and a pervasive computerization the demand for modern collaboration techniques grows rapidly. First videoconferencing meetings were usually realized in dedicated rooms, equipped with a professional audio-video hardware and a very fast network connection, which allowed for the low latency transfer of the complex multimedia data. For the last couple of years the same functionality have been successively enabled in many software solutions, that could be widely used also in a personal computer environment, e.g. Skype [1], Big Blue Button [2], Open Meetings [3] or Adobe Connect Pro [4]. All of these applications have a built in support for different types of synchronous collaboration techniques, including text based chats, teleconferencing, as well as a fully interactive audio-video communication.

Beside the communication between the session participants, distant collaboration usually involves working with the data. Depending on the discipline, the type of this data can be different and could vary from simple presentations to a very complex multimedia content sharing. Most of the existing videoconferencing systems have a built-in support for different kind of synchronous resource sharing, including white boards or screen capture functionalities. However, these solutions are only sufficient when a single presenter

shares the resources, which are stored locally on his computer.

Unfortunately, none of the existing videoconferencing systems allow the collaboration that would be centered around the distributed datasets, usually stored on different remote servers. One of the most challenging area in this approach is a cooperative visualization of the scientific, multidimensional resources. Most of the modern simulations and experiments are so complex, that they must be realized in dedicated computing centers. The size of the processed data is usually so large that they cannot be easily transferred between distant computers and must be stored in a place where they were generated. On the other hand, many scientific activities require a real time cooperation of researchers representing different disciplines, which should have a possibility to share these data remotely. Moreover, with the growing popularity of wireless networks and ubiquitous computing, this collaboration should also be accessible with the use of different types of mobile devices, including tablets and cell phones.

In this paper we propose a different approach to a video based collaboration, which is centered around the distributed datasets and could be effectively realized in a mobile environment. The system we have developed processes all the data remotely on dedicated servers and transcodes them into a series of digital images representing different views of a visualized resources. Successive frames are compressed using a video codec and synchronously broadcast to all session participants. Users can also broadcast the audio-video signal captured from their cameras and microphones, which is later combined with the visualization frames into a single video stream. This approach takes off all the complex computations from the mobile clients, leaving them only with the video decompression, assuring thereby a highly interactive visualization of the remote data.

This paper is organized as follows. Section 2 covers some of the previous works covering remote data visualization in a collaborative environment. In section 3 we describe in details the system architecture and the technologies we have used to implement it. Section 4 presents the results of the system performance tests and section 5 concludes this paper drawing up further work.

## II. RELATED WORK

There are two general approaches to the collaborative visualization of the distributed datasets. The first category involves systems where the data are transferred from distant servers to session participants, which process them locally using computational power of their devices. In this model a selected session moderator manipulates the remote data using his device, sending thereby successive directives to the main session server. The server broadcasts these directives in real time to the rest of connected users, which in response adequately synchronize their local resources.

The references [5] and [6] present two different systems which derive from this approach and make use of the Virtual Reality Modeling Language, which is a popular text file standard for representing 3D scenes on the Internet. The communication between session participants is realized in separate channels, by means of text chats and teleconferencing modules. The VRML standard has also been successfully adopted into the mobile data visualization. Exemplary solutions were presented in [7] and [8], where the authors introduced two different systems running on Sony Ericsson P800 and PocketPC Compaq iPAQ respectively.

A slightly different approach has been presented in the reference [9], which covers collaborative data visualization in a grid environment. The authors made use of the Interactive Data Language, which is a programming language dedicated mostly to solve 2D / 3D interactive visualization problems. The data synchronization in the proposed solution was realized by means of Narada Brokering Messaging Service. Similar example was also presented in the reference [10], where the authors made use of a shared export concept, which allows capturing of different inputs from a running application and they later broadcast to distant users.

The second category of collaborative visualization solutions involves systems, where all the complex computational tasks are realized on dedicated servers and broadcast to users as a series of graphical images (frames) representing different data views (movement, zooming, 3D objects rotation or animation). When the visualization is realized in a multiuser environment, the image series generated by the server should be broadcast to all session participants simultaneously, allowing thereby a synchronous data sharing [12].

One of the biggest challenges in this approach is a compression of the image data, which in case of very complex resources could be the cause of a network communication latency. Dependent on the dimension of the input data, different compression techniques could be used. The authors of [11] and [12] have developed their systems using popular lossless compression algorithms, including ZLIB, LZO, BZIP2 or RLE. However, higher compression ratios could be achieved with the use of a lossy image compression method, for example a JPEG standard [13]. Similar results could be also obtained with the use of JPEG2000, which is the newer version of the JPEG algorithm. Beside high compression ratios, the JPEG2000 has a built in support for the progressive image transmission and the regions of interest concept, which in context of a mobile visualization could increase the overall efficiency of the system [14, 15].

With the use of the Motion JPEG2000 technique and a JPIP server, the JPEG2000 could also be adopted to the animated data compression. However, in this area, much better results could be achieved by means of a dedicated video codec. The references [16] and [17] introduce exemplary collaborative systems, which generate video sequences on the server and push them individually to every session participant, where they are later decoded and displayed on the screen. The authors made use of the H.263 and H.261 video codecs respectively. A very similar system have been developed by the authors of [18], where the MPEG-4 standard was used for a video compression. A server side encoding was realized by means of the MPEG4IP library, which compressed the video and streamed it to the Apple's Darwin Media Server using RTSP protocol. Apple's server republished this video to the rest of the session participants. The MPEG-4 standard has also been successfully adopted to the remote visualization of a 3D data on different mobile devices [19, 20].

We believe that the most effective approach to the data visualization on mobile devices is by means of the image based streaming techniques, which take off most of the computational power from thin handhelds, leaving them only with data decompression. Modern mobile devices still have many limitations compared to the desktop computers, which obstruct effective rendering of the complex data, e.g. CPU and GPU power, RAM and hard drive capacities, battery lifetime or screen sizes. Unfortunately, none of the existing mobile visualization systems allow fully interactive, real time audio-video collaboration of distant users. On the other hand, currently available videoconferencing systems allow only local data sharing, but they do not support distributed datasets visualization. Moreover, most of these systems were designed in order to cooperate with the desktop computers, and they couldn't be easily adopted to the mobile environment.

## III. SYSTEM ARCHITECTURE

### A. Collaborative Visualization Overview

The system consists of a client application, run on thin mobile devices, and a server application, which processes the complex data. A server is also responsible for the session management and data synchronization between all connected users. The general architecture of our system has been shown on Fig. 1.

All users connected to the server are able to view and listen to the session proceeding but only one of them could additionally be a session presenter. Only the currently chosen presenter is able to control the remote data and broadcast his camera's signal to the rest of participants. For the whole session all users are also able to ask the questions and comment the presentation in front of the group using their devices' microphones.

At the beginning of each session the presenter initializes the remote data on the server, which could be either read from a disk, database or even generated in real time by a dedicated rendering machine. Once the data are loaded by the server they are rendered off the screen, producing there-



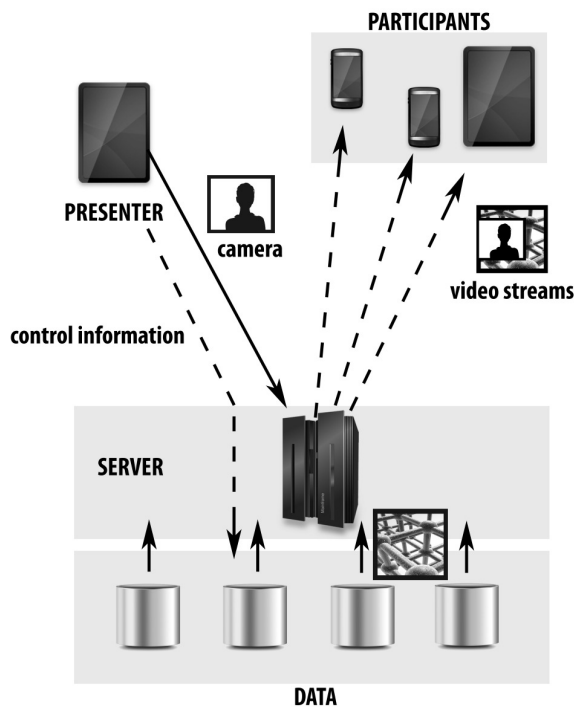


Fig 1. A general architecture of the system.

by a series of digital images, which represent different views of the input resources. Our framework is completely transparent on the data source and their dimensions, which could be either 2D, 3D or even animated data. Two dimensional data are represented as single pictures, while 3D and 4D data produce the whole series of images, which represent data rotation and animation frames respectively.

The generated image sequences are compressed by the server on-the-fly with the use of a dedicated video codec and streamed to all session participants in real time. Every connected user receives his own video stream, individually customized to the capabilities of his device and current network bandwidth. At the beginning of each session the system detects the screen size of a user's handheld, allowing thereby frames scaling on the server. The network bandwidth is also measured during the connection, and it is periodically tested for eventual variations later during the session. Based on these two parameters the server dynamically adapts the resolutions and bit rates of all outgoing video streams.

Once the session is initialized, the presenter is able to manipulate the remote data using his mouse or touch gestures. Dependent on the data dimension the presenter can zoom the view, move to different regions of an image, rotate the 3D model over the X and Y axes or even pause the animation and swap between its successive frames. Adequate directives are sent in real time from the presenter's device to the server, which in response processes the input data, generates adequate video frames and streams them to users. The bit rates of all outgoing videos are changed dynamically by the server

dependent on the current state of the presentation. During the data motion (moving, zooming, rotating) the bit rate is automatically decreased, because there is no need to display the output in full details. Once the motion stops the server encodes the last frame using a higher bit rate, increasing thereby its quality. If more details are needed, users are also able to download the last frame from the server in a completely uncompressed form. This approach saves the network bandwidth and allow highly interactive performance of the system.

In addition to the remote data visualization all session participants are able to collaborate using a built in videoconferencing solution. The signals captured from the presenter's camera and microphone are streamed to the server, where they are processed in real time. The camera signal is decomposed, producing thereby an uncompressed series of images, which are later combined with the visualization frames and audio packets into a single video stream. The session presenter decides which signal should be visible at the moment for the rest of participants: his camera or the remote data view. Dependent on his choice the server encodes appropriate video frames.

Additionally, during the whole session every user is able to individually manipulate currently presented data, e.g. viewing the 3D objects from different angles. In that case only the frames, which represent the visualized data are received from the server. The video signal is additionally combined with the presenter's audio packets, allowing thereby simultaneous lecture listening. In a free look mode every user receives his own instance of the remote data, having a chance to control them independently from the rest of participants. At any point during the session the free look mode could be switched back to the presenter's video transmission.

### B. Server Side Application

A server side application contains of a central session management module and a processing cluster, as shown in Fig. 2. At the beginning of every session users connect to the session management module, which stores the information about data locations and the addresses of all processing units being part of the system. The session manager also acts as a proxy for the audio-video communication and provides different types of real time collaboration techniques, e.g. shared board, which allows synchronous drawing on the presentation and marking its different regions of interest. The central management server is also responsible for the authentication and dynamic roles switching between session participants.

Data generation and audio-video encoding are realized in the processing units, which are built of two different machines working in parallel: an audio-video encoder and a rendering server. Dependent on the number of concurrent users a single processing cluster could be built of more than one pair of the encoding and rendering servers, allowing thereby a better load balancing.

Every collaborative session begins in the central manager module, which at the startup collects the information about all connected clients, including their screen resolutions and current network bandwidths. Based on these parameters and a number of concurrent users it selects the least loaded en-

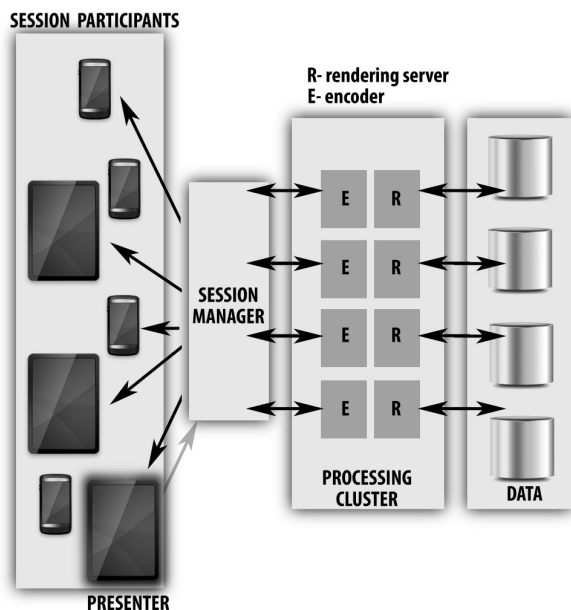


Fig 2. The server side application works in a distributed environment, where every part of the system is run on a different machine.

coding server from the cluster, responsible for further data processing. The selected encoder prepares an individual broadcasting object for each of the outgoing videos, which are run in separate threads, allowing thereby a parallel encoding of multiple video streams on a single machine. In case of many simultaneous users the broadcasting threads could be additionally distributed between different processing units of the cluster.

Successive directives received from the presenter are broadcast in real time by the central management server to every encoder involved in the session. Based on these directives the encoding servers request successive video frames from appropriate rendering machines. Dependent on the data source the rendering machines load appropriate frames from a database, storage device or a dedicated rendering server and streams them to the encoders using a TCP socket connections. During a typical visualization session the renderer and encoder exchange large amount of the image data, so it is recommended that they communicate using a broadband network connection. We have not implemented any compression technique at this point, because it would require an extra CPU power from both servers, which could be the cause of an additional latency.

Successive frames received from the rendering server are passed to the appropriate encoding threads, where they are scaled, compressed using a video codec and broadcast to the session manager, which republishes them to the rest of users. Additionally, during the whole session the presenter broadcasts his video camera signal to the session manager, which republishes it to the encoding servers. Each of the encoders decompress this signal and combine it with the appropriate images received from the rendering server.

The audio-video communication is realized through the session manager, which republishes the live streams in both directions. This approach takes off a lot of computational power from the processing units, reducing the number of the video streams they must encode. For example, when different session users have the same screen resolutions and similar network capabilities the encoder generates only a single video stream, which is later republished by the session manager. On the other hand, with the use of the video proxy, the users are able to dynamically switch between different processing units, without the need of leaving their current session rooms.

### C. Implementation

Our system consists of a client application and a set of distributed server side programs. All server side applications have been implemented using Java language. Additionally, the session manager makes use of the Wowza Media Server, which supports audio-video streams republishing. The communication between the processing units and the session manager is realized with the use of the Adobe's Real Time Messaging Protocol, which allows both a live audio-video streaming and a control information transmission.

The current version of the client application has been developed using the ActionScript 3.0 technology, which with the use of the Adobe's Flash Player could be deployed on most of the modern mobile operating systems, including Google's Android, BlackBerry OS and Apple's iOS (using the Adobe's iPhone packaging library). Video compression is realized with the use of the Sorenson Spark codec, which is an improved version of the H.263 standard, dedicated for a low latency Internet communication. The audio is compressed using the low latency Nellymoser Asao codec. However, the system is completely transparent in sense of the client application technology and could be very easily adapted to any other mobile operating system and audio-video codecs.

## IV. RESULTS

We have run a series of tests of our system, checking both the server and client applications performances. The server side applications have been deployed on three different machines: an Intel Xeon 5050 3GHz running the session management module, a dual core Intel Xeon X5355 2.6 GHz, which was rendering the data visualization frames, and a quad core Intel Xeon E5420 2.5 GHz responsible for the audio-video encoding. A client application was tested on three different mobile devices: a 7" Samsung Galaxy Tab tablet and a 3.7" HTC Desire cell phone, both running the Google's Android 2.2 operating system, and a 10.1" Apple's iPad equipped with the iOS 4.3. The servers were communicating using a 1Gb wired LAN connection, while the client's devices used a standard 802.11g wireless connection. All tests were run for two different sizes of the video streams, 320x240 and 640x480 pixels respectively, which are the most popular resolutions of the video cameras available in modern desktops and mobile devices. All videos were en-

coded using 15 frames per seconds parameter, which is a sufficient setting for a smooth video reception.

The purpose of a client's application test was to measure the CPU usage during three different activities: a single live video decoding, a camera signal encoding, as well as both of them run at the same time. During the third of the above tests the signal captured from the video camera was broadcast to the encoding server, where it was combined with the visualized data frames and pushed back to the same client. However, none of the tests that involved camera signal publishing could be run on the Apple's iPad, because this device has not been equipped with the video camera. The results of the client's application tests are presented in tables 1 and 2.

The video streams decoded on all three devices were displayed very smoothly, consuming less than a half of the processor power in most cases. During the camera signal encoding the average CPU usage increased, but has not affected the overall performance of the system. The iPad's CPU usages during the video decompressions were slightly higher compared to the both Android devices. This is caused by the fact, that Flash Player has not been natively available for the iOS system and our application had to be transcoded into the Objective-C using the Adobe's AIR 2.7 SDK. Nevertheless, we have not noticed any differences in the client's application effectiveness on any of the tested devices.

TABLE I.

AVERAGE CPU USAGES OF THREE DIFFERENT MOBILE DEVICES DURING THE ENCODING AND DECODING OF A 320 X 240 VIDEO STREAMS

Activity	Samsun Galaxy Tab	HTC Desire	iPad
Decoding	30%	32%	55%
Encoding	60%	60%	-
Decoding and encoding	80%	70%	-

TABLE II.

AVERAGE CPU USAGES OF THREE DIFFERENT MOBILE DEVICES DURING THE ENCODING AND DECODING OF A 640 X 480 VIDEO STREAMS

Activity	Samsun Galaxy Tab	HTC Desire	iPad
Decoding	35%	40%	80%
Encoding	90%	85%	-
Decoding and encoding	90%	90%	-

In addition to the client's application performance, we have also measured the efficiency of the encoding server during a sample multiuser session. We experimented with different numbers of simultaneous video streams encodings, which varied from 1 to 60 connections. Beside the CPU usages of the server, we have also calculated the average times of the single video frame encodings and decodings. We have also tried to estimate the highest possible number of concurrent video encodings for a single machine used in this experiment. Tables 3 and 4 present the obtained results.

The maximum number of efficient parallel encodings were 55 and 20 for 320x240 and 640x480 videos resolutions respectively. Below these levels all videos were streamed very smoothly, having less than a half of a second latency. When the number of concurrent users reached the maximum values,

the server's performance lowered and the network communication latency increased, preventing thereby the efficient video encodings. However, with the use of the multiple processing units the overall load of the system could be distributed between different machines, allowing thereby a higher number of simultaneous connections.

TABLE III.

THE PERFORMANCE OF THE ENCODING SERVER DURING DIFFERENT MULTIUSER SESSIONS USING 320 X 240 VIDEO STREAMS

	1 user	20 users	40 users	55 users
CPU usage (quad core)	4%	85%	240%	380%
Single frame decoding time [ms]	0.3	0.5	0.8	6.1
Single frame encoding time [ms]	1.4	1.9	3.6	23.8

TABLE IV.

THE PERFORMANCE OF THE ENCODING SERVER DURING DIFFERENT MULTIUSER SESSIONS USING 640 X 480 VIDEO STREAMS

	1 user	5 users	10 users	20 users
CPU usage (quad core)	15%	75%	240%	380%
Single frame decoding time [ms]	1.3	1.6	3.0	65
Single frame encoding time [ms]	4.8	5.9	12.9	110

## V. CONCLUSION AND FUTURE WORK

In this paper we presented the system for the collaborative visualization of distributed datasets on mobile devices. In our approach the remote data are rendered on dedicated servers, where they are combined with the users' audio-video signals. The output image sequences are compressed using a dedicated video codec and broadcast to all session participants, presenting the visualized data and the camera view alternately. All users can also communicate in real time using a built in teleconferencing solution.

We have also run a series of the performance tests for both the client and server applications. The results we have obtained showed, that our system allows an effective, highly interactive data visualization, even with many simultaneous users connected to a single server. It is also compatible with most of the modern mobile devices, including tablets and cell phones.

In the future we are planning to implement the support for other popular video codecs, including the H.264 and VP8 standards. We also want to improve the server side data processing with the use of a dedicated graphical unit, e.g. NVIDIA CUDA, which should increase the overall performance of the system.

## VI. ACKNOWLEDGMENT

This work has been supported by the eea grant PL-0262.

## REFERENCES

- [1] Skype, <http://www.skype.com/>
- [2] Big Blue Button videoconferencing system, <http://bigbluebutton.org/>

- [3] Open Meetings videoconferencing system, <http://code.google.com/p/openmeetings/>
- [4] Adobe Connect Proc, <http://www.adobe.com/products/adobeconnect.html>
- [5] K. Engel, T. Ertl, „Texture-based Volume Visualization for Multiple Users on the World Wide Web”.
- [6] S. Lovegrove, K. Brodli, „Collaborative Research Within a Sustainable Community: Interactive Multi User VRML and Visualization”.
- [7] M. Mosmondor, H. Komericki, I.S. Pandzic, „3D Visualization of Data on Mobile Devices”, *IEEE MELECON '04*, 2004.
- [8] R. R. Lipman, „Mobile 3D visualization for steel structures”, *Animation in Construction* 13, 119-125, 2004.
- [9] M. Wang, G. Fox, M. Pierce, „Grid-based Collaboration in Interactive Data Language Applications”.
- [10] S. Lee, S. Ko, G. Fox, „Adapting Content for Mobile Devices in Heterogeneous Collaboration Environments”.
- [11] Z. Constantinescu, M. Vladoiu, „Adaptive Compression for Remote Visualization”, *BULETINUL Universitatii Petrol, Gaze din Ploiesti*, vol. LXI, p. 49-58, 2009.
- [12] K. Engel, O. Sommer, T. Ertl, „A Framework for Interactive Hardware Accelerated Remote 3D-Visualization”.
- [13] K. Ma, D. M. Camp, „High Performance Visualization of Time-Varying Volume Data over a Wide-Area Network”, IEEE, 2000.
- [14] D. Dragan, D. Ivetic, „Architectures of DICOM based PACS for JPEG2000 Medical Image Streaming”, *ComSIS*, vol. 6, No. 1, 2009.
- [15] N. Lin, T. Huang, B. Chen, „3D Model Streaming Based on JPEG 2000”.
- [16] K. Engel, O. Sommer, C. Ernst, T. Ertl, „Remote 3D Visualization using Image-Streaming Techniques”.
- [17] M. Hereld, E. Olson, M.E. Papka, T.D. Uram, „Streaming visualization for collaborative environments”.
- [18] F. Goetz, G. Domik, „Remote and Collaborative Visualization with openVisaar”.
- [19] Y. Noimark, D. Cohen-Or, „Streaming Scenes to MPEG-4 Video Enabled Devices”, *IEEE Computer Graphics and Applications*, 2003.
- [20] L. Cheng, A. Bhushan, R. Pajarola, M.E. Zarki, „Real-Time 3D Graphics Streaming using MPEG-4”, 2004.

# Computerized Three-Dimensional Craniofacial Reconstruction from Skulls Based on Landmarks

Leticia Carnero Pascual, Carmen Lastres Redondo, Belén Ríos Sánchez, David Garrido Garrido,  
Asunción Santamaría Galdón  
Universidad Politécnica de Madrid. CeDIInt-UPM. Edif. CeDIInt-UPM. Campus de Montegancedo. 28223  
Pozuelo de Alarcón, Spain  
Email: {lcarnero, clastres, brios, dgarrido, asun}@cedint.upm.es}

**Abstract**—Human identification from a skull is a critical process in legal and forensic medicine, specially when no other means are available. Traditional clay-based methods attempt to generate the human face, in order to identify the corresponding person. However, these reconstructions lack of objectivity and consistence, since they depend on the practitioner. Current computerized techniques are based on facial models, which introduce undesired facial features when the final reconstruction is built. This paper presents an objective 3D craniofacial reconstruction technique, implemented in a graphic application, without using any facial template. The only information required by the software tool is the 3D image of the target skull and three parameters: age, gender and Body Mass Index (BMI) of the individual. Complexity is minimized, since the application database only consists of the anthropological information provided by soft tissue depth values in a set of points of the skull.

## I. INTRODUCTION

THE goal of forensic craniofacial reconstruction is the identification of human and osseous remains, estimating the facial appearance of the individual associated to an unknown skull. It is a critical process in legal and forensic medicine, specially when no other means are available to identify the person [1]. So far, this task has been performed by traditional 'plastic' methods, using clay. This process is carried out by an artist, who models the soft tissue knowing tissue depth in some landmark points on the skull. Depths elsewhere are interpolated between these points by intuition. In that process, replicas of real skulls are used, in order to avoid damaging them [1]. This procedure has important disadvantages: on the one hand, it is an artistic method, which means the process is subjective. In fact, it is non-repeatable, since obtained results always differ between practitioners, and also between reconstructions. On the other hand, the technique is slow, and it usually takes one or two days, even for skilled practitioners. The method is also dirty and expensive, due to the materials required. In fact, a repetition of the process (the creation of a new reconstruction from the same skull using different individual parameters) means the use of new additional materials. Finally, the generated reconstructions are not easily transportable, which makes difficult their distribution or sharing.

This work was supported by the Spanish Ministry of Industry, Tourism and Commerce.

All these disadvantages result in an increasing importance of the computer-based facial reconstruction techniques [2]. Based on this fact, several works and proposals have been developed from the 90's until nowadays [3]-[8]. They all suggest the different advantages of computerized 3D craniofacial reconstruction: it is a consistent process (the output results are the same when the same input data is used), and objective (it does not depend on any practitioner). Moreover, computerized methods can be executed in a short time, and they do not require extra material resources to repeat the reconstruction process from the same skull. All the previous facts make computer-based methods better than traditional procedures.

Current computer-based reconstruction techniques build the final reconstruction starting from a reference facial model. Most published computerized techniques ([3], [4], [8], e.g.) use a generic facial template, or a specific best look-alike template, based on several subject properties (BMI, gender and age). This reference template is then fitted to the target skull knowing tissue thickness in some landmarks on that skull, and interpolating thickness in between these reference facial points, based on a generic smooth deformation. Finally, they add some extra information to improve the results, such as manual modeling features (nose, eyes, etc.) or a texture simulating the skin. The main problem of these procedures is they focus on human resemblance, instead of reliability: using a specific facial template, unwanted facial features of that template remain visible in the final reconstruction. Besides, applying a generic deformation means a problem when the differences between the reference depth tissue values in landmarks and the reference face thickness are considerably large. On the other hand, the results are not skull-specific, but just "smooth".

In attempt to solve the previous shortcomings, some techniques ([6], for example) use specific deformations over the generic facial template. This deformation is obtained from a reference skull, which is deformed towards the target skull. Then, that deformation is extrapolated and applied to the facial template. Other computer-based facial reconstruction proposals (for instance [5], [7], [9]), instead of starting from a generic facial surface, they build a reference statistical model from a database of 3D-scanned real faces. Thus, the problem of unrealistic and unreliable characteristics of the reconstructions is minimized.

All the previous methods and their results are limited, though, in the shape of the reference facial template they use to build the final craniofacial reconstruction: the output results will always contain specific features present in the reference template, which may distort the physical appearance of target person to identify. In addition, those techniques only use the information given by some points on the skull, instead of considering the complete skull surface; this disregards any individual particularity which should affect the final reconstruction morphology. Moreover, all these techniques require high-complexity databases and procedures to perform the final reconstruction. In order to decrease complexity, and to avoid any unsuitable information which may be introduced by that reference facial surface, and also trying to consider as much information as may provide the skull geometry, we propose an alternative computer-based craniofacial reconstruction technique, which is not based on a reference facial template. This reconstruction technique has been implemented in an application, which only starts from anthropological information, consisting in statistical soft tissue depth values in a set of points on the skull. Thus, complexity of the application database is considerably reduced. The only input data required by the application to generate the facial soft tissue mesh are the 3D image of the target skull, and a set of parameters of that skull: age, gender and BMI range. Facial features (eyes, nose, ears and mouth) are not included in the generated reconstruction, as they cannot be confidently deduced only from skull [1].

In the following section, a general description of the application is presented. Section 3 and 4 concentrate on examining the two main functional modules in the application: Landmark Insertion Module and Skin Mesh Generation Module. Then, in Section 5, a comparison of different results is presented. Finally, Section 6 discusses the conclusions of the computer-based facial reconstruction technique here presented.

## II. APPLICATION DESCRIPTION

The application presented in this paper can be represented by the scheme shown in Fig. 1.

The application input data is introduced by the final user (the forensic doctor). It comprises the following elements: firstly, a 3D image of the target skull. Secondly, 66 landmark points placed on the skull surface, where soft tissue depth is known (user will be able to decide whether introducing these positions manually or automatically by the application). The last input data required by the application is a set of characteristic parameters of the target person: age, gender and BMI range. Age and gender can be deduced from skull morphology [1], so they will always be known by the user. However, BMI range will be an unknown parameter, which will have to be estimated.

The Landmark Insertion Module is in charge of placing each landmark point in each position over the skull, and assigning the corresponding soft tissue depth value, according to age, gender and BMI parameters. This functional block will be described in Section 3 of this paper.

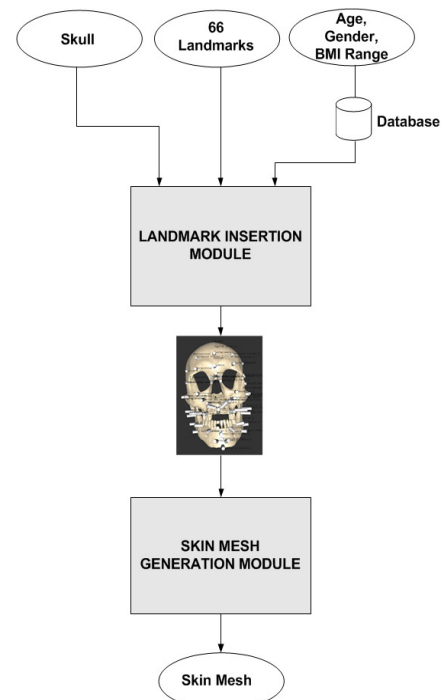


Fig 1: Application scheme

Finally, the Soft Tissue Generation Module is the responsible for generating the skin mesh from the set of landmarks, where soft tissue depth is known. This block will be analyzed in Section 4. The application database consists of a set of tissue depth values in each reference point, varying according to age, gender and BMI attributes. This fact means that the database complexity is very small compared to those consisting of CT images, as in [5], [7], [9]. Consequently, the calculation of tissue thickness values in all the landmark points is very fast.

## III. LANDMARK INSERTION MODULE

The Landmark Insertion Module (LIM) is in charge of placing 66 reference points on the skull surface, and assigning them a tissue depth value, based on a set of parameters of the person: age, gender and BMI range (previously introduced by the user in the system via the graphic user interface). The reference points used for this purpose are two sets of points traditionally used in forensic medicine, as depicted on Fig. 2.

The first set (black points on Fig. 2) results from an anthropological study, presented in [10]. They are compulsory points, since they make possible to generate soft tissue in frontal and lateral sides of a skull. However, that set of points would leave empty the top and the back of the skull. A second set of points has been considered to generate soft tissue around the whole skull (see red points on Fig. 2). That set of points has been selected so that any user can recognize them unequivocally. Moreover, the soft tissue depth variations in that zone can be disregarded, and their magnitude can be approximated by tissue depth in point 1.



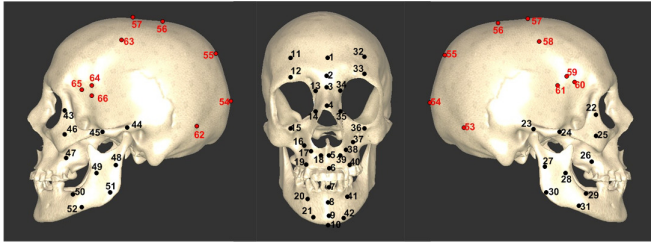


Fig 2: Landmark definition used for craniofacial reconstruction in this work (left, front and right views). In black: set of 52 points to generate the facial reconstruction in facial zone [10]. In red: set of 14 points to generate the craniofacial reconstruction in neurocranium [11]

Based on the previous fact, the system will take 66 positions on a skull 3D image, and the age, gender and BMI range of that person (which will be introduced via the graphic user interface). Then, tissue thickness will be determined in all those reference points according to the parameters of the person. In the following subsections, these two processes (landmark insertion and tissue depth load) will be analyzed.

#### A. Landmark Insertion

The landmark insertion process places the set of 66 reference points on the skull 3D image. Two different ways to accomplish this task have been implemented: manually and automatically.

In the manual procedure, user inserts all landmarks directly on the skull image. The list of 66 reference points is displayed in the graphic user interface, via a combo-box element. The user only has to select one reference point and click on the skull image on its corresponding position. The procedure is order-independent.

In the automatic procedure, all the landmarks will be placed automatically on the image. In order to perform that task, the skull 3D image is projected on the front, right and left planes, and landmark positions are calculated into these projection planes, since those positions are quasi-invariant in every skull. For this purpose, the skull needs to be oriented previously, in order to place it in front position respect to viewer camera. Once all points have been placed on the projected images, an inverse transformation is applied over them to recover the whole 3D image, with all the landmarks placed on it. Likewise, user is allowed to modify any resultant position, if inaccurate.

#### B. Tissue Depth Load

Once all reference points have been placed on the skull 3D image, tissue thickness is assigned to each one, according to the age, gender and BMI range parameters previously introduced. Soft tissue depth values in reference points are known thanks to an anthropological study to characterize tissue depth information of Spanish population, performed by Legal Medicine School of Madrid, and based on a previous study of Belgian population [10]. This study is still in progress, and until this moment, it has been carried out with 160 people, men and woman, aged between 20 and 90 years old. Tissue thickness values are classified into several groups: according to gender, men and women; according to age, five groups can be found: between 20 and 29 years, be-

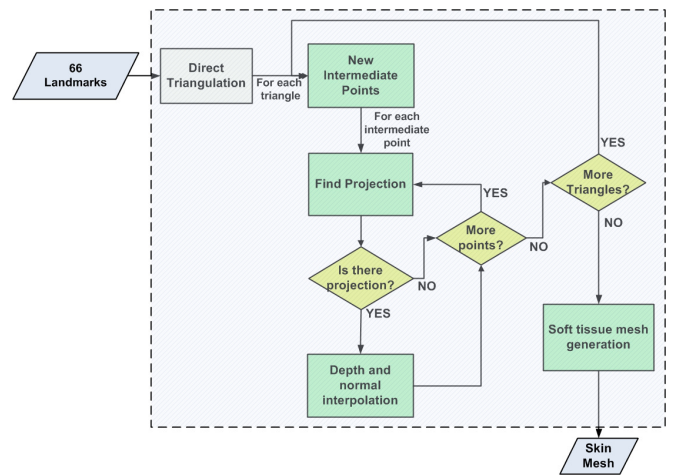


Fig 3: SMGM block diagram

tween 30 and 39, between 40 and 49, between 50 and 59, and older than 60; and finally, according to BMI, population can be divided into 3 groups: people with BMI lower than 20, people with BMI between 20 and 25, and people whose BMI is higher than 25.

Considering this classification, 30 groups of population result. For each one, a tissue thickness mean value is available in the database for every landmark. Based on that fact, and depending on the gender, age and BMI range values of the corresponding person, the system will access to the corresponding entry in the database (population group and landmark), and will assign its depth value to each landmark.

## IV. SKIN MESH GENERATION MODULE

The Skin Mesh Generation Module (SMGM) represents the main functional module in the application here presented. From output data generated in LIM, it manages to construct a full 3D mesh representing soft tissue (skin) belonging to the skull. The general aim of this module is to generate a set of intermediate points on the skull surface, whose depth values can be interpolated from thickness values in reference points. The whole set of points (landmarks and intermediate points) will integrate the final skin mesh. For this purpose, the module receives the set of 66 reference points (their positions and depths), and determines new tissue thickness values in each intermediate point, attending to its location (closeness to the rest of landmarks). For this reason, the presented craniofacial reconstruction technique considers all the information contained in the skull geometry, not only soft tissue thickness in the landmark points. Fig. 3 illustrates this procedure.

Therefore, the main functions participating in the whole process are the intermediate points generation and projection, the interpolation of intermediate depths and normals, and the soft tissue mesh generation. In the subsequent subsections, these four processes will be analyzed.

#### A. Intermediate Points Generation

First step in skin mesh generation process is the creation of a set of new intermediate points, which will integrate the resulting final mesh. Those new intermediate points are created by using a new triangulation (transparent to the user). In



a further process, those new generated points will be projected towards the skull geometry, so that new tissue depth can be obtained on them.

Therefore, the whole process of intermediate point generation comprises two main tasks: the construction of the reference triangle network, and the creation of the new intermediate points in those reference triangles. First task is carried out from the 66 landmarks positions. Then, a manual triangulation is performed, to optimize the amount of resulting triangles, and their shape and distribution. Fig. 4 illustrates the definition of the reference triangle network.

Regarding the second task (creation of new intermediate points), for each resulting triangle, several intermediate positions are calculated, both inside the triangle and on its edges.

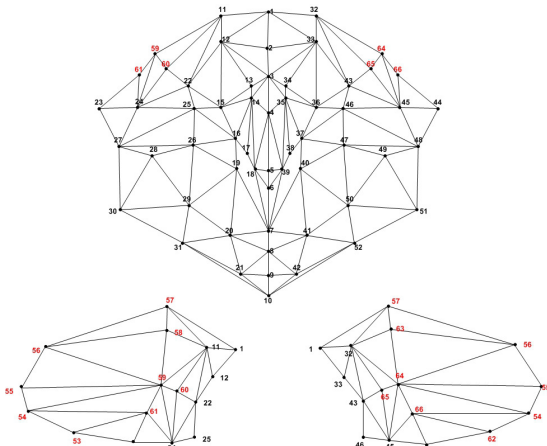


Fig 4: Reference triangle network. The 52 facial points (black), and the 14 extra points (red) are indexed using the same numerical sequence as in Fig. 2

Creation of intermediate points in each edge consists in dividing that edge in equally-sized segments. Generation of intermediate points inside each triangle attends to a regular triangle subdivision [12]. This triangle subdivision is performed according to the *level of detail*, LOD, defined as the number of evaluation points on one edge minus two. The number  $n$  of intermediate points generated inside a triangle can be obtained from LOD, following equation 1. Fig. 5 illustrates this relation.

$$n = \sum_{i=1}^{LOD-2} i \quad (1)$$

### B. Intermediate Points Projection

Once a set of numerous intermediate points has been generated, next step is to project all those points on the skull surface, in order to obtain a set of intermediate points where soft tissue depth can be added. Projection process is different depending on the location of the intermediate point to be projected. Based on this fact, two types of projections are performed: projection of points inside a reference triangle, and projection of points in a triangle edge.

Projection of intermediate points located inside a reference triangle is performed using the normal vector of that tri-

angle. Projection of intermediate points located on an edge will be carried out using the vector defined by equation 2:

$$\vec{p} = \vec{n}_1 + \vec{n}_2 \quad (2)$$

Where  $n_1$  and  $n_2$  are the normal vector of the triangles sharing that edge. In both cases, projection will not be performed unless the condition  $d < d_{max}$  is satisfied, being  $d$  the distance between the original intermediate point and the point projected on the skull mesh, and  $d_{max}$  is a threshold value. This condition prevents an intermediate point from being projected too far from its 3 nearest landmarks, which constitute the reference triangle. This may happen in several regions, for example, inside eye sockets. It is a fundamental condition, since tissue depth associated to the new intermediate points on the skull mesh will be obtained from tissue depth values on those 3 nearest landmarks (as it will be described in section C). Therefore, it is very important to ensure all the projected points have the same landmark neighbors as their corresponding original intermediate point.

### C. Intermediate Depth and normal interpolation

In the previous step, a set of intermediate points on the skull surface were calculated. Next task is to calculate their tissue depth values, in order to obtain the final set of points which will integrate the skin mesh. In all cases, an intermediate skin position can be obtained as:

$$\vec{p}' = \vec{p}_i + \vec{n}_i \cdot l_i \quad (3)$$

Where  $p_i$  is the position vector of the projected intermediate point,  $n_i$  is the normal vector in that point, and  $l_i$  the thickness associated to it. The way to compute  $l_i$  and  $n_i$  will differ, depending on the location of the original intermediate points in the reference triangles.

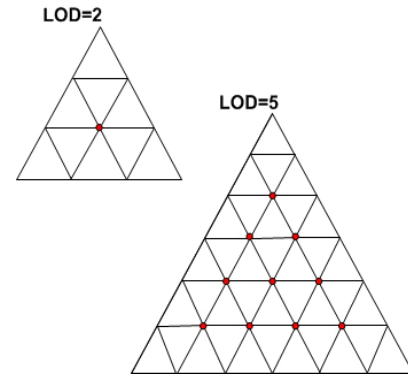


Fig 5: Examples of inner triangle subdivision, with LOD=2 and LOD=5. Intermediate points are highlighted

For skull intermediate points coming from projection of points located inside a reference triangle ( $p_i'$ ),  $n_i$  and  $l_i$  are interpolated from  $l_1, l_2, l_3$ , and  $n_1, n_2, n_3$ , the depth and normal values associated to the three landmarks integrating the reference triangle (influence landmarks). The subsequent equations are used:

$$l_i = u \cdot l_1 + v \cdot l_2 + w \cdot l_3 \quad (4)$$

$$\vec{n}_i = u \cdot \vec{n}_1 + v \cdot \vec{n}_2 + w \cdot \vec{n}_3 \quad (5)$$

$$u = \frac{\text{area}(\vec{p}_1', \vec{p}_2, \vec{p}_3)}{\text{area}(\vec{p}_1, \vec{p}_2, \vec{p}_3)} \quad (6)$$

$$v = \frac{\text{area}(\vec{p}_1, \vec{p}_1', \vec{p}_3)}{\text{area}(\vec{p}_1, \vec{p}_2, \vec{p}_3)} \quad (7)$$

$$w = \frac{\text{area}(\vec{p}_1, \vec{p}_2, \vec{p}_i')}{\text{area}(\vec{p}_1, \vec{p}_2, \vec{p}_3)} \quad (8)$$

For skull intermediate points coming from projection of points located in a triangle edge ( $p_i$ ),  $n_i$  and  $l_i$  are interpolated from  $l_1, l_2$ , and  $n_1, n_2$ , those depth and normal values associated to the two landmarks integrating that edge (influence landmarks). The following equations are used:

$$l_i = u \cdot l_1 + v \cdot l_2 \quad (9)$$

$$\vec{n}_i = u \cdot \vec{n}_1 + v \cdot \vec{n}_2 \quad (10)$$

$$u = \frac{\text{distance}(\vec{p}_i', \vec{p}_2)}{\text{distance}(\vec{p}_1, \vec{p}_2)} \quad (11)$$

$$v = \frac{\text{distance}(\vec{p}_1, \vec{p}_i')}{\text{distance}(\vec{p}_1, \vec{p}_2)} \quad (12)$$

#### D. Mesh Generation

Once the set of intermediate tissue points has been generated (from the original 66 landmarks), the next step is to build a 3D mesh from both sets of points. Considering that the final number of points is greater than 11000, an automatic triangulation algorithm is required. For this purpose, a Delaunay triangulation has been implemented. In Fig. 6, a skull and its point sets (landmarks and intermediate points) are shown. Fig. 7 and Fig. 8 shows some examples of reconstructions obtained from different skulls, by means of the proposed application.

In the previous images, the achievements and limitations of this craniofacial reconstruction method can be detected. Regarding the achievements, zones where skull geometry shows soft variations are well reconstructed; this is the case of the forehead, the chin, the upper part and lateral faces of the nose. However, limitations can be found in zones where skull varies, for example, in zygomatic arch zones and near eyes and mouth, especially. In those zones, a greater number of landmarks would be needed, in order to obtain an appropriate soft tissue depth interpolation.

#### V. COMPARISON OF RESULTS

In this section, different results are presented, in order to prove that the craniofacial reconstruction method here described verifies these four statements: firstly, craniofacial reconstructions are different for different skulls. Secondly, they are different for a certain skull, using different BMI ranges. Thirdly, craniofacial reconstructions depend on the skull geometry, existing correspondance between skull morphology and skin mesh. And finally, the procedure is not subjective, since craniofacial reconstructions only depend on tissue thickness values in the 66 landmarks and skull geometry.

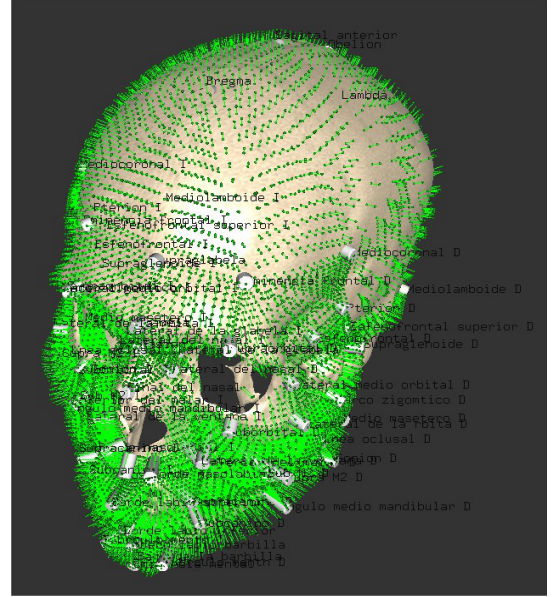


Fig 7: Set of landmarks (white cylinders) and skin intermediate points generated (green spheres) for a generic skull



Fig 8: Examples of reconstructions using the skull of a 27-year-old man. (Left): BMI<20. (Center): 20<BMI<25. (Right): BMI>25

To perform the test, 145 3D-scanned skulls (69 women and 76 men) have been used: 4 samples aged between 20 and 29, 14 samples between 30 and 39, 9 samples between 40 and 49, 18 samples between 50 and 59, and 100 samples older than 60 years old.

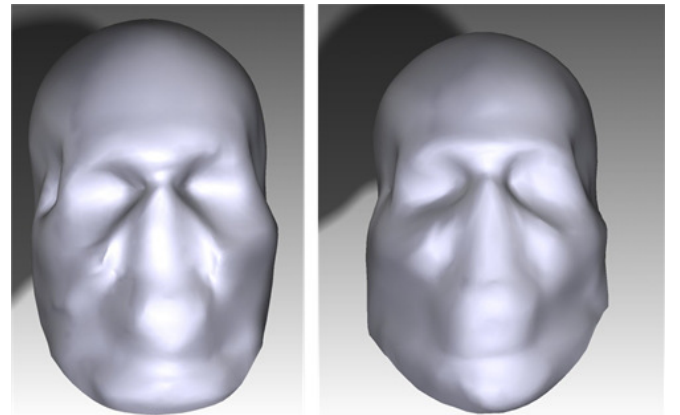


Fig 6: Examples of reconstructions using different skulls. (Left): 53-year-old man with BMI>25. (Right): 34-year-old man with BMI>25

TABLE I.  
LIST OF MEASURES (IN CM.) TAKEN IN 7 SKULLS AND RECONSTRUCTIONS

Individual	Distance 1-10 (facial length)			Distance 23-44 (bizigomatic breadth)		
	Skull	20<BMI<25	BMI>25	Skull	20<BMI<25	BMI>25
25-year-old man	15.19	15.85	15.9	10.99	13.65	14.06
34-year-old man	16.18	17.17	17.26	11.64	13.24	13.85
39-year-old man	15.29	16.27	16.39	12.21	13.85	14.67
45-year-old woman	14.09	15.06	15.15	11.22	13.21	13.57
53-year-old man	15.42	16.62	16.9	12.2	14.23	14.59
73-year-old woman	14.18	15.37	16.45	11.9	13.65	13.87
75-year-old woman	14.51	16.62	16.7	11.15	12.96	13.24

Using these skulls, several reconstructions have been performed varying BMI range values in each skull. In order to contrast objectively all existing changes, two representative measures have been taken in each skull and its corresponding reconstruction: distance between landmarks 1 and 10 (facial length) and between landmarks 23 and 44 (bizigomatic breadth). Table I shows the measures taken from 7 different sample skulls. Measures corresponding to BMI<20 are not presented due to the fact that there are not tissue depth values available in some population groups, since the anthropological database is still being completed.

According to those results, differences between reconstructions belonging to the same skull using different BMI values have been proved, and also differences between reconstructions from different skulls. Based on this fact, the objectivity of the present craniofacial reconstruction method can be ensured.

In order to improve the validation process of the proposed method and test its accuracy, some further tests are being performed to study the variations of the surface in the reconstruction meshes, in comparison with the variations corresponding to real skin meshes (extracted from TAC images). This process is still in progress, since there are not enough TAC images available to construct a reliable sample of real soft tissue.

## VI. CONCLUSION

In this paper, a computerized 3D craniofacial reconstruction technique has been presented. A graphic application has been developed implementing this technique, which enables to generate objectively the soft tissue of any individual, starting only from the skull and a set of 66 reference points where tissue depth is known.

The method consists in generating a great number of intermediate points, where tissue depth is interpolated from tissue thickness in landmark points. This process only comprises projections, normal calculations, arithmetic operations, and finally, a triangulation to build the final reconstruction. Moreover, the complexity of the application database is low, as it only consists of a set of soft tissue thickness values in the reference points. These two facts contribute to the low computational cost of the application.

The presented craniofacial reconstruction technique was developed to ensure the objectivity of the process, since it only considers skull geometry and individual parameters (age, gender and BMI range). On the other hand, it extracts all the local information contained in the target skull, since its entire surface is sampled. This means an important advantage in regions with smooth variations (forehead and chin, for example), where every irregularity will affect the final reconstruction; otherwise, they would be disregarded in case of only considering the set of reference landmarks. However, in places where skull geometry is variant, this tendency to replicate the skull variations is not suitable and it must be improved. In those places, more landmarks would be necessary.

Tests with 145 skulls have been performed, in order to compare the corresponding generated reconstructions. Future work focuses on improving the validation process, comparing reconstructions with real skin meshes extracted from TAC images.

## ACKNOWLEDGMENT

Authors would like to thank the Legal Medicine School of Universidad Complutense de Madrid, for having contributed with their knowledge on forensic anthropology.

## REFERENCES

- [1] C. Wilkinson, *Forensic Facial Reconstruction*. New York: Cambridge University Press, 2004.
- [2] S. D. Greef and G. Willems, "Three-dimensional Cranio-Facial Reconstruction in Forensic Identification: Latest Progress and New Tendencies in the 21st Century," *Forensic Science International*, vol. 50, pp. 12-17, 2005.
- [3] A. W. Shahrom, P. Vanezis, R. C. Chapman, A. Gonzales, C. Blenkinsop, and M. L. Rossi, "Techniques in facial identification: computer-aided facial reconstruction using a laser scanner and video superimposition," *International Journal of Legal Medicine*, vol. 108, pp. 194-200, 1996.
- [4] A. J. Tyrrell, M. P. Evison, A. T. Chamberlain, and M. A. Green, "Forensic three-dimensional facial reconstruction: historical review and contemporary developments," *Journal of Forensic Sciences*, vol. 42, pp. 653-661, 1997.
- [5] P. Tu, R. I. Hartley, W. E. Lorensen, M. Allyassin, R. Gupta, and L. Heier, "Face Reconstructions using Flesh Deformation Modes," in *Computer Graphic Facial Reconstruction*, J. G. Clement and M. K. Marks, Eds.: Elsevier Academic Press, 2005, pp. 145-162.
- [6] G. Subsol and G. Quatrehomme, "Automatic 3D facial reconstruction by feature-based registration of a reference head," in *Computer-graphic Facial Reconstruction*, J. G. Clement and M. K. Marks, Eds.: Elsevier Academic Press, 2005, pp. 145-162.

- [7] D. Vandermeulen, P. Claes, D. Loeckx, S. D. Greef, G. Willems, and P. Suetens, "Computerized craniofacial reconstruction using CT-derived implicit surface representations," *Forensic Science International*, vol. 159, pp. 164-174, 2006.
- [8] R. Liang, Y. Lin, L. Jiang, J. Bao, and X. Huang, "Craniofacial Model Reconstruction From Skull Data Based on Feature Points," in *Computer-Aided Design and Computer Graphics, 2009. CAD/Graphics '09. 11th IEEE International Conference* Huangshan, 2009, pp. 602-605.
- [9] P. Claes, D. Vandermeulen, S. D. Greef, G. Willems, and P. Suetens, "Statistically Deformable Face Models for Cranio-Facial Reconstruction," in *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis— ISPA2005*, S. Lončarić, H. Babić, and M. Bellanger, Eds. Croatia, 2005, pp. 347– 352.
- [10] S. d. Greef, P. Claes, D. Vandermeulen, W. Mollemans, P. Suetens, and G. Willems, "Large-scale in-vivo Caucasian facial soft tissue thickness database for craniofacial reconstruction," *Forensic Science International*, vol. 154, pp. S126-S146, 2006.
- [11] P. M. Moore-Jansen, S. D. Ousley, and R. L. Jantz, *Data Collection Procedures for Forensic Skeletal Material*. Knoxville: University of Tennessee: Forensic Anthropology Center, Department of Anthropology, 1994.
- [12] A. Vlachos, J. Peters, C. Boyd, and J. Mitchell, "Curved PN Triangles," in *2001 symposium on Interactive 3D Graphics* New York, 2001, pp. 159-166.



# DCFMS: A Chunk-Based Distributed File System for Supporting Multimedia Communication

Cosmin Marian Poteras  
University of Craiova  
Software Engineering Department  
Bvd. Decebal 107, Craiova, 200440, Romania  
Email: cpoteras@software.ucv.ro

Constantin Petrisor  
University of Craiova  
Software Engineering Department  
Bvd. Decebal 107, Craiova, 200440, Romania  
Email: costly\_petrisor@yahoo.com

Mihai Mocanu  
University of Craiova  
Software Engineering Department  
Bvd. Decebal 107, Craiova, 200440, Romania  
Email: mmocanu@software.ucv.ro

Cristian Marian Mihaescu  
University of Craiova  
Software Engineering Department  
Bvd. Decebal 107, Craiova, 200440, Romania  
Email: mihaescu@software.ucv.ro

**Abstract**—It is well known that the main drawback of distributed applications that require high performance is related to the data transfer speed between system nodes. The high speed networks are never enough. The application has to come out with special techniques and algorithms for optimizing data availability. This aspect is increasingly needed for some categories of distributed applications such as computational steering applications, which have to permanently allow users to interactively monitor and control the progress of their applications. Following our previous research which was focused on the development of a set of frameworks and platforms for distributed simulation and computational steering, we introduce in this paper a new model for distributed file systems, supporting data steering, that is able to provide optimization for data acquisition, data output and load balancing while reducing the development efforts, improving scalability and flexibility of the system. Data partitioning is being performed at a logical level allowing multimedia applications to define custom data chunks like frames of a video, phrases of text, regions of an image, etc.

## I. INTRODUCTION

**R**EAL time visualization and computational steering are key elements when running a category of applications known as distributed (discrete event) simulation [1] [2]. Generally, simulation refers to the numerical evaluation of a model. It is well known that running simulations on distributed high performance environments might become embarrassingly slow if the analyze phase is performed as a post-processing phase. A simulation has to be exhaustively executed for all input data sets and data can only be analyzed as a post-simulation phase, even if in some cases the process may reveal useless results from the beginning. Therefore, we focused our recent research towards the need to design a high performance distributed simulation framework [3] whose main goal is to optimize scientific simulations. Our framework uses the concept of state-machines for representing general purpose parallel processing tasks and allows the researcher to visualize, analyze and steer the ongoing simulation avoiding irrelevant areas of the simulation process.

The main performance bottleneck that we dealt with while developing the state machine based distributed system (SMBDS) was the data handling itself (acquiring data very fast, dealing with multiple data sources, controlling the network availability, a.o.). Another important aspect that we've noticed was that in many situations it was more efficient to migrate the processing task (state machine) to the host that actually holds the input data than acquiring the data throughout the network. For that we needed a way to query all nodes and find out where the data resides before migrating.

There are many categories of distributed processing applications that demand high data availability. In a distributed environment a set of nodes holds the input data for the entire system. Each node of the system might also become data holder (data storage) as it might output data needed by other nodes in their assigned processing. It comes naturally that the data flow is a crucial factor for achieving the desired performance. It is a nice to have feature that a distributed file system commonly offers. If data flow would be entirely handled at the application level, the entire development process would be significantly slowed down, the application's maintenance would be less flexible, and there would be important doubts on the data transfers efficiency. Obviously this is not a desirable solution for handling data flow in a distributed environment. Instead one could separate the data flow handling into a standalone module whose main role is to acquire, store and provide the data required by the application's processes in the most efficient way.

In this paper we describe the Distributed Chunks Flow Management System (DCFMS) that enables and supports data steering of distributed simulation applications. The system acts like a file system while it adds two new innovative features: logical partitioning and data awareness. Logical partitioning allows the application to define the how the files shall be splitted into chunks. This is very important for avoiding unnecessary transfers of the entire file while only a part of

it is needed, instead the transfers fit exactly the application's needs. The data awareness allow the application to query information related to data location. Most of the times in distributed environments it is desirable to migrate processes towards data than the other way around. This feature allow the application to decide whether to send data towards processes or processes towards data.

The rest of the paper is organized as follows. Section II discusses briefly some related work. Section III-A introduces the new model of DCFMS, as a distributed file system. Section III-B focuses on a description of the support for distributed data flow and load balancing issues in DCFMS and gives some implementation details for Chunker classes. Preliminary performance results are overviewed in Section IV. Section V concludes the paper and outlines the future trends of development.

## II. RELATED WORK

In this section we will mention two of the most popular and widely used distributed data transfer systems which have similar components with the ones introduced in this paper: Apache Hadoop - HDFS and BitTorrent Protocol.

BitTorrent Protocol [4] is a file-sharing protocol designed by Bram Cohen used in distributed environments for transferring large amounts of data. The idea behind BitTorrent is to establish peer-to-peer data transfer connections between a group of hosts, allowing them to download and upload data inside the group simultaneously. The torrents systems that implement BitTorrent protocol use a central tracker that is able to provide information about peers holding the data of interest. Once this data reaches the client application, it tries to connect to all peers and retrieve the data of interest. However, it is up to the client to establish the upload and download priorities. Torrents systems might be a good choice for distributed environments, especially for those based on slower networks. However, the main disadvantages of torrent systems are related to the centralized nature of the torrents tracker as well as leaving the entire transfer algorithms and priorities up to the client application which might cause important delays if the transfers trading algorithm chooses to serve a peer that might have a lower priority at the application level. The reliability of the entire system is concentrated around the tracker; if the tracker goes down, the entire system becomes not functional. Torrents are mainly systems that transfer files in distributed environments in raw format without any logical partitioning of the data. Such logical partitioning might often prove to be very important. For example if an imaging application needs a certain rectangle of an image it would have to download the entire file and then extract the rectangle by itself instead of just downloading the rectangular area and avoid transferring unnecessary parts of the file. As a remarkable advantage of torrents systems we could mention self-sustainability [5] due to peer independence and redundancy.

Hadoop Distributed Files System (HDFS) has been designed as part of Apache Hadoop [6] distributed systems framework. Hadoop has been built on top of the Google's Map-Reduce

architecture and HDFS. HDFS proved to be scalable, and portable. It uses a TCP/IP layer for internal communication and RPC for client requests. The HDFS has been designed to handle very large files that are sent across hosts in chunks. Data nodes can cooperate with each other in order to provide data balancing and replication. The file system depends closely on a central node, the name node whose main task is to manage information related to directory namespace. HDFS offers a very important feature for computational load balancing, namely it can provide data location information allowing the application to migrate the processing tasks towards data, than transferring data towards processing task over the network [7]. The main disadvantage of HDFS seems to be the centralized architecture built around the name node. Failure of the name node implies failure of the entire system. Though, there are available replication and recovering techniques of the name node, this might cause unacceptable delays in a high performance application.

## III. METHODS AND ALGORITHMS

### A. Conceptual Model for Distributed File System

In this section the conceptual model of our distributed file system will be introduced. The system is simple, based on a client-server architecture. The entire model has been built around the key element, data chunk. The data chunk usually represents a file partition but none the less it can be any data object required by the application's processes. Besides the data piece itself, a data chunk also contains meta-information describing the data piece, like: size, location inside source file, the data type, timestamp of latest update or the class that handles chunks of its type.

Figure 1 illustrates the systems model. The most important contribution of DCFMS is that it handles chunks of different types in an abstract mode without actually knowing what is inside the chunk, leaving the data partitioning up to the application level. This is very important from the application's perspective as it can define the way files are partitioned into chunks and how they can be put together again to reconstruct the initial file allowing the application to map data chunks to processing tasks in the most appropriate way for efficient processing. No restrictions are imposed by the DCFMS on data partitioning.

The Type Manager is the bridge between the abstract representation of data chunks and their actual type. The Type Manager is able to make use of external classes where all the file type specific functionality can reside. The classes are dynamically loaded whenever the application layer needs partitioning, files reconstruction as well as information related to the collection of chunks (i.e. the number of chunks). It is the applications' developer task to implement the data chunks handler classes. The DCFMS only provides a set of interfaces that helps the developer implement the partitioning logic.

For example, one might need to handle two types of files in their distributed application: image files and text files. In case of the image files a data chunk might be represented by a rectangular region of the initial image. Multiple such



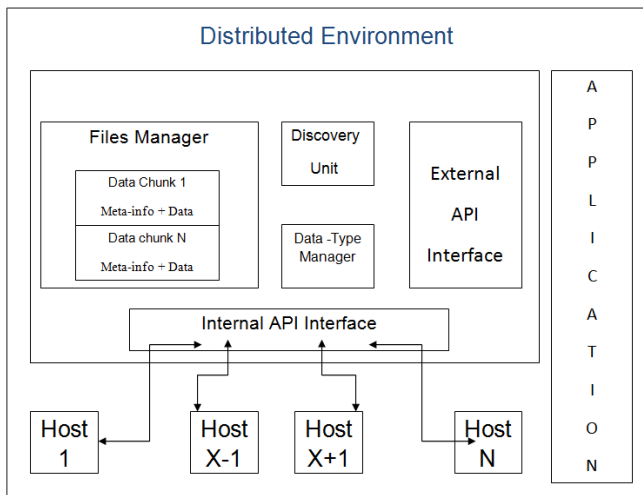


Fig. 1. DCFMS design model

chunks can cover the entire image. An image can be split into rectangular chunks by dynamically invoking the image partitioning method. In case a node needs an entire file that is spread all across the system, DCFMS can acquire all its chunks from different hosts and recompose the image by dynamically invoking the image reconstruction method. In case of a text file, the chunks can take the form of paragraphs, or pages, or simply an array of characters of a certain size. In a similar way the files can be dynamically partitioned and reconstructed.

Later in this paper we will discuss the development effort involved in writing such classes.

The proposed DCFMS is able to scale up dynamically at run time without using a central node. This functionality is achieved by the Discovery Unit which broadcasts and listens to discovery messages.

There are two API interfaces that allow DCFMS nodes to communicate with each other and also with the client application.

### B. Distributed Concepts Support and Implementation

1) *Data flow*: For a better understanding of the data flow algorithm, we will analyze a concrete scenario. Let's assume DCFMS consists of nodes  $N_0, N_1, \dots, N_n$ , and let node  $N_0$  be interested in acquiring data chunks  $C_1, C_2, \dots, C_m$ .  $N_0$  will broadcast a request for  $C_1, \dots, C_m$  to the entire DCFMS. Nodes  $N_1, \dots, N_n$  reply back to  $N_0$  with a subset of  $C_1, \dots, C_m$  that they host. As soon as replies arrive,  $N_0$  builds a chunks availability matrix having as rows the nodes  $N_1, \dots, N_n$  and as columns chunks  $C_1, \dots, C_m$ .  $(N_i, C_j)$  gets valued 1 if the chunk  $C_j$  is available on host  $N_i$ , otherwise it gets valued 0.  $N_0$ 's main goal is to establish as many connections as possible, but not more than one connection per serving host (at most  $n-1$  connections at a time). Chunks availability responses are performed in an asynchronous manner so that  $N_0$  won't have to wait for all responses before proceeding with transfers. Instead it will establish connections as the

responses arrive, overlapping chunks transfer with availability requests. Whenever a chunk transfer completes, the External API will be informed about it and the client application can start processing the newly acquired data. As chunks might spread across DCFMS while  $N_0$  transfers its chunks, the availability matrix will be constantly updated by sending new availability requests whenever a chunk transfer completes and  $N_0$  has established less than  $n-1$  connections (free download slots available).

2) *Support for load balancing*: In distributed applications it often happens that the processing of a data chunk requires less time than the transfer of the data itself. For this reason it might be a good practice to migrate the processing task towards the data than transferring data to the processing host. The DCFMS is able to provide through its external API locating information about the data it holds (data aware system). It is the application's task to migrate the processing tasks throughout the nodes in order to reduce or eliminate the data transfer time.

3) *Application developer's task: Implementing Chunker classes*: Chunker classes define how files or data objects are split into data chunks. A chunker class is nothing else than a class that implements a Chunker interface defining the following methods:

- GetChunk(chunkId)
- IsChunkAvailable(chunkId)
- ReconstructFile(filename)

Chunker classes are dynamically invoked at run time every time chunks or their associated meta-data are being requested. Data chunks are mapped to chunker classes by their meta-data.

## IV. EXPERIMENTAL RESULTS

We conducted a set of experiments for the preliminary performance evaluation of our DCFMS, as a standalone system, out of the scope of the framework in which it will be finally integrated. To evaluate DCFMS we've made use of two environments:

- A high performance Myrinet network of 4 Gbps bandwidth consisting of 8 identical hosts with Intel Core 2 Duo E5200 processors, 1GB of memory and the hard drive benchmarked at an average read speed of 57MB/s and write speed of 45MB/s.
- A regular Ethernet network of 100Mbps bandwidth consisting of 8 identical hosts 4 hosts with Intel Core 2 Quad processors and 4 GB of memory.

The purpose of the experiments was to determine how fast will perform the DCFMS in one-to-many and many-to-one scenarios. We've picked those two scenarios since they represent the worst and the best traffic demanding scenarios. In one-to-many scenario a certain file was hosted by one node and had to be transferred to all the other nodes starting simultaneous. The reverse work had to be performed in a many-to-one scenario, namely all hosts except one hosted the file, and they had to serve collectively the file to the client host. Each scenario has been run for different chunk

TABLE I  
MYRINET ONE-TO-MANY RESULTS

Chunk Size	Min. time 7 hosts (s)	Max. time 7 hosts (s)	Avg. Time 7 hosts (s)
256KB	14.197	16.973	15.730
512KB	9.072	10.291	9.704
1MB	9.110	10.534	9.929
2MB	8.800	9.802	9.404
5MB	9.779	11.926	10.960
10MB	9.221	12.253	10.99
15 MB	9.166	11.529	10.353
20MB	12.444	18.583	15.640
25MB	13.763	18.407	17.066

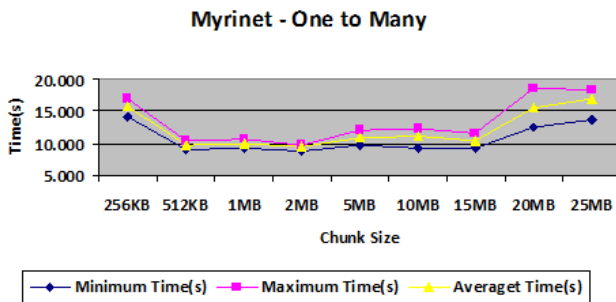


Fig. 2. Myrinet One-to-Many results

sizes. The test cases presented in this paper focus on the transfer speed of DCFMS rather than on the computational performance of a system based on it. Around 50 runs were performed for each case and the results were statistically processed, avoiding singularities. We appreciate that the results can be significantly improved by our DCFMS running in its real design environment instead of a testbed, by using the data awareness capabilities discussed in the previous sections.

#### A. Myrinet Network results:

Test case 1: One-to-Many one sender host and 7 receivers that request the same file of 180.6MB simultaneously. Results are presented in Table I and Figure 2

Test case 2: Many-to-One senders that will serve one host that requests the same file of 180.6MB from all senders. Results are presented in Table II and Figure 3

By examining the results obtained at test cases 1 and 2 we can conclude that the chunk size has an important impact on the file system's performance. To obtain best timings, the Myrinet-based application developer has to choose a chunk size between 2MB and 5MB.

#### B. Ethernet Network results:

As the Ethernet speed is far less than the Myrinet network, we decided to use a small file, namely a 9.8MB file.

Test case 3: One-to-Many one sender host and 7 receivers that request the same file of 9.8MB simultaneously. Results are presented in Table III and Figure 4

TABLE II  
MYRINET MANY-TO-ONE RESULTS

Chunk Size	Time (s)
256KB	11.362
512KB	5.693
1MB	5.466
2MB	4.168
5MB	3.692
10MB	4.111
15 MB	4.216
20MB	5.856
25MB	8.378

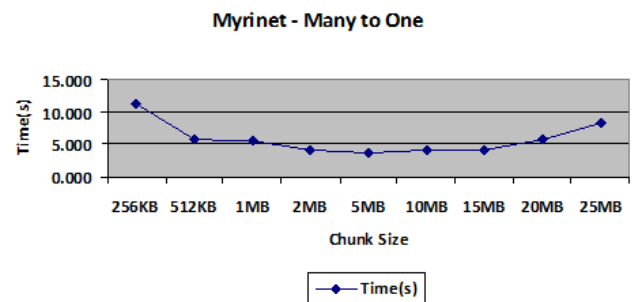


Fig. 3. Myrinet Many-to-One results

TABLE III  
ETHERNET ONE-TO-MANY RESULTS

Chunk Size	128KB	256KB	512KB	1MB	2MB
Minimum time of the 7 hosts (s)	77.241	80.833	64.011	70.572	57.159
Maximum time of the 7 hosts (s)	83.553	85.908	80.121	82.930	75.547
Average Time of the 7 hosts (s)	80.344	83.550	72.994	77.346	67.021

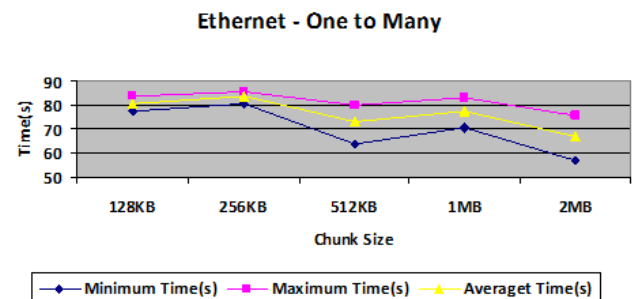


Fig. 4. Ethernet One-to-Many results

TABLE IV  
ETHERNET MANY-TO-ONE RESULTS

Chunk Size	128KB	256KB	512KB	1MB	2MB
Time (s)	29.697	29.048	28.940	28.726	28.752

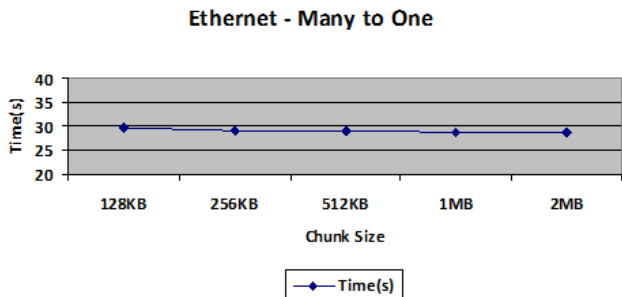


Fig. 5. Ethernet Many-to-One results

Test case 4: Many-to-One 7 senders that will serve one host that requests the same file of 9.8MB from all senders. Results are presented in Table IV and Figure 5

Unlike the Myrinet network, in case of the Ethernet the chunk size doesn't have a big impact on the performance. However the Ethernet network proved not to be the appropriate environment for high performance distributed applications that require high data availability.

V. CONCLUSIONS AND FUTURE WORK

In this paper we've introduced a new model of distributed files systems. The dynamic discovery feature of the system ensures the scalability of the system, the decentralized architecture improves the reliability, while the dynamic data handling offered by the chunker classes make the system flexible and easier to extend. Some other important features of the system, that worths mentioning are: load balancing support due to the data awareness feature and the ability to define chunks that can have any logical meaning.

Important contributions of the system relate to: custom logical partitioning defined at the application level (abstractly

handling) and load balancing support due to the data awareness (data location information) feature while maintaining a high data availability.

The system shows good performance in very high speed networks (Myrinet), but it can also be a good choice in Ethernet networks for applications not requiring transfers of high data volumes across the network.

As future development of the system we could mention the hosts' speed ranking which could be very significant when deciding the source hosts, the network traffic monitoring which could help deciding the route that should be followed for a faster download and none the less the system needs caching techniques.

Being designed as part of a distributed simulation framework [3], as mentioned in the Introduction, DCFMS shall be able to provide support for computational steering. Besides steering the simulation processes the researcher shall be able to also steer data storage, or alter data held by DCFMS while simulation is running. Some algorithms for optimal probabilistic replication of data when the system is in idle state would also be a nice to have feature in our future developments.

REFERENCES

- [1] R. J. Allan and M. Ashworth. A survey of distributed computing, computational grid, meta-computing and network information tools. Daresbury, Warrington WA4 4AD, UK,2001, pp. 38-42
- [2] Esnard, A. Richart, N. Coulaud, O. A Steering Environment for Online Parallel Visualization of Legacy Parallel Simulations.. Proceedings of DS-RT'06 - 10th IEEE International Symposium on Distributed Simulation and Real-Time Applications, 2006, pp.7-14
- [3] Cosmin Poteras, Mihai Mocanu Grid-Enabled Distributed Simulation— A State Machine Based Approach, Proceedings of TELFOR 2010, Belgrade, Serbia, pp. 1323-1326
- [4] Bram Cohen- The BitTorrent Protocol Specification, [http://www.bittorrent.org/beps/bep\\_0003.html](http://www.bittorrent.org/beps/bep_0003.html)
- [5] D. Menasche, A. Rocha, E. de Souza e Silva, R. M. Leao, D. Towsley, A. Venkataramani - Estimating Self-Sustainability in Peer-to-Peer Swarming Systems, Journal of Performance Evaluation Volume 67 Issue 11, November, 2010.
- [6] <http://hadoop.apache.org/>
- [7] Jiong Xie, Shu Yin, Xiaojun Ruan, Zhiyang Ding, Yun Tian, James Majors, Adam Manzanares, and Xiao Qin - Improving MapReduce Performance through Data Placement in Heterogeneous Hadoop Clusters, IPDPSW 2010, Atlanta, pp. 1-9



## Automatic classification of gestures: a context-dependent approach

Mario Refice  
IEEE Senior member  
Department of Electrical  
Engineering and Electronics,  
Polytechnical University of Bari,  
Italy Email: refice@poliba.it

Michelina Savino  
Department of Psychology and  
Educational Sciences,  
University of Bari, Italy  
Email: m.savino@psico.uniba.it

Michele Adduci, Michele Caccia  
Department of Electrical  
Engineering and Electronics,  
Polytechnical Univ. of Bari, Italy  
Email: {adduci.michele,  
michelecaccia84@gmail.com}

**Abstract**—Gestures represent an important channel of human communication, and they are “co-expressive” with speech. For this reason, in human-machine interaction automatic gesture classification can be a valuable help in a number of tasks, like for example as a disambiguation aid in automatic speech recognition. Based on the hand gesture categorization proposed by D. McNeill in his reference works on gesture analysis, a new approach is here presented which classifies gestures using both their kinematic characteristics and their morphology stored as parameters of the templates pre-classified during the training phase of the procedure. In the experiment presented in this paper, an average of about 90% of correctly classified gesture types is obtained, by using as templates only about 3% of the total number of gestures produced by the subjects.

### I. INTRODUCTION

**S**TUDIES on human gestures have been received considerable attention because they represent an important channel in human communication. The seminal work of Kendon [1] has set up a comprehensive scheme of classification and interpretation of human gestures in different languages and cultures. Based on these studies, McNeill [2] developed a detailed interpretation framework, according to which gestures and language are strictly intertwined, as they are “co-expressive” in human communication.

As an almost natural consequence, studies on human gestures have received considerable attention also by applications developers, typically in technological fields like human computer interaction (HCI). Most of these research works have dealt with the problem of recognizing human gestures automatically by means of special equipments (gloves or similar pointing devices) or just taking advantage of the available Computer Vision technology, or even developing new methods for gestures recognition. Among the huge amount of work currently available on this matter, in this paper only a few cases will be recalled, which refer to the mentioned main approaches. In the Gesture Interpretation Module, developed within the project SMARTKOM [3], for example, the gesture channel is combined with other two input

channels, i.e. face recording and speech. The project makes use of a very sophisticated equipment whose main purpose is to provide a useful laboratory environment for multimodal interaction studies. Yingen Xiong & Francis Quek [4] were able, using methods of computer vision, to analyze the hand motion of oscillating frequencies of gestures accompanying speech, and demonstrate that oscillatory gestures reveal portions of the multimodal discourse structure. Andrew D. Wilson & Aaron F. Bobick [5], on the other hand, proposed an extension of the standard HMM method of gesture recognition which shows a better performance in the representation, recognition and interpretation of pointing gestures.

In this paper, a different procedure is described, which aims at classifying hand gestures via a hybrid approach using both the spatial location of the movement and a morphological comparison of the movement with a set of reference templates obtained from the specific context. It makes use of standard equipments, i.e. standard 2D video recording, and is able to classify the main hand gestures of a human being while s/he is talking in a conversation environment. The application domain taken as a reference consists of automatic transcription systems where gestures capture can solve some interpretation ambiguities in the recognition of spoken sentences produced by a talker involved in a conversation with a single interlocutor or in front of an audience, such as in a conference environment.

Since, according to [2], gesture and speech are “co-expressive”, automatic gesture analysis can help in assigning the correct semantic or pragmatic salience during the speech recognition process. In fact, it has been observed that gestural beats are normally synchronized with prosodically prominent syllables in speech (see for example [6], [7], [8]), and that iconic/metaphoric gestures are normally realised in relation to semantically salient words.

As it can be inferred from what discussed above, in our work we deal with communicative gestures only (i.e. speech accompanying gestures), whereas those unconsciously produced – also called “idiosyncratic” gestures – have been obviously not considered.

The methodology adopted in this system is inspired by McNeill's [2] classification of hand gestures into four main categories (iconics, metaphorics, deictics, beats). For gesture identification, instead, it assumes as discrimination factors both the kinematic of the gesture itself and its classification based on a template matching technique.

The system here presented makes use of the OCV [9] package which is an open source set of software modules covering most of the functionalities involved in state of the art video processing techniques.

The recognition phase performs its function in real time and produces an output, which is subsequently analyzed to give the classification of the gestures produced by the subject.

In section II the main outcomes of the McNeill's experimental work, which is the background knowledge of the present application, is recalled. In section III the recognition procedure adopted in this work is presented, along with the classification procedure. Finally, in section IV some results of the developed system are shown and discussed with reference to a specific experiment.

## II. MCNEILL'S CLASSIFICATION SCHEME

As mentioned in section I, some basic assumptions can be made about gestures. First of all, they imply a movement, either of hands or head or some other part of the human body. With this respect the foundations of movements, namely of the hands, has to be acknowledged to McNeill work [2].

According to McNeill, spontaneous movements produced by humans while talking can be classified as:

**Iconics:** "they bear a close formal relationship to the semantic content of speech [...] hand appears to grip something and pull it from the upper front space back and down near to the shoulder." (McNeill, 1992: 12)

**Metaphorics:** "The gesture present an image of the invisible-an image of an abstraction. The gesture depicts a concrete metaphor for a concept [...] Hands rise up and offer listener an object." (McNeill, 1992:14)

**Beats:** "The hand moves along with the rhythmical pulsation of speech [...]. The typical beat is a simple flick of the hand or fingers up and down, or back and forth; the movement is short and quick and the space may be the periphery of the gesture space (the lap, an armrest of the chair, etc.)" (McNeill, 1992:15)

**Deictics:** "[...] is the familiar pointing [...]. Points to space between self and interlocutor" (McNeill, 1992:18)

Using data coming from his experiments, McNeill shows also a diagram of the final position of hand gestures; such diagrams are shown in Fig. 1 a), b), c), d) respectively. Here, dots represent the density of spatial usage for each gesture category.

Even though the McNeill scheme is a descriptive one, mainly having the purpose of describing the psychological background of the producer, the experiment described in the present paper assumes the mentioned classification scheme as one basic assumption in order to classify gestures accord-

ingly to their spatial location; a further discrimination is achieved by trying to match the image of the gesture with some pre-classified examples, which are context, and individual, dependent.

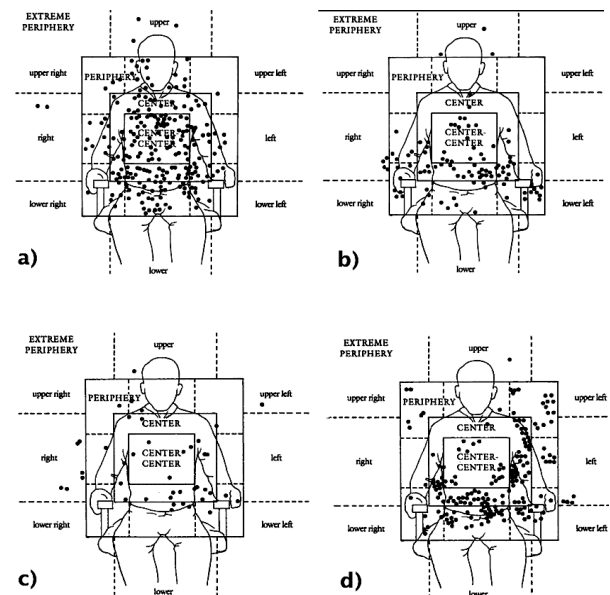


Fig 1. Spatial location and spatial density (dots) of gestures, according to McNeill's schematization, in four gesture types: a) iconics, b) metaphorics, c) deictics, d) beats (reproduced from McNeill, 1992:: 90-91).

## III. METHOD AND PHASES OF THE SYSTEM

Our application system consists of two different and separate phases: the training phase and the identification phase. In the training phase, the database containing the examples – here called templates – is built up through some basic modules, i.e. image acquisition, image preprocessing, features extraction and classification. These templates are used in the second phase for the automatic identification procedure which is based on template matching criteria. The basic features used in this work consist of a number of kinematic parameters, like the position of the centre of gravity of both hands, the speed of their movement, the angle of their movements, and the classification assigned to the template of that movement during the training phase.

As a preliminary stage, we used two videoclips showing two popular moto racers, Jorge Lorenzo and Valentino Rossi, recorded during an interview. These videos are freely available [10] [11]. The two excerpts last about 10 minutes and appear to be recorded under the same conditions. Moreover, in the videos the two subjects wear the same type of dress. In the paper, these two videos will be used as a reference for describing the system in some details.

### A. Image preprocessing and movements classification

The steps which have been considered in this system consist of:



1. skin detection by means of color transformation
2. motion detection by means of background subtraction
3. cleaning up of the resulting image
4. contours extraction by means of edge detection.

The color transformation step aims to represent the color image according to the HSV (Hue, Saturation, Value) scale instead of the standard RGB (Red, Green, Blue). The reason for this conversion is due to the possibility of tuning the computer vision algorithm for each color channel of the image, thus avoiding the correlation effect of the components (Red, Green, Blue) with respect of the light intensity captured by the image. In the HSV representation, these components are uncorrelated each other. Here the following values have been experimentally determined for the three mentioned components:  $0 > H < 20$   $30 > S < 150$   $80 > V < 255$ . These values have demonstrated to better represent the skin characteristics of the two subjects. Result of the skin color detection algorithm is shown in Fig. 2, where the step for background subtraction has also been applied.

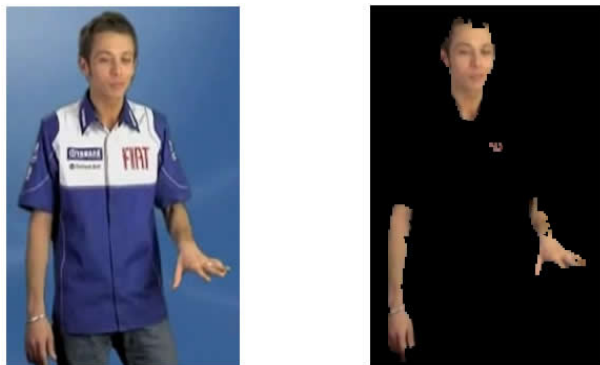


Fig. 2 Skin color detection and background subtraction

This procedure, of course, is not able alone to isolate the subject's hands from the rest, such as face and/or other possible noise of the image, but it eliminates all the static background of it, which is of no interest for the present application. On the other hand, this step allows considering also the head movements, which might be used in further applications.

It is worth noting that the isolation of hand from head movements can be obtained in a quite straightforward way by considering the coordinates of the contours extracted in the step 4 of this procedure, or by using some useful parameters provided by the OpenCv package, selecting only those contours of interest.

Since in this case we are looking for hand movements, a background subtraction allows isolating the parts of the body which have changed position with respect to the previous frame. Since people tend to move their hands more than their

heads, in most of the cases such background subtraction will isolate only the hands from the rest of the image.

At this stage, the image has to be cleaned up for eliminating the noise still present in it, as it appears in Fig. 2. For this purpose, the standard algorithms of "blurring" and "smoothing" are applied. Fig. 3 shows the result of such filtering procedure, where a Gaussian filter and a subsequent threshold operation with a threshold value of 100 (experimentally determined) have been used.

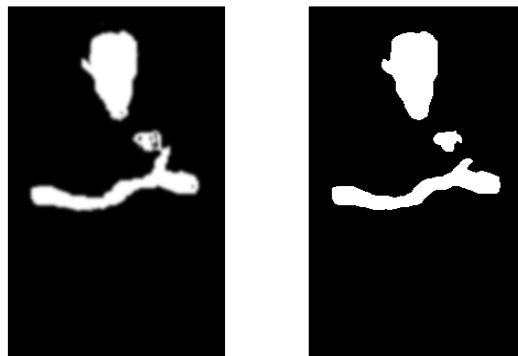


Fig 3. Result from blurring and a threshold filtering of the image

The contours of the image portions of interest are obtained by using the edge detector algorithm proposed by Canny [12]. In our case, a rectangular contour has been considered. Such a contour shape has also the advantage of identifying the Centre Of Gravity (COG) of each hand as the centre of the rectangle. On the basis of COG identification, the relating kinematic parameters, such as speed and angle, can be easily computed.

The result of the edge detection is shown in Fig. 4. It is worth noting that the rectangular shape has the disadvantage of not being sensitive to the hand shape (position/orientation of the fingers), it has nevertheless the advantage of being independent from the hand shape. As a consequence, the information derived so far can be used to identify all gesture types realised by moving hands.

The morphology of the hand (closed vs. open fingers, for example), which can be discriminative for some types of gestures, will be taken into account in a subsequent template matching step described in subsection III.B. It is also important noting that, as shown in Fig. 4, the region detected by the mentioned algorithm does not select the hand but also the arm. This is due to the skin detection method previously described, which is of course not able to distinguish the hand from the arm since they are both characterised by the same skin color. However, this feature does not affect the hand movement classification scheme adopted here.

Using the kinematic information, the simple algorithm shown in Table I allows determining the position of each movement, according to the spatial location of hand gestures proposed by David McNeill and illustrated in section II.



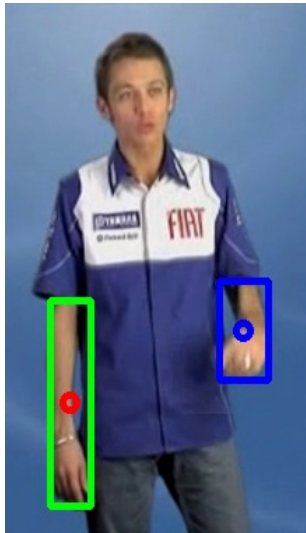


Fig 4. Extraction of contour and Centre Of Gravity of both hands

It is worth noting that hand movements are measured with respect to the rest position. In our case, this position corresponds to the subjects' hands in their pockets.

### I. Templates

As previously mentioned, in this paper a novel aspect of the proposed system consists of assuming that the correct classification of the captured gestures depends also on the predetermined assignment of a class to a gesture prototype. This assignment task is performed through a training phase, and such a task allows the classification to be context- and subject-dependent.

In the training phase, a set of prototypical gestures are selected, classified and stored in a reference database. In order to save the needed computational time, a limited amount of features are extracted by each template, and they are used for the final automatic classification phase.

The features here adopted are the Hu moments [13] of the image.

Generally speaking, for each image a set of moments can be computed using the following definitions:

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y) \quad (1)$$

and the normalised ones as:

$$M_{ij}^{norm} = \frac{M_{ij}}{\sum_x \sum_y I(x, y)} \quad (2)$$

being  $I(x, y)$  the intensity of the pixel  $(x, y)$ .

The moment  $M_{00}$  represents the area of the image, while the centroid of it corresponds to

$$(x_c, y_c) = (M_{10}/M_{00}, M_{01}/M_{00}) \quad (3)$$

and may be assumed also as the Center Of Gravity (COG).

TABLE I.  
ALGORITHM FOR DETERMINING THE POSITION OF THE MOVEMENTS

```

if absccissas_absolute_difference < 30
{
// if the present COG has the same abscissa
// of the previous one

    if ordinal_absolute_difference < 30
    // if the present COG has the same ordinal
    // of the previous one

// the COG has not significantly moved
direction = centre

    else if ordinal_difference > 0
    // the present COG is lower
    direction = to lower
    else if ordinal_difference < 0
    // the present COG is upper
    direction = to upper
}
else if absccissas_difference > 0
{
// if the present COG absissa is towards right

    if ordinal_absolute_difference < 30
    // if the present COG has the same ordinal
    direction = to right
    else direction = unidentified
    // non implemented direction
}
else if absccissa_difference < 0
{
// if the present COG absissa is towards left

    if ordinal_absolute_difference < 30
    // if the present COG has the same ordinal
    direction = to left
    else direction = unidentified
}
else direction = unidentified

```

From these coordinates, the relative moments – also known as central moments – can be obtained, which are translation invariants:

$$\mu_{ij} = \sum_x \sum_y (x - y_c)^i (y - y_c)^j I(x, y) \quad (4)$$

The Hu moments can be derived from the normalized central moments of the image, and the first seven of them have been demonstrated to be able to represent the features of an image, being invariants under different geometrical variations [14].

According to this set of features, four prototypical templates have been chosen for the examples presented in this paper. These templates represent all the prototypical gestures produced by the subjects under examination. For coding advantage, the four templates are numbered from 5 to 8, as shown in Fig. 5.

For tuning purpose of the classification algorithm, a number of counter-examples have also been considered, which correspond to prototypes of the unconscious gestures produced by the subject. In this way, an optimal value of false acceptance vs false rejection (error rate) can be obtained.

For the reported examples, these counter-examples are coded by numbering them from 0 to 4, as shown in Fig. 6. Here a typical case of counter-example is represented by the template 4, where the two hands are connected together, i.e. a hand movement which cannot be certainly considered as intentionally communicative.

The features (Hu moments) of all the coded gestures are included in the database and are used in the classification algorithm.

We have also been developing a user-friendly interactive procedure which allows the system user selecting the most representative frames, computing the feature, storing them in the database, and finding the most useful set of empirical parameters to be used during the recognition phase. The details of such procedure are beyond the scope of this paper and will not be discussed here any further.

### B. The recognition phase

Each frame of the video under examination undergoes the image preprocessing steps described in Section III. A, the kinematic features are computed and the gesture spatial position is determined.

A moving window detects the regions of interest (in this case, hands and/or arms), computes the Hu moments of the

image, and compares them with the stored templates information. Among the possible successful comparisons, the one having the minimum value of the Mahalanobis distance [15] is selected

The matching between any template and the examined gesture is checked by applying the Mahalanobis distance:

$$D_M(x) = \sqrt{(x - y)^T S^{-1} (x - y)} \quad (5)$$

where S is the covariance matrix.

This distance is also known as generalized squared inter-point distance, because it is scale invariant and takes into account the correlations within the data. If it is close to zero, the two vectors are considered similar (or coincident), while they are not if the distance is greater than 1.

Table II shows the classification algorithm of the proposed system. It makes use of both the kinematic parameters computed in this phase (including the gesture spatial position) and the image features stored in the database. In this way, both the spatial classification proposed by McNeill and the template matching approach are taken into account, where the latter provides the needed context and individual variability for the classification.



Fig 5. Templates used for intentional gestures (numbered from 5 to 8)



Fig 6. Templates referring to unconscious gestures (numbered from 0 to 4)

TABLE II  
THE CLASSIFICATION ALGORITHM

```

if ordinal_absolute_difference < 10
    && absissae_absolute_difference < 10
    // small movements are neglected
return "No hands movement ";

if num_template >= -1 && num_template < 5
    && position == undefined
return "Unconscious ";

if num_template == 4 && position == center
    // templates numbered up to 5 represent
    // unconscious gestures
    // such as hand in a pocket or crossed arms
return " Unconscious ";

if num_template == 5 && speed_absolute_difference < 10
    && position == previous_position
    // template related with united hands
    // speed lower than 10 pixel / frame
return "Iconic" ;

if num_template == 6
    && ( position == lower || position == center )
return "Metaphoric" ;

if num_template == 7 && angle_absolute_difference < 0.3
    // angles are measured in radians
    // for small variations (0.3 radians = 10 degrees )
    // the direction is the same
return "Deictics" ;

if num_template == 8
    && ( position == lower || position == to right || position == to left )
    && speed_absloute_difference > 10
    && angle_absolute_difference > 3 )
    // movements larger than 10 pixels / frame
    // 3 r adiants = about 180 degrees , means opposite
    // direction
return "Beat" ;

if no_previous_rule_valid
return "Gesture non recognised-non valid " ;

```

#### IV. RESULTS AND DISCUSSION

In order to test the performance of the system, both in terms of its robustness in classifying gestures and its generality with respect to the used templates, two test trials are here presented.

The first trial classifies the hand gestures produced by Jorge Lorenzo by making use of the templates related to the same subject.

The other one classifies the hand gestures produced by Valentino Rossi, by making use of the templates extracted from Jorge Lorenzo's video instead. As mentioned above, for each classification session a file is automatically produced which can be inspected and statistically analyzed for both evaluation of results and a possible further tuning of the system. The two videos last for about 10 minutes each, and a total of 543 gestures and 354 gestures were produced, respectively.

The prototype gestures used as templates were randomly selected among all hand gestures produced by the subject during the interview. Of course, the frames corresponding to these prototypes have been eliminated from the total amount of gestures analysed in test the procedure. Results of automatic classification are shown in Table III.

TABLE III  
GESTURES CORRECTLY CLASSIFIED

	Deictic	Iconic	Metaphoric	Beat
Jorge Lorenzo (calibration)	99%	None	100%	85%
Valentino Rossi (test)	83%	None	100%	78%

Note that results for iconic gestures are due to the fact that, for this category, the Jorge Lorenzo had realised only one gesture (used as template), whereas Valentino Rossi had never produced iconic gestures during his interview.

As it was expected, the test in autocorrelation gives a better performance but results of the crossed test appears to be also encouraging.

In order to test the accuracy of the system in classifying gestures, we submitted the set of parameters measured in the recognition phase to a Neural Network, and to a clustering algorithm, namely a RBFN (Radial Basis Function Network) and K-mean clustering analysis.

The results of the RBFN model on the Jorge Lorenzo video are reported in Table IV, whereas Table V shows the results of the same process for the Valentino Rossi video.

Unfortunately the well-known K-means algorithm does not provide any figure which is able to give an estimation of the clustering goodness, since the number of clusters is an input parameter for the algorithm. We have performed several runs on the data, adopting a number of clusters spanning from 2 to 12, and found empirically that 5 clusters show the best compromise between the number of clusters and the population of each cluster. This result is confirmed also by the model analyzed by the RBF Network, as previously shown in Tables IV and V. Of course, in the latter case, a considerable amount of computational time is required.

We conclude therefore that the accuracy of the gesture classification produced by our proposed system is compatible with that coming from a Neural Network and a clustering algorithm

However, some considerations need to be pointed out.

First of all, the considered scenes. In the examples presented, the two subjects belong to a scene that never changes, they wear the same dresses and the illumination of the scene does not change during the video recording. This particular situation helps in solving most of the problems which are usually encountered in the automatic tracking of objects. This might appear as a limitation of the proposed approach. On the other hand, these ideal environmental characteristics can be commonly found in the video recordings of a conference speaker, i.e. the kind of application domain we are looking at.

TABLE IV  
GESTURES CLASSIFIED BY RBFN (JORGE LORENZO)

```

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances           443           81.5838 %
Incorrectly Classified Instances        100           18.4162 %
Kappa statistic                         0.6924
Mean absolute error                     0.1041
Root mean squared error                 0.2261
Relative absolute error                  41.7695 %
Root relative squared error              64.1395 %
Total number of Instances               543

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall   F-Measure   Class
0.559     0.055     0.755      0.559    0.643       Deictic
0.903     0.272     0.804      0.903    0.851       Unconscious
0.785     0.021     0.836      0.785    0.81        Beat
0.958     0.002     0.958      0.958    0.958       Unrecognized/Unconscious
1         0         1          1        1           Metaphoric

=== Confusion Matrix ===

  a   b   c   d   e   <-- classified as
71  53   2   1   0 |  a =  Deictic
22 271   7   0   0 |  b =  Unconscious
 1  13  51   0   0 |  c =   Beat
 0   0   1  23   0 |  d = Unrecognized/Unconscious
 0   0   0   0  27 |  e =  Metaphoric
    
```

TABLE V  
GESTURES CLASSIFIED BY RBFN (VALENTINO ROSSI)

```

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances           348           98.3051 %
Incorrectly Classified Instances         6             1.6949 %
Kappa statistic                         0.949
Mean absolute error                     0.0076
Root mean squared error                 0.0588
Relative absolute error                  5.8156 %
Root relative squared error              23.2732 %
Total number of Instances               354

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall   F-Measure   Class
0.979     0         1          0.979    0.99        Unconscious
1         0         1          1        1           Unrecognized/Unconscious
1         0         1          1        1           Beat
1         0.018    0.76       1        0.864       Deictic
1         0         1          1        1           Metaphoric

=== Confusion Matrix ===

  a   b   c   d   e   <-- classified as
284  0   0   6   0 |  a =  Unconscious
 0  14   0   0   0 |  b = Unrecognized/Unconscious
 0   0  25   0   0 |  c =   Beat
 0   0   0  19   0 |  d =  Deictic
 0   0   0   0   6 |  e =  Metaphoric
    
```

Secondly, the approach here proposed strongly relies on the availability of suitable templates which describe the morphology of the gestures. However, the reported examples demonstrate that only a limited amount of templates is needed: 9 templates (4 positive and 5 negative), meaning about 3% of the total amount of gestures to be examined. Moreover, this methodology guarantees the adherence of the classification to the individual subject, and the few templates selected during the calibration phase perform well also for similar environmental conditions. Even the possibility of having a larger set of templates does not affect the performance of the system, since the comparison is made among few parameters (the Hu moments) and does not require a heavy computation time.

We are aware that our system does not solve the general problem of automatically classifying human gestures from a video. On the other hand, we are confident that our simple system can be useful for some specific applications, where its limited computational time can be more effective than more sophisticated, resource-consuming and, in any case, not always highly performing systems

#### REFERENCES

- [1] A. Kendon, "Some relationships between body motion and speech. An analysis of an example", in A. Siegman & B. Pope (Eds.) *Studies in Dyadic Communication*, Elmsford, Pergamon Press, New York, 1972, pp. 172-210.
- [2] D. McNeill, "Hand and mind: what gestures reveal about thought", University of Chicago Press Chicago, 1992
- [3] Rui Ping Shi, Johann Adelhardt, Anton Batliner, Carmen Frank, Elmar Noeth, Viktor Zeissler, Heinrich Niemann, "The Gesture Interpretation Module", *SMARTKOM: Foundations of Multimodal dialog systems*, W. Wahlster Ed., Springer, Berlin 2006, pp.209-219
- [4] Yingen Xiong and Francis Quek, "Hand Motion Gesture Frequency Properties and Multimodal Discourse Analysis", *International Journal of Computer Vision* 69(3), 353–371, 2006.
- [5] Andrew D. Wilson & Aaron F. Robick, "Parametric Hidden Markov Models for Gesture Recognition", *IEEE Trans. On Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, September 1999, pp.884-900
- [6] Y. Yasinnik, M. Renwick, S. Shattuck-Hufnagel, "The timing of speech-accompanying gestures with respect to prosody", in *Proc. of the International Conference "From Sound to Sense"*, MIT, Cambridge, Mass, June 10-13, C97-C102, 2004.
- [7] M. Savino, L. Scivetti, M. Refice, "Integrating Audio and Visual Information for Modelling Communicative Behaviours Perceived as Different", in *Proceedings of Language Resources and Evaluation Conference (LREC 2008)*, Marrakesh 28-30 May 2008 (on CD-ROM).
- [8] A. Esposito, D. Esposito, M. Refice, M. Savino, S. Shattuck-Hufnagel, "A Preliminary Investigation of the Relationship Between Gesture and Prosody in Italian", in A. Esposito, M. Bratanic, E- Keller, M. Marinaro (eds), *Fundamentals of Verbal and Nonverbal Communication and the Biometric Issue*, IOS Press, NATO Security through Science Series, Amsterdam, 2007, pp. 65-74.
- [9] OpenCV : <http://opencv.willowgarage.com/wiki/>
- [10] Jorge Lorenzo video clip : <http://www.youtube.com/watch?v=DJLDn4MOF2Y>
- [11] Valentino Rossi video clip: <http://www.youtube.com/watch?v=d3C8VWrmkcc>
- [12] J. Canny, "A computational approach to edge detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, pp. 679-714, 1986.
- [13] M.K. Hu, "Visual Pattern Recognition by Moments Invariants" – *RE Transaction in Information Theory*, vol. IT-8, pp.179-187, 1962.
- [14] A. Papoulis, *Probability, random variables, and stochastic processes*, McGraw-Hill, 1991.
- [15] P.C. Mahalanobis, "On the generalized distance in statistics", *Proceedings of the National Institute of Sciences of India* 2 (1): 49–55, 1936.

# Concurrency Control Techniques for a Multimedia Database System

Cosmin Stoica Spahiu

University of Craiova

Faculty of Automation, Computers and Electronics

Craiova - Dolj, Romania

Email: stoica.cosmin@software.ucv.ro

**Abstract**—The paper presents the concurrency control methods used to provide simultaneous access to databases, in a multimedia relational database management system. This is an original system that integrates methods for extracting visual characteristics (color and texture characteristics) from images and for executing content-based visual queries. In order to accomplish this, it was defined an original new data type called IMAGE. This data type is used to store the images along with the characteristics extracted and other important information. The problems that should be handled refers to process multiple requests and access the same set of data in a concurrent environment. The databases must be protected with a synchronization algorithm to ensure that the information doesn't get corrupted when multiple clients' requests access concurrently the same set of data.

## I. INTRODUCTION

THE VISUAL data along with other types of multimedia information is very complex. It needs a lot of storage space and it has to permit querying in order to be retrieved from large images collections. To solve all these demands in an efficient way, a multimedia database management system (MMDBMS) is needed.

Most of the systems existing on the market nowadays offer no support at all, or only partial support for the multimedia data. In [5][6] it is proposed an original solution: it is implemented a multimedia database relational system that includes all the algorithms needed to extract color and texture characteristics from images, store them inside the databases and executing visual-based queries.

It is a TCP/IP client-server system based on the SQL language. The system can be used to manage medium sized databases, containing up to several tens of thousands of records.

One of the major objectives of a every DBMS is to allow multiple users to access simultaneously the databases existing on the system.

If a single-user database system is taken into account, the active user can access all the information in the database without any concern that other users could modify the same set of data, at the same time. However, this kind of system has no use in real world. The biggest advantage is when multiple users, executing multiple operations in the same time, can access the same set of data. That is why in such systems it is vital to exist a module for managing concurrency and data consistency [12]. Its main role is to check that all the

operations are executed in such a way that they appear to be executed one at a time (in a serializing mode).

This kind of data sharing implies the existence of specific algorithms for solving conflicts that can appear when the same set of data is accessed. The way these kind of conflicts are solved, depends on the type of the requests taken into account: retrieval or updates.

When discussing about retrieval, no other control is necessary, except the one that is provided by default by the operating system. This is due to the fact that the physical access to disk can be done only in a sequential manner. As long as no user makes any update to the data, it is not important in which order the data is accessed. The operating system will manage all the requests and will specify in each case the concurrency solving algorithm needed in order to minimize the total response time.

If the users' requests imply updating the data (insert, update or delete), it is necessary to have specific algorithms for dealing with concurrent writings. The problems that could appear refer to the cases when several users send requests for modifying the same set of data, or some of them send updating requests and the others send retrieval requests. The simplest way to solve these problems is to block the access to the database while each request is solved. In this way, each request is solved sequentially, but the performances of the system are highly affected.

The paper has the following structure: Section 2 presents the related work, Section 3 presents the MMDBMS and the adopted solution, and Section 4 presents the conclusion and future work in this project.

## II. RELATED WORK AND CONCURRENCY METHODS

The interaction of two or several read/write operations can generate inconsistencies into the database and/or non-valid results (the result obtained in a sequential execution can be different than the result of a concurrent execution). These problems might appear if the same set of data is accessed. Depending to the type of operation that is executed, several types of anomalies can be observed:

- loosing an update (write/write conflict): the update of one operation is lost because of an update of another operation which has not taken into account the result of the first one



- improper reading (write/read conflict): the data is read before the first operation has finished the update and modified the data
- non-repeatable reading (read/write conflict): It is also called "dirty reading". It is met when two consecutive readings of the same data return different results.

The simplest algorithm that can be used to avoid the problems presented above is to restrict the access to database to one operation at a time. The access of all the other operations will be restricted. Two methods are needed in order to implement this: lock(data) and unlock(data).

The types of locks that can be used are: SHARED and EXCLUSIVE locks [1][2]. An exclusive lock is the commonly used locking strategy that provides an exclusive control on the data set. A shared lock can be acquired when a command wants only to read a data set, and not to modify it. If it has already acquired a shared lock on the data set, no other operations can acquire an exclusive lock on that data [1][2].

In order to manage the active locks, a lock manager module should be implemented in order to maintain a list of records for each locked data. The locks will be stored in this list in the order in which they arrive.

It is presented next the strategies used by three well known database management systems for managing multimedia data (images files) and controlling the concurrency: MySQL, Microsoft SQL Server and Oracle Database Server.

#### A. MySQL Server

Most of the systems existing on the market nowadays offers only partial support for managing multimedia data, or no support at all. This is due to the fact that multimedia data needs a lot of disk space, making databases to become huge even for a relative small number of records [8]. In these cases it is recommended to have a special system file structure for the disc (NTFS recommended) and the free space to be carefully supervised.

MySQL does not contains any dedicated data type or methods for images management[9]. The only data type that can be used is BLOB. A BLOB is a binary large object that stores objects in an unstructured manner. BLOB attributes have no character set. The sorting and comparing operations are based on the numeric values of the bytes.

#### B. Microsoft SQL Server

MS SQL Server offers two special data types: image and text. Both data types are treated in a similar manner and no supplementary support is offered. The system does not include any methods for extracting visual characteristics from images or for executing special operations.

More than that, MS SQL Server 2008 recommends to void using these data types, as they will be removed in a future version of the system [11].

The multi-user environment is maintained using two concurrency control techniques: Pessimistic and Optimistic concurrency control techniques. Users specify the type of concurrency control by selecting transaction isolation levels for connections or concurrency options on cursors [11].

When the pessimistic concurrency control technique is used, the locks prevents users from modifying data in a way that could affect the other users. After a user performs an action that activates a lock, the other users cannot perform actions that would conflict with that lock, until it is deactivated by the owner. This is called pessimistic control because it is mainly used in environments where it is high contention for data, and where the cost of protecting data with locks is less than the cost of rolling back transactions when concurrency conflicts occur [10][11].

When optimistic concurrency control technique is used, users do not lock data when they read it. There are two ways for this method to be implemented: optimistic with values and optimistic with row versioning [10].

The optimistic with values method is used when there is only a slight chance that another user to update a row in the interval between when a cursor is opened and when the row is updated. When a user updates data, the system checks to see if another user changed the data after it was read. If another user updated the data, an error is raised. Typically, the user receiving the error rolls back the transaction and starts over. This method is mainly used in environments where there is low contention for data, and where the cost of occasionally rolling back a transaction is lower than the cost of locking data when read [10][11]. In this case the user can deal with occasional error indicating another user has modified the row.

The optimistic with versioning method is based on row versioning. The underlying table must have a version identifier of some type that the server can use to determine whether the row has been changed after it was read into the cursor. In SQL Server, that capability is provided by the timestamp data type, which is a binary number that indicates the relative sequence of modifications in a database. Each time a row with a timestamp column is modified in any way, SQL Server stores the current timestamp. If a table has a timestamp column, then the timestamps are taken down to the row level. The server can then compare the current timestamp value of a row with the timestamp value that was stored when the row was last fetched to determine whether the row has been updated. The server does not have to compare the values in all columns, only the timestamp column. If an application requests optimistic concurrency with row versioning on a table that does not have a timestamp column, the cursor defaults to values-based optimistic concurrency control [10] [11].

#### C. Oracle Database

The Oracle Database System provides the full solution for efficient management and retrieval of multimedia data (images, audio, and video), by using the Oracle Multimedia feature (formerly known as Oracle interMedia) [12].

The images are managed using the ORDImage object data type, which supports the storage, management, and manipulation of images. Each object includes all the attributes, methods, and SQL functions and procedures needed for management.

For concurrency control, The Oracle Database divides the locks types in three main categories [12]:



- DML locks (data locks): used to protect the data. The locks from this category can lock, either the entire table, or only specific rows in the table. Row-Level Locking are used by read committed and serializable transactions. A table lock can be held in any of several modes: row share, row exclusive, share, share row exclusive, and exclusive.
- DDL locks (dictionary locks): are used to protect the structure of objects (e.g.: the structure of the tables)
- Internal locks: are used to protect the datafiles (internal database structures). They are completely automatic and does not need any user interference.

The locks manager can upgrade the locks to a higher level when needed. If a user owns a shared lock (to execute SELECT operations) and at a certain step he executes an UPDATE, the lock will be converted to an exclusive lock [12].

### III. CONCURRENCY MANAGEMENT FOR A MULTIMEDIA DATABASE MANAGEMENT SYSTEM

#### A. General presentation of the system

The implemented system is a databases management system that can be used both for executing simple text-based queries, and more complex content-based visual queries. The content-based visual queries use the color and texture characteristics that were automatically extracted from images, in order to compute the images similarity [5][6][7][8].

This tool is easy to be used because it respects the SQL standard. It does not need advanced knowledge in informatics and has the advantage of low cost. It is a good alternative compared to a classical database management system, which would need higher costs for server acquisition and for designing the applications that execute content-based retrieval operations [5][6][20].

The MMDDBMS permits databases and tables creation, constraints definition, inserting images and alphanumeric information, and executing simple text-based queries or complex content-based queries using color and texture characteristics.

An original element of the system is a new data type that was defined, called IMAGE. This type is used to store both the image itself and the vectors of characteristics (texture and color histogram) [5][6].

Another original aspect is that the system integrates all the algorithms needed to process the images, extract the characteristics and execute retrieval queries based on the content.

When discussing about multimedia data, especially images, it is not important to find an exact match between two images. It is more important to be able to find similar images. There are many algorithms that can be used for processing the images and extracting the color and texture characteristics, but there is not any certain method that can be considered to provide the best results in any situation. The quality of the results depends to the type of images taken into account [21].

Our system is designed to be used mainly in medical domain where the experiments indicated that the best results were obtained using Gabor filters [3][16][17] for texture characteristic and the histogram representation quantized to 166 values, for color characteristic [20][8].

The similarity was computed using Euclidian distance for the texture characteristic and histograms intersection for the color characteristic. The users have the possibility to choose for each executed query what characteristic to be used to compute the similarity: only texture, only color, or both of them (each with an weight of 50%)[5].

#### B. Concurrency management

The second important aspect of this MMDDBMS is that it provides simultaneous access to information for many clients via TCP/IP network. The problems that should be handled refers to process multiple requests and access the same set of data in a concurrent environment.

The system must include a synchronization algorithm to ensure that the information doesn't get corrupted when multiple clients' requests access concurrently the same set of data. However, in most of the cases the information is frequently read and only occasionally written. It is far more efficient to allow all reading requests to be executed simultaneously and only write requests to be executed in an exclusive manner.

The locking mechanism that was chosen for the system is based on L. Lamport's bakery algorithm [2][22][23]. This algorithm was chosen because it offers a good balance between performances and implementation complexity.

There are two types of locks used: shared locks used for reading (e.g.: SELECT) and exclusive locks used for writing (e.g.: INSERT). These types of locks are used only at the table level of granularity. There are not defined row-level locks or others locks at a higher level of granularity.

If a SELECT command is retrieved (that implies reading from database), a read-lock will be enabled on the tables (files) involved in the operation. This lock will be active until the tables (files) will no longer be used. It is a non-exclusive lock, meaning that all other reading requests will be permitted, each of them activating their own read-lock [7].

If an INSERT command or other command that involves writing into database will be received meanwhile, it cannot be executed. No writes are permitted while any read-lock is active. Instead it will be put in a waiting queue for a random period of time. The write operation can be executed only when no other lock is active. After all locks are inactivated for a specific table, the write-lock can be activated. This type of lock is an exclusive one. No other request (read or write) can be accepted while this is active [7].

When an operation activates a lock, it can include one or several tables. If there is no foreign key defined on the requested table, only one table will be locked. If the table includes foreign keys, all the connected tables will be locked using the same type of lock for all of them.

In order to override the critical section when locks are activated or upgraded, it is used the Lamport's bakery synchronization algorithm [23]. This way it is not possible for two different users to lock accidentally the same resources.

When a lock is no longer needed, it will be deactivated directly without using any synchronization algorithm.

The basic idea for the Lamport's bakery algorithm is quite simple. Each user's request receives a serving number when a lock is needed. The holder of the lowest number is the next one that gets access to resources [23]. The implementation of the algorithm is presented next[24]:

```
Algorithm 1. The Bakery Algorithm
waiting[i] <- true;
No[i] <- max(No[0], ..., No[n-1])+1;
waiting[i] <- false;
for j <- 0 ... n-1 do {
    while waiting[j] do nothing;
    while (No[j] != 0) and
        (No[j] < No[i])
        do nothing;
}
*ENTER critical section:
No[i] <- 0;
* Activate requested lock
```

To implement this algorithm, there are need two lists. There is one entry in each list for every lock request. The first array stores the priority number. The other list contains a boolean value for each request specifying if that request is in line to receive a number.

When a new lock request arrives and needs to be enabled, first it sets its boolean value to true. Then it is assigned the next number available for waiting its turn. After it receives a number, its no longer waiting so it sets its waiting value to false. Next, the lock request goes through the first list and if there is a request with a lower number, or a request that's waiting for a number, it waits until that request is finished or assigned a higher number. After the lock manager traverses the list it searches for the request with the lowest number in order to be served and activate the lock [24].

#### IV. CONCLUSION

The paper presented the way concurrency is managed in an original implementation of a multimedia database management system. This system has integrated methods for extracting the color and texture characteristics from images and executing content-based visual queries. In order to accomplish this, it was defined a new data type called IMAGE that is used to store the images along with the extracted characteristics and other important information.

The problems that should be handled are: processing multiple requests and accessing the same set of data simultaneously. The system must include a synchronization algorithm to ensure that the information doesn't get corrupted when multiple clients' requests access concurrently the same set of data. The adopted solution uses a read/write locking mechanism that is based on Lamport's bakery algorithm for entering into critical section and activating the locks. When a lock is activated, the whole table is locked. There are not defined other levels of granularity.

In the future work, the system will include other types of locks defined at the row level. The DBMS will automatically

chose what is the best type of lock that should be used for each request in part.

#### V. ACKNOWLEDGEMENT

The support of the The National University Research Council under Grant CNCISIS IDEI 535 is gratefully acknowledged.

#### REFERENCES

- [1] IITL Education Solutions Limited, *Introduction to Database Systems*, Pearson Education India, 2008.
- [2] A.S. Tanenbaum, *Modern Operating Systems (Second Edition)*, Prentice Hall, 2001.
- [3] A. Del Bimbo, *Visual Information Retrieval*, Morgan Kaufmann Publishers. San Francisco USA, 2001.
- [4] T. Gevers, *Image Search Engines: An Overview*, Emerging Topics in Computer Vision. Prentice Hall, 2004.
- [5] C. Stoica Spahiu, *A Multimedia Database Server for information storage and querying*, International Symposium on Multimedia - Applications and Processing (MMAP'09), Vol. 4, pp. 517 - 522, 2009.
- [6] C. Stoica Spahiu, L. Stanescu, D.D. Burdescu, and M. Brezovan, *File Storage for a Multimedia Database Server for Image Retrieval*, The Fourth International Multi-Conference on Computing in the Global Information Technology (ICCGI 2009), pp.35-40, 2009.
- [7] C. Stoica Spahiu, *Testing the performances of a multimedia database system*, International Journal of Computer Science and Applications (IJCSA), 2011 (to be published).
- [8] C. Stoica Spahiu, L. Stanescu, D.D. Burdescu, M. Brezovan, *Visual Interface for Content-based Query in Image Databases*, Intelligent Interactive Multimedia Systems and Services (KES 2009), Vol. 226, pp. 231 - 240, 2009.
- [9] *MySQL 5.0 Reference Manual* <http://dev.mysql.com/doc/refman/5.0/en/index.html>
- [10] K. Delaney, F. Guerrero, *Database Concurrency and Row Level Versioning in SQL Server 2005* <http://msdn.microsoft.com/en-us/library/cc917674.aspx>
- [11] *SQL Server 2008 Books Online*, January 2009 <http://msdn.microsoft.com/en-us/library/ms187875.aspx>
- [12] *Oracle9i Database Online Documentation (Release 2 (9.2))* [http://download.oracle.com/docs/cd/B10501\\_01/server.920/a96524/c21cnsis.htm](http://download.oracle.com/docs/cd/B10501_01/server.920/a96524/c21cnsis.htm)
- [13] C. Carson, S. Belongie, H. Greenspan, J. Malik, *Blobworld: Image segmentation using expectation-maximization and its application to image querying*, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24(8), pp. 1026-1038, 2002.
- [14] D. Comaniciu, P. Meer, *Robust analysis of feature spaces: color image segmentation*, IEEE Conference on Computer Vision and Pattern Recognition, pp. 750-755, 2003.
- [15] M. Cooper, *The tractibility of segmentation and scene analysis*, International Journal of Computer Vision, Vol. 30(1), pp. 27-42, 1998.
- [16] J.J. Henriksen, *3D surface tracking and approximation using Gabor filters*, 2007.
- [17] M. Lindenbaum, R. Sandler, *Gabor Filter Analysis for Texture Segmentation*, Technical Report CIS-2005-05, Technion - Computer Science Department, 2005.
- [18] M. M. Martnez, *An introduction to content-based information retrieval by normalized compression distance*, Master Thesis, 2009.
- [19] R. Yong, H. Thomas, S. Chang, *Image Retrieval: Current Techniques, Promising Directions, and Open Issues*, Visual Communication and Image Representation, Vol. 10, pp. 39-62, 1999.
- [20] L. Stanescu, D.D. Burdescu, M. Brezovan, C. Stoica Spahiu, and A. Ion, *A New Software Tool For Managing and Querying the Personal Medical Digital Imagery*, International Conference on Health Informatics, pp. 199-204, 2009.
- [21] H. Tamura, T. Mori and T. Yamawaki, *Textural Features Corresponding to Visual Perception*, SMC, Vol. 8, pp. 460-473, 1978.
- [22] G.L. Peterson, *Myths About the Mutual Exclusion Problem*, Information Processing Letters, Vol. 12(3), pp. 115116, 1981.
- [23] L. Lamport, *A New Solution of Dijkstra's Concurrent Programming Problem*, Communications of the ACM 17(8), pp. 453-455, 1974.
- [24] J. Emerson, *Thread Synchronization and Critical Section Problem*, <http://www.jonemerson.net/dev/articles/ThreadSynchronizationAndSemaphores.html>

# Automated annotation system for natural images

Gabriel Mihai, Liana Stanescu  
University of Craiova,  
Faculty of Automation,  
Computers and Electronics,  
Bvd. Decebal,  
No.107, Romania.  
{ mihai\_gabriel, stanescu }@software.ucv.ro

**Abstract**—Automated annotation of digital images remains a highly challenging task. This process can be used for indexing, retrieving, and understanding of large collections of image data. This paper presents an image annotation system used for annotating natural images. The proposed system is using an efficient annotation model called Cross Media Relevance Model for the annotation process. Image's regions are described using a vocabulary of blobs generated from image features using the K-means clustering algorithm. Using SAIAPR TC-12 Dataset of annotated images it is estimated the joint probability of generating a word given the blobs in an image. The annotation process of each new image starts with a segmentation phase. An original and efficient segmentation algorithm based on a hexagonal structure is applied to obtain the list of regions. Each meaningful word assigned to the annotated image is retrieved from an ontology derived in an original manner starting from the hierarchical vocabulary associated with SAIAPR TC-12 and from the spatial relationships between regions.

**Keywords**—Image annotation, image segmentation, ontology, relevance models.

## I. INTRODUCTION

THE automated task used to assign semantic labels to images is known as automatic image annotation. The importance of this task has increased with the growth of the digital images collections. It is a challenge that has been identified as one of the hot-topics in the new age of image retrieval [26]. Image annotation is a difficult task for two main reasons: semantic gap problem - it is hard to extract semantically meaningful entities using just low level image features and the lack of correspondence between the keywords and image regions in the training data.

Representing the content of the image using image features and then performing non-textual queries like color and texture is not an easy task for users. They prefer instead textual queries and this request can be satisfied using automatic annotation.

There are many annotation models proposed and each model has tried to improve a previous one. These models were splitted in two categories:

- a) Parametric models: Co-occurrence Model [1], Translation Model [2], Correlation Latent Dirichlet Allocation [4]
- b) Non-parametric models: Cross Media Relevance Model (CMRM) [3], Continuous Cross-Media Relevance Model

(CRM) [10], Multiple Bernoulli Relevance Model (MBRM) [11], Coherent Language Model (CLM) [12]

The annotation process implemented in our system is based on CMRM. Using a set of annotated images [20] the system learns the joint distribution of the blobs and words. The blobs are clusters of image regions obtained using the K-means algorithm. Having the set of blobs each image from the test set is represented using a discrete sequence of blobs identifiers. The distribution is used to generate a set of words for a new image.

Each new image is segmented using an original segmentation algorithm [13] which integrates pixels into a grid-graph. The usage of the hexagonal structure improves the time complexity of the methods used and the quality of the segmentation results. An evaluation of this algorithm against other well know segmentation algorithms like Normalized Cuts segmentation algorithm [14], Efficient Graph-Based segmentation algorithm [15], Mean-Shift segmentation algorithm [16], Color set back-projection algorithm [17] is presented in [17][18]. This algorithm was also used for an image annotation system presented in [24].

The meaningful keywords assigned by the annotation system to each new image are retrieved from an ontology created in an original manner starting from the information provided by [20]. The concepts and the relationships between them in the ontology are inferred from the word's list, from the ontology's paths and from the existing relationships between regions.

The remainder of the paper is organized as follows: related work is discussed in Section 2, Section 3 provides details about the segmentation algorithm used, Section 4 contains a description of the annotation model, Section 5 presents the dataset used for experiments, Section 6 provides a description of the modules included in system's architecture, Section 7 contains the evaluation of the annotation system and Section 8 concludes the paper.

## II. RELATED WORK

Object recognition and image annotation are very challenging tasks. For this reason a number of models using a discrete image vocabulary have been proposed for the image annotation task. One approach to automatically annotating images is to look at the probability of associating words with image regions. Mori et al. [1] used a Co-occurrence Model

in which they looked at the co-occurrence of words with image regions created using a regular grid. To estimate the correct probability this model required large numbers of training samples. Each image is converted into a set of rectangular image regions by a regular grid. The keywords of each training image are propagated to each image region. The major drawback of the above Co-occurrence Model is that it assumes that if some keywords are annotated to an image, they are propagated to each region in this image with equal probabilities.

Duygulu et al [2] described images using a vocabulary of blobs. Image regions were obtained using the Normalized-cuts segmentation algorithm. For each image region 33 features such as color, texture, position and shape information were computed. The regions were clustered using the K-means clustering algorithm into 500 clusters called "blobs". The vector quantized image regions are treated as "visual words" and the relationship between these and the textual keywords can be thought as that between two languages, such as French and German. The training set is analogous to a set of aligned bitexts - texts in two languages. Given a test image, the annotation process is similar to translating the visual words to textual keywords using a lexicon learned from the aligned bitexts. This annotation model called Translation Model was a substantial improvement of the Co-occurrence model.

Jeon et al. [3] viewed the annotation process as analogous to the cross-lingual retrieval problem and used a Cross Media Relevance Model to perform both image annotation and ranked retrieval. The experimental results have shown that the performance of this model on the same dataset was considerably better than the models proposed by Mori et al. [1] and Duygulu et al. [2]. The essential idea is that of finding the training images which are similar to the test image and propagate their annotations to the test image. CMRM does not assume any form of joint probability distribution on the visual features and textual features so that it does not have a training stage to estimate model parameters. For this reason, CMRM is much more efficient in implementation than the above mentioned parametric models.

There are other models like Correlation LDA proposed by Blei and Jordan [4] that extends the Latent Dirichlet Allocation model to words and images. This model is estimated using Expectation-Maximization algorithm and assumes that a Dirichlet distribution can be used to generate a mixture of latent factors.

In [5] it is proposed the use of the Maximum Entropy approach for the task of automatic image annotation. Maximum Entropy is a statistical technique allowing predicting the probability of a label given test data. The image is represented using a language of visterms (visual terms) which are clusters of rectangular regions.

In [6][25] it is described a real-time ALIPR image search engine which uses multi resolution 2D Hidden Markov Models to model concepts determined by a training set. A computational efficiency is obtained in [25] due to a fundamental change in the modeling approach. In [6] every image was characterized by a set of feature vectors residing on grids at several resolutions. The profiling model of each concept is

the probability law governing the generation of feature vectors on 2-D grids. Under the new approach, every image is characterized by a statistical distribution. The profiling model specifies a probability law for distributions directly.

In [7] Latent Semantic Analysis (LSA) [8] and Probabilistic Latent Semantic Analysis (PLSA) [9] are explored for automatic image annotation. A document of image and texts can be represented as a bag of words, which includes the visual words – vector quantized image regions and textual words. Then LSA and PLSA can be deployed to project a document into a latent semantic space. Annotating images is achieved by keywords propagation in this latent semantic space.

An improved model of CMRM is proposed in [10], the Continuous Cross-Media Relevance Model (CRM) which preserves the continuous feature vector of each region and this offers more discriminative power. A further extension of the CRM model called the Multiple Bernoulli Relevance Model (MBRM) is presented in [11]. The keyword distribution of an image annotation is modeled as a multiple Bernoulli distribution, which only represents the existence/nonexistence binary status of each word.

All the above mentioned methods predict each word independently given a test image. They can model the correlation between keywords and visual features but they are not able to model the correlation between two textual words. To solve this problem, in [12] it is proposed a Coherent Language Model (CLM) extended from CMRM. This model defines a language model as a multinomial distribution of words. Instead of estimating the conditional distribution of a single word it is estimated the conditional distribution of the language model. The correlation between words is explained by a constraint on the multinomial distribution that the summation of the individual words distribution is equal to one. The prediction of one word has an effect on the prediction of another word.

### III. THE SEGMENTATION ALGORITHM

For image segmentation we have used an original and efficient segmentation algorithm [6] based on color and some geometric features of an image. The novelty of our algorithm concerns two main aspects:

a) minimizing the running time - a hexagonal structure based on the image pixels is constructed and used in color and syntactic based segmentation

b) using an efficient method for segmentation of color images based on spanning trees and both color and syntactic features of regions. A similar approach is used in [7] where image segmentation is produced by creating a forest of minimum spanning trees of the connected components of the associated weighted graph of the image.

In figure 1 it is presented the hexagonal structure used by the segmentation algorithm:

A particularity of this approach is the basic usage of the hexagonal structure instead of color pixels. In this way the hexagonal structure can be represented as a grid-graph  $G = (V, E)$  where each hexagon  $h$  in the structure has a corresponding vertex  $v \in V$ , as presented in Figure 1. Each

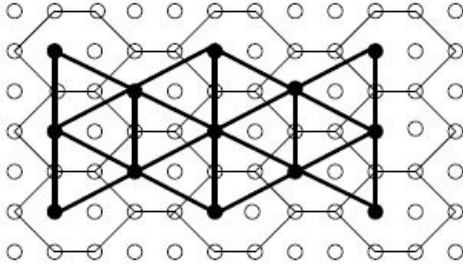


Fig. 1. The grid-graph constructed on the hexagonal structure of an image

hexagon has six neighbors and each neighborhood connection is represented by an edge in the set  $E$  of the graph. To each hexagon two important attributes are associated: the dominant color and the coordinates of the gravity center. For determining these attributes were used eight pixels: the six pixels of the hexagon frontier, and two interior pixels of the hexagon.

Image segmentation is realized in two distinct steps:

c) a pre-segmentation step - only color information is used to determine an initial segmentation. A color based region model is used to obtain a forest of maximum spanning trees based on a modified form of the Kruskal's algorithm. For each region of the input image it is obtained a maximal spanning tree. The evidence for a boundary between two adjacent regions is based on the difference between the internal contrast and the external contrast between the regions

d) a syntactic-based segmentation - color and geometric properties of regions are used. It is used a new graph which has a vertex for each connected component determined by the color-based segmentation algorithm. The region model contains in addition some geometric properties of regions such as the area of the region and the region boundary. A forest of minimum spanning trees is obtained using a modified form of the Boruvka's algorithm. Each minimum spanning tree represents a region determined by the segmentation algorithm.

#### IV. THE ANNOTATION MODEL

The Cross Media Relevance Model is a non-parametric model for image annotation and assigns words to the entire image and not to specific blobs - clusters of image regions, because the blob vocabulary can give rise to many errors. Some principles defined for the relevance models [22, 23] are applied by this model to automatically annotate images and for ranked retrieval. Relevance models were introduced to perform a query expansion in a more formal manner. Given a training set of images with annotations this model allows predicting the probability of generating a word given the blobs in an image. A test image  $I$  is annotated by estimating the joint probability of a keyword  $w$  and a set of blobs:

$$P(w, b_1, \dots, b_m) = \sum_{J \in T} P(J) P(w, b_1, \dots, b_m | J).$$

For the annotation process the following assumptions are made:

a) it is given a collection  $C$  of un-annotated images  
 b) each image  $I$  from  $C$  to can be represented by a discrete set of blobs  $I = \{b_1 \dots b_m\}$

c) there exists a training collection  $T$ , of annotated images, where each image  $J$  from  $T$  has a dual representation in terms of both words and blobs:  $J = \{b_1 \dots b_m; w_1 \dots w_n\}$

d)  $P(J)$  is kept uniform over all images in  $T$

e) the number of blobs  $m$  and words in each image ( $m$  and  $n$ ) may be different from image to image.

f) no underlying one to one correspondence is assumed between the set of blobs and the set of words; it is assumed that the set of blobs is related to the set of words.

$P(w, b_1, \dots, b_m | J)$  represents the joint probability of keyword  $w$  and the set of blobs  $(b_1, \dots, b_m)$  conditioned on training image  $J$ . An intuitive interpretation of this probability is how likely  $w$  co-occurs with individual blobs given that we have observed an annotated image  $J$ .

In CMRM it is assumed that, given image  $J$ , the events of observing a particular keyword  $w$  and any of the blobs

$(b_1, \dots, b_m)$  are mutually independent, so that the joint probability can be factorized into individual conditional probabilities. This means that  $P(b_1, \dots, b_m | J)$  can be written as:

$$P(w, b_1, \dots, b_m | J) = P(w | J) \prod_{i=1}^m P(b_i | J)$$

$$P(w | J) = (1 - \alpha_j) \frac{\#(w, J)}{|J|} + \alpha_j \frac{\#(w, T)}{|T|}$$

$$P(b | J) = (1 - \beta_j) \frac{\#(b, J)}{|J|} + \beta_j \frac{\#(b, T)}{|T|}$$

where:

- $P(w | J)$ ,  $P(b | J)$  denote the probabilities of selecting the word  $w$ , the blob  $b$  from the model of the image  $J$ .
- $\#(w, J)$  denotes the actual number of times the word  $w$  occurs in the caption of image  $J$ .
- $\#(w, T)$  is the total number of times  $w$  occurs in all captions in the training set  $T$ .
- $\#(b, J)$  reflects the actual number of times some region of the image  $J$  is labeled with blob  $b$ .
- $\#(b, T)$  is the cumulative number of occurrences of blob  $b$  in the training set.
- $|J|$  stands for the count of all words and blobs occurring in image  $J$ .
- $|T|$  denotes the total size of the training set.
- The prior probabilities  $P(J)$  can be kept uniform over all images in  $T$

The smoothing parameters  $\alpha$  and  $\beta$  determine the degree of interpolation between the maximum likelihood estimates and the background probabilities for the words and the blobs



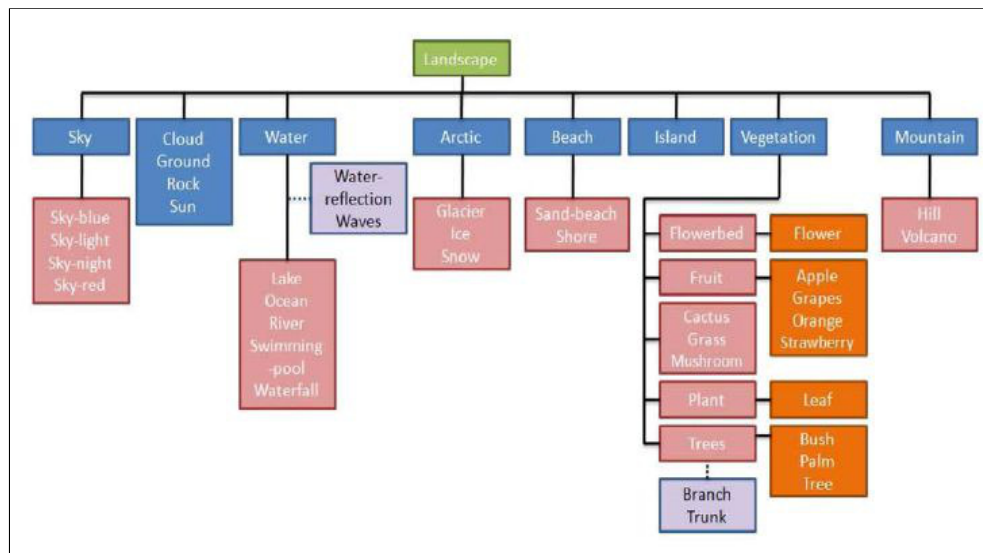


Fig. 2. The hierarchical structure of the Landscape-Nature branch.

respectively. The values determined after experiments for the Cross Media Relevance Model were  $\alpha = 0.1$  and  $\beta = 0.9$ .

### V. DATASET

We have used for our experiments the segmented and annotated SAIAPR TC-12 [20][27] benchmark which is an extension of the IAPR TC-12 [21] collection for the evaluation of automatic image annotation methods and for studying their impact on multimedia information retrieval. IAPR TC-12 was used to evaluate content based image retrieval and multimedia image retrieval methods [28][29]. SAIAPR TC-12 benchmark contains the pictures from the IAPR TC-12 collection plus: segmentation masks and segmented images for the 20,000 pictures, region-level annotations according an annotation hierarchy, region-level annotations according an annotation hierarchy, spatial relationships information. Each image was manually segmented using a Matlab tool named Interactive Segmentation and Annotation Tool (ISATOOL). ISATOOL allows the interactive segmentation of objects by drawing points around the desired object, while splines are used to join the marked points, which also produces fairly accurate segmentation with much lower segmentation effort. Each region has associated a segmentation mask and a label from a predefined vocabulary of 275 labels. This vocabulary is organized according to a hierarchy of concepts having six main branches: Humans, Animals, Food, Landscape-Nature, Man-made and Other. In figure 2 it is presented the hierarchical structure of the Landscape-Nature branch.

For each pair of regions the following relationships have been calculated in every image: adjacent, disjoint, beside, X-aligned, above, below and Y-aligned. The following features have been extracted from each region: area, boundary/area, width and height of the region, average and standard deviation in x and y, convexity, average, standard deviation and skewness in two color spaces: RGB and CIE-Lab.

The dataset contains several folders of images, each folder having the structure presented in figure 3:

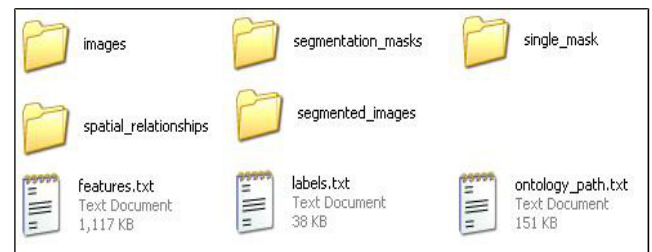


Fig. 3. The structure of images' folder

where:

a) images folder contains the initial images that were manually segmented

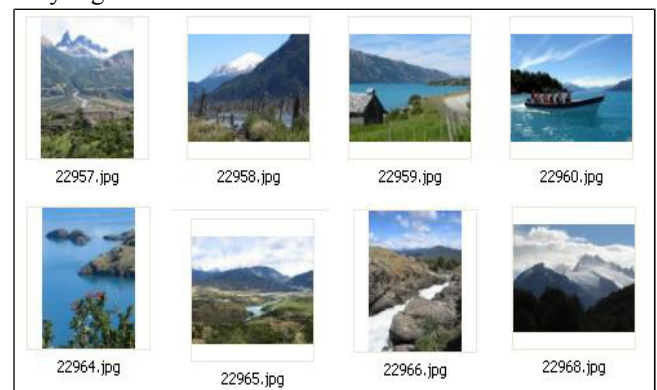


Fig. 4. Initial images

b) segmentation\_masks folder contains files having the extension .mat (Matlab files). For each image's region a file is provided containing a segmentation mask which can be seen as a matrix with 0 and 1 values. A value of 1 in a matrix location means that the pixel having that position in the original image belongs to that region.

c) single\_mask folder contains a single .mat file per image, representing the mask of the entire image.

d) spatial\_relationships contains a file per image with information about the spatial relationships detected between each pair of regions

e) segmented\_images folder contains manually segmented images having the regions shown with boundary

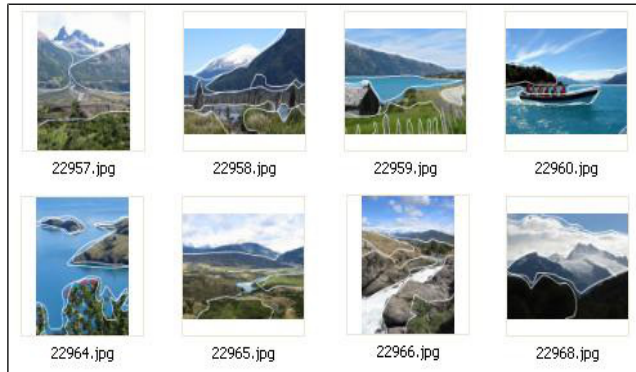


Fig. 5. Manually segmented images

22006	1	0.2724016	1	0.4166667	0.5620036
0.2899823		1.024098	0.3124049	0.02863759	0.3462361
171.2322		152.0555	131.734	12.08801	12.00721
13.60371		-0.5269354	-0.1912287	-0.07343204	82.12726
1.691708		7.70454	2.5315	0.9239739	2.134627
-0.418851		0.1365035	-0.1540672	131	
22006	2	0.05034722	0.2291667	0.2666667	0.7199739
0.6114282		0.1999893	0.2625338	0.04321839	0.1761364
138.2292		127.312	55.6731	13.85575	10.91165
13.46869		-0.2995422	-0.7418427	1.213454	75.24667
-6.5022		31.02281	2.812094	1.992792	5.63132
-0.8145331		0.6401796	-0.8357982	223	

Fig. 6. Features' values for each region

f) features.txt contains the values of the extracted features from each region

g) 2206 identifies the picture (Figure 6 - 22006.jpg), values 1 and 2 represent the index of each region and the rest of the values represent the values of the extracted features

h) labels.txt file contains the information needed to identify the words assigned to each image region, each word being indicated by his index. Using this information and the list of all words available in the wlist.txt file (being available for all folders at a higher level) having a pair (word index, word) on each line we can determine the words assigned to regions

22957	1	223	sky
22957	2	258	vegetation
22957	3	168	mountain
22957	4	168	mountain
22957	5	168	mountain
22958	1	168	mountain
22958	2	131	hill
22958	3	224	sky-blue
22958	4	34	bush
22958	5	34	bush
22958	6	29	branch

Fig. 7. Words assigned to regions

i) where 22957 and 22958 represent images' identifiers, 1, 2...5 or 1,2 ...6 represent the index of each region.

j) ontology\_path.txt file contains the path in the ontology for each word associated to a region

22957	1	entity->>landscape-nature->_sky
22957	2	entity->>landscape-nature->vegetation
22957	3	entity->>landscape-nature->mountain
22957	4	entity->>landscape-nature->mountain
22957	5	entity->>landscape-nature->mountain
22958	1	entity->>landscape-nature->mountain
22958	2	entity->>landscape-nature->mountain->hill
22958	3	entity->>landscape-nature->_sky->sky-blue
22958	4	entity->>landscape-nature->vegetation->trees->bush
22958	5	entity->>landscape-nature->vegetation->trees->bush
22958	6	entity->>landscape-nature->vegetation->trees->_branch

Fig. 8. Ontology's paths assigned to regions.

## VI. SYSTEM'S ARCHITECTURE

System's architecture is presented in figure 9 and contains 6 modules:

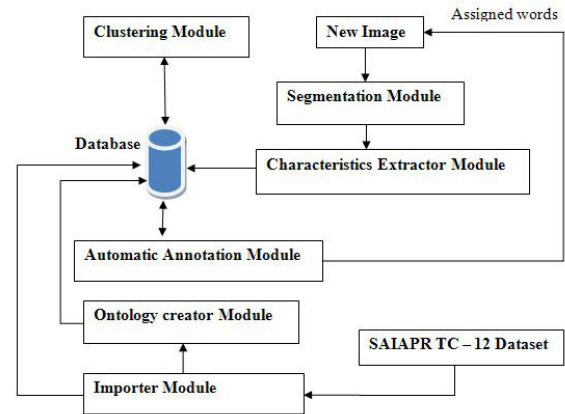


Fig.9. System's architecture

a) Importer module – this module is used to extract the existing information in the dataset. Having available segmentation's mask for each image's region this module detects the pixels that belong to that region. By parsing the content of the features.txt file the module extracts a list of feature vectors that are stored in the database. These feature vectors are clustered by the Clustering module for obtaining a list of blobs. The existing information in the labels.txt and ontology\_path.txt files about the words assigned to regions and the paths in the ontology is extracted and is made available to the Ontology creator module.

b) Ontology creator module - using the information provided by the Importer module and an original approach this module creates an ontology that is used for annotating new images. The existing ontology's paths are used to establish the hierarchical structure of the ontology. Each path is converted to several hierarchical relationships of parent-child type. The information contained in the spatial\_relationships folder is used to generate several relationships in the ontology having spatial-relationship type. Each word is represented as a concept in the ontology having as unique identifier his index in the wlist.txt file. The ontology is represented as a Topic Map [30] using the XTM syntax [31]. In the bellow table are presented two ontology concepts (Mountain and Landscape) modeled as topics in the Topic Map and a hierarchical relationship between them modeled as an association:



Topics	<pre> &lt;topic id= "168"&gt;   &lt;instanceOf&gt;     &lt;topicRef       xlink:href="#semantic-class"/&gt;     &lt;/instanceOf&gt;     &lt;baseName&gt;    &lt;base-       NameString&gt;Mountain&lt;/base-       NameString&gt;     &lt;/baseName&gt;   &lt;/topic&gt; </pre>
	<pre> &lt;topic id= "148"&gt;   &lt;instanceOf&gt;     &lt;topicRef       xlink:href="#semantic-class"/&gt;     &lt;/instanceOf&gt;     &lt;baseName&gt;    &lt;base-       NameString&gt;Landscape&lt;/base-       NameString&gt;     &lt;/baseName&gt;   &lt;/topic&gt; </pre>
Association	<pre> &lt;association id="148-168"&gt;   &lt;instanceOf&gt;     &lt;topicRef       xlink:href="#parent-child"/&gt;     &lt;/instanceOf&gt;     &lt;member&gt;       &lt;roleSpec&gt;         &lt;topicRef           xlink:href="#parent"/&gt;         &lt;/roleSpec&gt;         &lt;topicRef           xlink:href="#148"/&gt;       &lt;/member&gt;       &lt;member&gt;         &lt;roleSpec&gt;           &lt;topicRef             xlink:href="#child"/&gt;           &lt;/roleSpec&gt;           &lt;topicRef             xlink:href="#168"/&gt;         &lt;/member&gt;       &lt;/association&gt; </pre>

Segmentation module – this module is using the segmentation algorithm described in Section 3 to obtain a list of regions from each new image. The segmentation algorithm is using some methods during the segmentation process:

SameVertexColor – used to determine the color of a hexagon

ExpandColorArea – used to determine the list of hexagons having the color of the hexagon used as a starting point and has  $O(n)$  as running time where  $n$  is the number of hexagons from a region with the same color.

ListRegions – used to obtain the list of regions and has  $O(n^2)$  as running time where  $n$  is the number of hexagons from the hexagonal network.

ContourRegions – used to obtain the contour of each region and has  $O(n)$  as running time where  $n$  is the number of hexagons from a region with the same color

Characteristics extractor module - this module is using the regions detected by the Segmentation module. For each segmented region it is computed a feature vector that contains visual information of the region such as area, boundary/area, width and height of the region, average and

standard deviation in  $x$  and  $y$ , convexity, average, standard deviation. All feature vectors obtain are stored in the database in order to be accessible for other modules.

Clustering module - we have used K-means algorithm to quantize the feature vectors obtained from the training set and to generate blobs. After the quantization, each image in the training set was represented as a set of blobs identifiers. For each blob it is computed a median feature vector and a list of words that were assigned to the test images that have that blob in their representation.

Automatic annotation module - for each region belonging to a new image it is assigned the blob which is closest to it in the cluster space. The assigned blob has the minimum value of the Euclidian distance computed between the median feature vector of that blob and the feature vector of the region. In this way the new image will be represented by a set of blobs identifiers. Having the set of blobs and for each blob having a list of words we can determine a list of potential words that can be assigned to the image. What needs to be established is which words better describe the image content. This can be made using formulas (3) and (4) of the Cross Media Relevance Model. For each word it is computed the probability to be assigned to the image and after that the set of words having a probability greater than a threshold value will be used to annotate the image.

## VII. EVALUATION OF THE ANNOTATION SYSTEM





In order to evaluate the annotation system we have used a testing set of 400 images that were manually annotated and not included in the training set used for the CMRM model. This set was segmented using the original segmentation algorithm described above and a list of words having the joint probability greater than a threshold value was assigned to each image. Then the number of relevant words automatically assigned by the annotation system was compared against the number of relevant words manually assigned by computing a recall value. Using this approach for each image we have obtained a statistic evaluation having the structure presented in Table 1.

After computing the recall value for each image it was obtained a medium recall value equal to 0.73.

## VIII. CONCLUSIONS AND FUTURE WORK

In this paper we described a system that can be used for annotating natural images. The CMRM annotation model implemented by the system was proven to be very efficient by several studies. This model learns the joint probability of words and blobs based on a well know benchmark: SAIAPR TC-12. This benchmark contains a large-size image collection comprising diverse and realistic images, includes an annotation vocabulary having a hierarchical organization, well defined criteria for the objective segmentation and annotation of images. Because the quality of an image region and the running time of the segmentation process are two important factors for the annotation process we have used a segmentation algorithm based on a hexagonal structure which was proved to satisfy both requirements: a better quality and a smaller running time. Each new image was annotated with

TABLE 1. STATISTIC EVALUATION OF THE SYSTEM

Index	Image	Relevant words automatically assigned (RWAA)	Words manually assigned (WMA)	Recall = RWAA/WMA
0		sky-blue, sand-beach, ocean	sand-beach, ocean, boat, palm, hut, sky-blue	3/6 = 0.50
1		sky-blue, grass, ocean, cloud	grass, ocean, boat, cloud, sky-blue, branch	4/6 = 0.66
2		sky, mountain, lake	lake, vegetation, mountain, cloud, sky	3/5 = 0.60
3		mountain, sky-blue, sand-dessert	mountain, lake, sand-dessert, sky-blue	3/4 = 0.75

words taken from an ontology created starting from the information provided by the benchmark: the hierarchical organization of the vocabulary and the spatial relationships between regions. The ontology created in an original manner was represented using the Topic Map standard, each concept being modeled as a topic item and each relationship as an association having a specific type.

Further extensions of the system will include the two models of image retrieval provided by CMRM: Annotation-based Retrieval Model and Direct Retrieval Model.

ACKNOWLEDGMENT

The support of The National University Research Council under Grant CNCSIS IDEI 535 is gratefully acknowledged.

REFERENCES

[1] Y. Mori, H. Takahashi, R.Oka: Image-to-word transformation based on dividing and vector quantizing images with words. In: MISRM'99 First Intl. Workshop on Multimedia Intelligent Storage and Retrieval Management (1999)

[2] P. Duygulu, K. Barnard, N. de Freitas, D. Forsyth: Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In Seventh European Conf. on Computer Vision, pp. 97–112 (2002)

[3] J. Jeon, V. Lavrenko, R. Manmatha: Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. In: Proceedings of the 26th Intl. ACM SIGIR Conf., pp. 119–126 (2003)

[4] D. Blei, Michael, and M. I. Jordan. Modeling annotated data. To appear in the Proceedings of the 26th annual international ACM SIGIR conference

[5] J. Jeon and R. Manmatha, "Using maximum entropy for automatic image annotation." in CIVR, pp. 24–32, 2004

[6] J. Li, J. Wang.: Automatic linguistic indexing of pictures by a statistical modeling approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2003)

[7] F. Monay and D. Gatica-Perez. PIsa-based image auto-annotation: constraining the latent space. In Proceedings of ACM International Conference on Multimedia (ACM MULTIMEDIA), pages 348–351, 2004.

[8] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. Journal of the Society for Information Science, 41(6):391–407, 1990.

[9] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. Machine Learning, 42(1-2):177–196, 2001.

[10] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In Proceedings of Advances in Neural Information Processing Systems (NIPS), 2004.

[11] S. L. Feng et al. Multiple bernoulli relevance models for image and video annotation. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1242–1245, 2004.

[12] J. Rong, J. Y. Chai, and L. Si. Effective automatic image annotation via a coherent language model and active learning. In Proceedings of ACM International Conference on Multimedia (ACM MULTIMEDIA), pages 892–899, 2004.

[13] D. Burdescu, M. Brezovan, E. Ganea, and L. Stanescu, "A New Method for Segmentation of Images Represented in a HSV Color Space", Lecture Notes in Computer Science, 5807, 606-617, 2009.

[14] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation", IEEE Transactions on pattern analysis and machine intelligence, Vol. 22, No. 8, 2000.

[15] P. Felzenszwalb and D. Huttenlocher, "Efficient Graph-Based Image Segmentation", Intl J. Computer Vision, vol. 59, no. 2, 2004.

[16] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach toward Feature Space Analysis", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, pp. 603-619, 2002.

[17] J. R. Smith, S. F. Chang.: "Tools and Techniques for Color Image Retrieval", Symposium on Electronic Imaging. In: Science and Technology - Storage & Retrieval for Image and Video Databases IV, volume 2670, San Jose, CA, February 1996. IS&T/SPIE. (1996)

[18] G. Mihai, A. Doringa, L. Stanescu, A Graphical Interface for Evaluating Three Graph-Based Image Segmentation, 2010, Proceedings of the International Multiconference on Computer Science and Information Technology pp. 735–740, 2010

[19] A. Iancu, B. Popescu, M. Brezovan, E. Ganea, "Region-based Measures for Evaluation of Color Image Segmentation", Proceedings of the International Multiconference on Computer Science and Information Technology pp. 717–722, 2010

[20] "Segmented and Annotated IAPR TC-12 dataset", <http://imageclef.org/SIAPRdata>

[21] "IAPR TC-12 Benchmark", <http://imageclef.org/photodata>

- [22] V. Lavrenko and W. Croft. "Relevance-based language models". Proceedings of the 24th annual international ACM SIGIR conference, pages 120-127, 2001.
- [23] V. Lavrenko, M. Choquette, and W. Croft. "Cross-lingual relevance models". Proceedings of the 25th annual international ACM SIGIR conference, pages 175-182, 2002.
- [24] E. Ganea, M. Brezovan, "An Hypergraph Object Oriented Model for Image Segmentation and Annotation", Proceedings of the International Multiconference on Computer Science and Information Technology pp. 695-701, 2010
- [25] J. Li, J.Z.Wang, "Real-time computerized annotation of pictures", IEEE transactions on pattern analysis and machine intelligence, Vol. 30, No. 6. (June 2008), pp. 985-1002
- [26] R. Datta, D. Joshi, J. Li, J. Z. Wang, Image retrieval: ideas, influences, and trends of the new age, ACM Computing Surveys 40 (2) (2008) 1-60.
- [27] H. J. Escalante, C. A. Hernández, J. A. Gonzalez, A. López-López, M. Montes, E. F. Morales, L. Enrique Sucar, L. Villaseñor and M. Grubinger, "The segmented and annotated IAPR TC-12 benchmark", Computer Vision and Image Understanding, Volume 114, Issue 4, April 2010, Pages 419-428
- [28] P. Clough, M. Grubinger, T. Deselaers, A. Hanbury, H. Müller, "Overview of the ImageCLEF 2006 photographic retrieval and object annotation tasks", Evaluation of Multilingual and Multimodal Information Retrieval – 7th Workshop of the CLEF, LNCS vol. 4730, Springer, Alicante, Spain, 2006, pp. 579-594.
- [29] M. Grubinger, P. Clough, A. Hanbury, H. Müller, "Overview of the ImageCLEF 2007 photographic retrieval task", Advances in Multilingual and Multimodal Information Retrieval – 8th Workshop of CLEF, LNCS vol. 5152, Springer, Budapest, Hungary, 2007, pp. 433-444.
- [30] Topic Maps, <http://www.topicmaps.org/>
- [31] XTM syntax, <http://www.topicmaps.org/xtm/>

# Fuzzy UML and Petri Nets Modeling Investigations on the Pollution Impact on the Air Quality in the Vicinity of the Black Sea Constanta Romanian Resort

Elena-Roxana Tudoroiu  
Technical University of  
Cluj-Napoca, 15 Constantin  
Daicoviciu Street, Cluj-Napoca,  
Romania  
tudelena@excite.com

Adina Astilean  
Technical University of  
Cluj-Napoca, 15 Constantin  
Daicoviciu Street, Cluj-Napoca,  
Romania  
adina.astilean@aut.utcluj\_ro

Tiberiu Letia  
Technical University of Cluj-  
Napoca, 15 Constantin  
Daicoviciu Street, Cluj-Napoca,  
Romania  
tsletia@gmail.com

Gabriela Neacsu  
"Spiru Haret" University of Constan-  
ta, 42-44 Unirii street, Constanta, Ro-  
mania  
gabrielle\_neacsu@yahoo.com

Maroszy Zoltan  
Ecological University Bucharest,  
Romania  
maroszy.zoltan@gmail.com

Nicolae Tudoroiu  
Concordia University  
1455 De Maisonneuve Blvd.  
West, Montreal, Quebec, Cana-  
da H3G 1M6  
tnicolae@excite.com

**Abstract**—The purpose of this research is to investigate the use of an intelligent neural-fuzzy modeling strategy based on Unified Modeling Language (UML) diagrams and Petri nets models of the pollution sources impact on the air quality along the Romanian coast of Black Sea, especially in Constanta vicinity. This is possible by monitoring the physical and chemical parameters of the air quality, such as temperature, wind speed, Carbon Dioxide (CO<sub>2</sub>), methane (CH<sub>4</sub>), Nitrogen Oxide, ozone, water vapours concentrations provided by several "in-situ" measurements stations spread in the critical points from Constanta area. Moreover, we will try to disseminate the information collected and to investigate adequate actions to prevent the continuous degradation of the environment. The values of air quality-monitored parameters vary with the position of the sampling sites in quasi-large range; consequently a direct correlation between these indicators will be useful. Air pollution sources cause the "greenhouse effect" with a high impact on the live and fauna, degrading progressively the Black Sea ecosystems. Closing, in our research we try to present the benefit of the UML diagrams in combination with Petri nets models developed on a wide database concerning the air pollution degree inside Constanta Romanian Black sea resort to predict the future results.

## I. INTRODUCTION

THE direct impact of pollutants discharged into the atmosphere by agents economic generally occur in areas relatively close by them, range from a few tens or hundreds of meters up to several kilometers according to physical parameters, the power output of the source and especially climatic conditions, intensity and wind. Considerable influence on the spread of pollutants in the air has the climate conditions for example: thermal stratification of air, turbulence, convection, precipitation and wind. Thermal stratification of the air may be stable or unstable, and affects the vertical dispersion of pollutants. Thus, a stable stratification prevents the diffusion of pollutants in height, and resulting in their concentration in the soil near the source, while the unstable stratification favors the diffusion of pollutants. Turbulence also occurs in the dispersion of

pollutants, representing the disordered movements of the air in the form of small vortices. Dynamic and thermal convection is the most important process that produces heat exchange between terrestrial surface and the air above, and between different layers of the atmosphere. In addition, in the summer months heat exchanges in turbulent times, the thermal convection during the nights having a irregular character, predominantly downward movements. This is because solid particles which floating in the air are cooled by radiation, increasing their density and become more heavy being forced to get down on the ground. In winter time, during the night radiation's balance becomes negative, prevailing only downward movements and the turbulent exchange is very small and oriented to the terrestrial surface. In this case, the sediments of the irrespirable particles, along with other pollutants are deposited on the ground at night when the population is less exposed. In the same coastal Black sea area the cloud is also an important factor in pollutant dispersion, the frequency of cloudy days being greater than the sunny days, usually situated between 86 and 93 days on the coast and between 90 and 93 on the dry zone. The wind direction and intensity influence the horizontal spread of the pollutants. The parameters of wind analyzed are the average speed, wind gust, and wind direction recorded for a determined period in years.

## II. THE POLLUTION IMPACT ON THE AIR QUALITY IN CONSTANTA'S METROPOLITAN AREA

The atmosphere air is a mixture of uncontaminated air, water vapours and different impurities. The natural impurities represent a small amount and could be: meteoric debris, natural powder particles, NaCl, NO<sub>2</sub>, SO<sub>2</sub>, HCl, H<sub>2</sub>S, ozone, bacteria, pollen, powder, and condensation nuclei, and impurities created by the industrial activities. In this research we monitor the level of "greenhouse gas effect" emission, during 2006-2007, concerning the following air pollution agents: CO, NH<sub>3</sub>, N<sub>2</sub>O, ozone, H<sub>6</sub>C<sub>6</sub>, and the debris. The

concentrations of all these air pollution agents were presented in the conference paper [7]. The air pollution agents with high impact on the air quality considered in our research, shown in Table 1, are also well documented and modeled in [2].

TABLE 1:  
THE AIR QUALITY CLUSTER CLASSIFICATION [2]

Air quality	SO <sub>2</sub>	NO <sub>2</sub>	CO	O <sub>3</sub>	PM <sub>10</sub>
	1h [ $\mu\text{g}\cdot\text{m}^{-3}$ ]	1h [ $\mu\text{g}\cdot\text{m}^{-3}$ ]	8h [ $\mu\text{g}\cdot\text{m}^{-3}$ ]	1h [ $\mu\text{g}\cdot\text{m}^{-3}$ ]	1h [ $\mu\text{g}\cdot\text{m}^{-3}$ ]
Very good	0-25	0-25	0-1.10 <sup>3</sup>	0-33	0-15
Good	25-50	25-50	1000-2000	33-65	15-30
Favourable	50-120	50-100	2000-4000	65-120	30-50
Satisfactory	120-250	100-200	4000-10000	120-180	50-70
Bad	250-500	200-400	10000-30000	180-240	70-150
Very bad	500-	400-	30000-	240-	150-

In this table the cluster classification of the air quality is based on the limits set for health protection, fauna and flora protection separately, and the dispersion conditions depend on the horizontal and vertical airflow, rain falls and wind direction, especially [2].

### III. FUZZY-UML TO FUZZY-PETRI NET CONVERSION—MODELING STRATEGY

This modeling strategy could be applied successfully to model the impact of pollutant agents on air quality following the same development from [4] based on the idea presented in Figure 1.

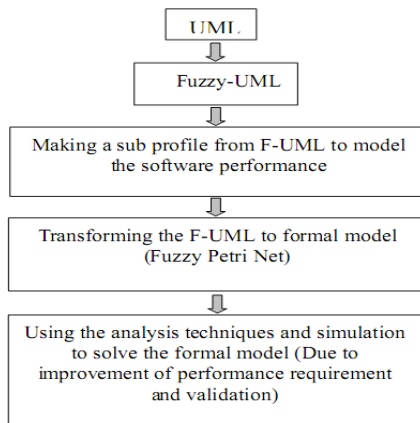


Fig. 1. Fuzzy UML and Fuzzy Petri Net models conversion [4]

The UML diagram supports behavioral and structural aspects of the modeling systems, and consequently the fuzzy concept is concerning the both, the fuzzy structure (fuzzy data model) and the fuzzy behavior (the models are needed to support system functionality fuzzily). Concerning the data modeling the uncertainty in data structure is entered by presenting an UML class diagram fuzzily. The fuzzy behavior modeling is resolved by

presenting the use case, sequence and state diagrams fuzzily [4], [11]-[12]. Several researches have been performed to deal with the semi-formal problem of UML. Some of these researches have only used a transformation algorithm, that transforms the created model into a Petri net as a mathematical and formal model that, in turn, contains the visual of modeling and pursues the verification operations with further ability [4], [8]. According to the season (scenario) we consider the sensor of parameter condition (UML diagram attributes) during the time intervals. Based on these values of the attributes the new fuzzy values of the condition and event parts will be generated, therefore the fuzzy UML diagram is also generated, as a support for the implementation of fuzzy Petri net model. The fuzzy UML state diagram created will be converted to a fuzzy Petri net model according to the steps presented in [8]: The Artificial Intelligence (AI) problems are typically solved via state-space approach to design algorithms intended for reaching one or more target states from the selected database initial states. The transition between the states is carried out by applying an appropriate set of fuzzy set rules selected according an expert knowledge from the given database, especially fuzzy IF-THEN production rules and database [3], [5], such in the following simple problem, implemented as a Petri-like net model in Figure 2:

Production Rules:

PR1: IF (P2) AND (P1) THEN P7,

PR2: IF P7 THEN P6,

PR3: IF (P7) AND (P3) THEN P4,

PR4: IF (P4) THEN P5

Database: P1, P2, P3

The most common operators of fuzzy applied to fuzzy sets are AND (minimum), OR (maximum) that have binary arguments, and Negation (complementation) having unary argument. A fuzzy set, unlike conventional sets, include all the elements of the universal set of the domain but with varying membership values in the interval  $[0, 1]$ , and fuzzy logic introduced for the first time and well developed by Prof. Zadeh [1]. Sometimes it is possible to handle two or more output variables (places) that may occur as in the following linguistic description rule:

R4:=IF (P<sub>in1</sub>) is E<sub>1</sub> THEN P<sub>out1</sub> is E<sub>out1</sub> AND P<sub>out2</sub> is E<sub>out2</sub> AND P<sub>outN</sub> is P<sub>outN</sub> This case can be modeled by two separate Fuzzy nets shown in figure 5.

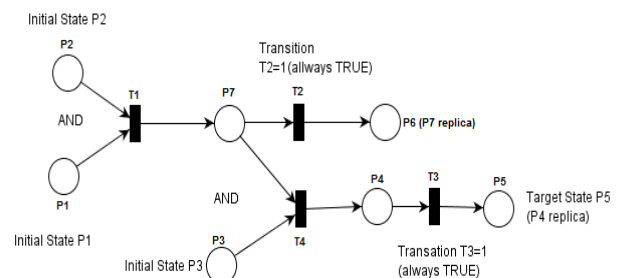


Fig.2 The generic overall Petri net model

In this model the token will be bearer of fuzzy sets, the edges (arcs) ( $E_i$ ) will be evaluated by linguistic expression

from IF THEN rules and the transitions ( $T_i$ ) represent basically fuzzy relations according to the IF THEN rule.

Any IF-THEN rule defined by previous PR1-PR4 represents the simplified representation of two basic generic rules that could be modeled by the following Petri nets:

R1:= IF ( $P_{in1}$ ) is  $E_1$  AND ( $P_{in2}$ ) is  $E_2$  AND ..AND ( $P_{inN}$ ) is  $E_N$  THEN  $P_{out}$  is  $E_{out}$ , with the Petri net model shown in Figure 3.

R2:= IF ( $P_{in1}$ ) is  $E_1$  THEN  $P_{out}$  is  $E_{out1}$  (missing edges), alone or together with similar rules, i.e.

R3:= IF ( $P_{in2}$ ) is  $E_2$  THEN  $P_{out}$  is  $E_{out2}$ , with the Petri net model shown in Figure 4.

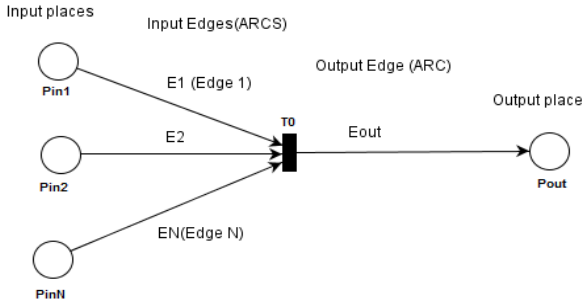


Fig.3 Petri net model implementation based on rule R1.

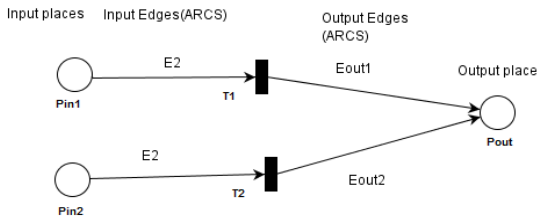


Fig.4: Petri net model implementation based on rules R2-3.

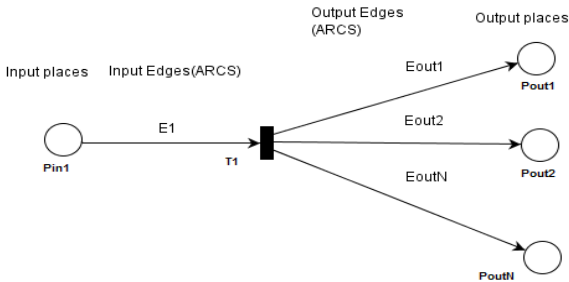


Fig.5 Petri net model implementation based on rule R4.

If the following linguistic description is necessary:

R5:=IF ( $P_{in1}$ ) is  $E_{in1}$  THEN  $P_{out1}$  is  $E_{out1}$

R6:=IF ( $P_{out1}$ ) is  $E_{in2}$  THEN  $P_{out2}$  is  $E_{out2}$ ,

then it can be modeled as shown in figure 6.

Furthermore, we will formalize three basic forms of inconsistency due to incompleteness of Knowledge Bases [3]:

1. Dangling condition (Figure 7)

R7:= IF ( $P_1$ ) is  $E_{12}$  AND ( $P_2$ ) is  $E_{22}$  AND ( $P_3$ ) is  $E_{32}$  THEN  $P_{out}$  is  $E_{out}$

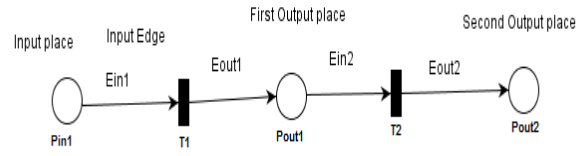


Fig.6. Petri net model implementation based on rules R5-6.

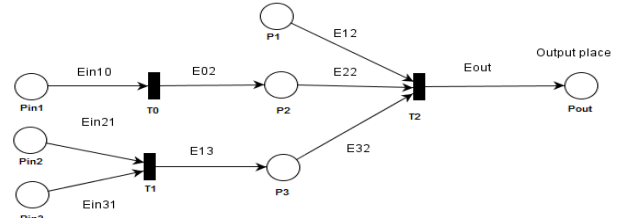


Fig.7: Dangling rule implemented by a Petri net model

From this description the predicate  $P_1$  of the antecedent part of rule R7 is absent from the available database and the consequent part of any rule. Consequently,  $P_1$  can never be generated by firing of any rules.

2. Useless conclusion

It occurs when the predicates in the consequent part of a rule are absent from the antecedent part of the rules in the knowledge base, such as in figure 8. The consequent predicate is called a useless conclusion as no other rules use it as an antecedent predicate.

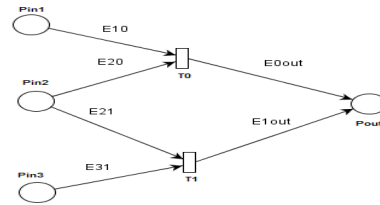


Fig. 8 Useless conclusion rule implemented by a Petri net model

R8.1:= IF ( $P_{in1}$ ) is  $E_{10}$  AND  $P_{in2}$  is  $E_{20}$  THEN  $P_{out}$  is  $E_{0out}$

R8.2:= IF ( $P_{in2}$ ) is  $E_{21}$  AND  $P_{in3}$  is  $E_{31}$  THEN  $P_{out}$  is  $E_{1out}$

A simplified fuzzy Petri Net model for our problem is implemented in figures 9-10 for the weather and for air quality cluster classification, similarly as in [5].

IV. CONCLUSION

This research work is dedicated to investigate the possibility of applying several Fuzzy UML and Petri Nets architectures for simulation and prediction of the performance of air quality of the Constanta Black Sea resort city environment. Therefore we explore different techniques to build several fuzzy Petri net models starting from fuzzy UML diagrams using the experimental data set provided by “in-situ” samples measurements sites.



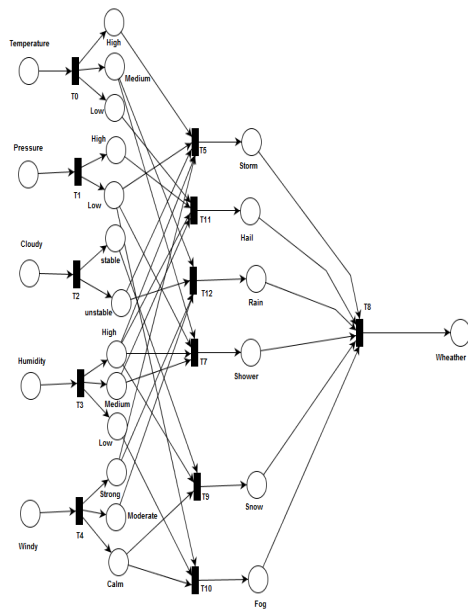


Fig. 9 The simplified weather forecast fuzzy Petri net model

#### REFERENCES

- [1] L. A. Zadeh, "The role of fuzzy logic in the management of uncertainty in Expert System", *Fuzzy Sets Systems*, Elsevier, North Holland, Vol. 11, pp. 199-227, 1983.
- [2] Hájek, V. Olej, "Air Quality Modelling by Kohonen's Self-organizing Feature Maps and LVQ Neural Networks", *WSEAS Transactions on Environment and Development*, ISSN: 1790-5079, Issue 1, Volume 4, January 2008, pp.45-55.
- [3] Amit Konar, "Artificial Intelligence and Soft Computing-Behavior and Cognitive Modeling of the Human Brain", *CRC Press LLC*, 2000 N.W Corporate Blvd., Boca Raton, Florida, ISBN 0-8493-1385-6.
- [4] A. Haroonabadi, M. Teshnehlab, "A Novel Method for Behavior Modeling in Uncertain Information Systems", *World Academy of Science, Engineering and Technology*, Vol.41, 2008, pp. 959-966.

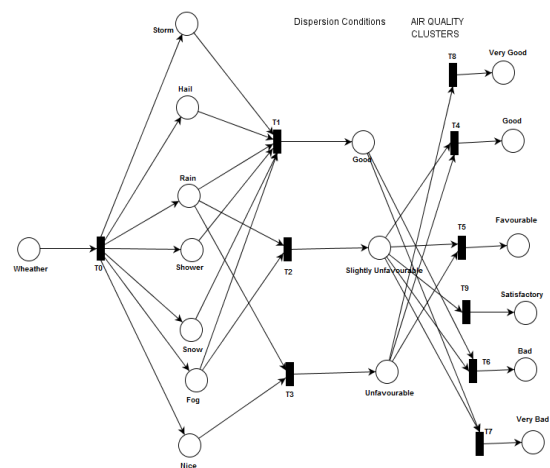


Fig. 10 The simplified air quality cluster classification fuzzy Petri net model

- [5] Jaroslav Knybel, Viktor Pavliska, "Representation of Fuzzy IF-THEN rules by Petri Nets", *research Report No.84, 2005, project 1M0572 of the MSMT Czech Republik*, submitted to ASSIS 2005.
- [6] Tudoroiu Elena-Roxana, V. Cretu, J. Paquet (2009), "Investigations using the rational unified process (RUP) diagrams for software process modeling", *Computer Science and Information Technology, IMCSIT Proceedings*, Mragovo, Poland, 2009, available also in Digital Library IEEEExplore 2009.
- [7] N. Tudoroiu, G. Neacsu, N. Ilias, V. Cretu, D. Curiac, "The Neural Simulator of the Pollution Factors Impact on the Quality of the Air along the Romanian Coast of Black Sea", *IMCSIT'09, International Multiconference on Computer Science and Information Technologies*, Mragowo, Poland, 2009, Conference Proceedings CD, ISSN 1896-7094, ISBN 978-83-60810-22-4, pp.453-460.
- [8] H. Motameni, Daneshfar I., Bakhishi J., Nematzadeh H., "Transforming Fuzzy State Diagram to Fuzzy Petri net", *Journal of Computer Engineering*, Vol.1, 2009, pp.29-44, [http://jacr.iausari.ac.ir/NO%201/5\\_Dec4.pdf](http://jacr.iausari.ac.ir/NO%201/5_Dec4.pdf)



# Pass-Image Authentication Method Tolerant to Video-Recording Attacks

Yutaka Hirakawa

Shibaura Institute of Technology 3-7-5, Toyosu, Koto-ku, Tokyo, 135-8548 Japan

Email: hirakawa@sic.shibaura-it.ac.jp

Motohiro Take

Shibaura Institute of Technology 3-7-5, Toyosu, Koto-ku, Tokyo, 135-8548 Japan

Email: m108075@sic.shibaura-it.ac.jp

Kazuo Ohzeki

Shibaura Institute of Technology 3-7-5, Toyosu, Koto-ku, Tokyo, 135-8548 Japan

Email: ohzeki@sic.shibaura-it.ac.jp

**Abstract**—User authentication is widely used in automatic teller machines (ATMs) and Internet services. Recently, ATM passwords have been increasingly stolen using small charge-coupled device cameras.

This article discusses a user authentication method in which graphical passwords instead of alphabetic ones are used as passwords in order for it to be tolerant to observation attacks. Several techniques for password authentications have been discussed in various studies. However, there has not been sufficient research on authentication methods that use pass-images instead of pass-texts.

This article proposes a user authentication method that is tolerant to attacks when a user's pass-image selection operation is video recorded twice. In addition, usage guidelines recommending eight pass-images are proposed, and its security is evaluated.

## I. INTRODUCTION

User authentication is widely used in automatic teller machines (ATMs) and many Internet services. A four-digit personal identification number (PIN) or a textual password is commonly used for user authentication. In Japan in October 2005, an ATM password was stolen using a wireless charge-coupled device (CCD) camera recording. The criminal group had set up many cameras at various ATMs in Tokyo. The bank's investigation revealed that user operation was captured by hidden cameras at more than 60 ATMs in the metropolitan area [1][2].

Biometric authentication technology and sneak shot camera detection technology are possible solutions [3][4][5][6] to this problem. However, because there are many ATMs installed around the world and the aforementioned solutions require additional equipment, the problem is still not solved.

In this article, we discuss an authentication method that uses pass-images instead of a textual password. In Japan, alphabetic characters are commonly used as authentication passwords for Internet services. However, alphabetic characters are not so familiar to elderly and younger people. Thus, authentication using pass-images might become a widely accepted method. In addition, this article discusses an authentication method that is tolerant to video-recording attacks. The security of the proposed authentication method is evaluated against random and video-recording attacks.

The remainder of this article is organized as follows: Chapter II describes the requirements of the pass-image authentication method. Chapter III briefs the existing techniques. Chapter IV explains the proposed authentication

method and Chapter V reports its security evaluation. Chapter VI discusses the usability of the method. Chapter VII describes the usage guidelines and Chapter VIII summarizes the article.

## II. REQUIREMENTS

We assume the use of pass-images at ATMs. The security of the authentication method is evaluated from the following two viewpoints:

### (1) Random attack

This is an attack that attempts to pass the authentication process by random operation. Because a four-digit PIN is used at ATMs, we adopt a success rate of less than 1/10000 as a requirement for a random attack.

### (2) Video-recording attack

Currently, many cell phones and handheld devices are equipped with a camera. In addition, wireless CCD cameras are inexpensive. Therefore, the risk of sneaking a shot is increasing.

At an ATM, password authentication may be conducted more than once; for example, in the case of multiple bank transfers. Therefore, we should be concerned about multiple video recordings of the pass-image selection operation.

The success rate of video-recording attacks is not standardised. However, because the success rate of a random attack is 1/10000, we adopt the same for a video-recording attack.

## III. RELATED WORK

Few studies on the authentication methods that use pass-texts have discussed observation attacks [7][8][9][10][11][12].

In [7], a password authentication technique called PIN Entry, which uses numeric key entry, is proposed. On the display, a white or black background is randomly shown. A user does not designate a password, but selects white or black as the password's background colour. To enter a password entry of one digit, a user designates the background colour by a different colour pattern four times. This method is safe against shoulder surfing; however, if the input operation is video recorded, the password can be easily discovered.

In [8], an interface for the textual password called S3PAS is proposed. Many characters are displayed on the interface. A user designates three points where a pass-character is in-

cluded in the triangle. This method is also safe against shoulder surfing; however, if the input operation is video recorded, the password can be easily discovered.

In [9] and [10], an authentication method called fake-Pointer is proposed, which uses numeric key entry. In this method, a disposable ‘answer selection information’ must be retrieved before each authentication. This answer selection information specifies a background mark such as a diamond, square, circle, or octagon for the displayed numeric password. At the time of authentication, a user presses the enter button that adjusts the password according to the background mark. If the answer selection information can be safely retrieved before each authentication, it is tolerant to video-recording attacks by recording twice. However, the studies do not discuss how to safely retrieve the information.

A textual password entry interface called mobile authentication is proposed in [11]. In this method, all the selectable texts are arranged in a square. Each text has a background colour. Each password is alphanumeric, and the texts are arranged in a  $10 \times 5$  square in which 10 colours are used. Each colour appears only once in each row. The colour pattern of a row is the permuted colour pattern of another row. In this method, a user provides a password and the correct background colours beforehand. During password entry, the user changes the background colour of a pass-character until it matches the correct background colour, and then presses the enter button. Although this technique has the restriction that all available texts must be displayed on the authentication interface, it is secure against video-recording attacks by recording twice.

Next, we review the methods in which pass-images instead of textual passwords are used.

In [13], a method called Déjà vu is proposed. In this method, a user selects five pass-images beforehand from numerous images produced by the computer. During authentication, a user selects a pass-image from 25 images displayed on the screen. Because a mechanically produced image is difficult for a user to memorize, [14] proposes the use of facial images as pass-images.

The techniques described in [13] and [14] are not safe against shoulder surfing because a user specifies a pass-image using a keyboard or mouse.

In [15], a method using graphical passwords is proposed. This method is similar to that described in [8]. This method is ambiguous, and security evaluation against shoulder surfing is not sufficient.

In the AWASE-E method [16], 25 images including one correct pass-image are displayed on the screen similar to those described in the methods in [13] and [14]; however, this method also allows the display of a screen on which no pass-image is present. If the pass-image is not present on the screen, a user must select the ‘no pass-image button’. Although this technique is highly ambiguous, its security against sneaking a shot is not sufficient.

Thus, there is no report on a pass-image authentication method tolerant to video-recording attacks where user operations are video recorded multiple times.

## IV. PROPOSED AUTHENTICATION METHOD

### A. Requirements for authentication interface

An authentication method is expected to be tolerant to video-recording attacks. Although a user’s selection operation of pass-images is video recorded, many pass-image candidates must exist when an attacker analyses the recorded video. To this end, providing secret information beforehand, such as correct position of each pass-image in the interface, is one solution, which is analogous to the technique used in [11]. However, it increases the amount of information that a user needs to memorize.

Thus, an authentication method must satisfy the following requirements:

- It should have sufficient ambiguity in pass-image selection operation in case the operation is video recorded and analysed.
- Any additional information except pass-images should not be asked beforehand.

### B. Authentication interface

When the password is “ffchopin”, it is an example of 8-length alphabetic password. Each character is chosen from 26 alphabetic characters. In this article, pass-images are used instead of characters. We assume there are  $N$  different images. And each pass-image is chosen from these  $N$  images. The password generally consists of number of pass-images. When the password is composed of eight pass-images, we use the description that the length of pass-images is eight. We use  $L$  to indicate the length of pass-images.

We proposed the authentication interface shown in Fig. 1. In the authentication interface, depth ( $D$ )  $\times$  width ( $W$ ) images are randomly selected and displayed.



A display example with 4 $\times$ 7 images

Fig. 1 Authentication Interface

For authentication operation, a user presses the following:

- Move button  
If a pass-image is displayed and a user wants to move it, a user uses the arrow button to move the image.
- Flash button

If no pass-image is displayed, a user presses the flash button to redisplay a new set of images.

- Selection button

If a pass-image is suitably positioned, a user presses the selection button. The system then shows a new display for the next pass-image selection.

The selection operation should be done for each pass-image. When the password is composed of  $L$  pass-images, the selection operation is repeated  $L$  times.

### C. Row restriction of each pass-image

In this article, we restrict each pass-image location in the authentication operation assuming the number of rows in the interface to be four. Following are the rules:

- The first pass-image is located at any place on the display. Assume that a user presses the selection button placing the first pass-image in the  $d_1$ -th row on the display.
- For the  $k$ -th ( $k \leq 4$ ) pass-image, a user can press the selection button placing the  $k$ -th pass-image in the  $d_k$ -th row, where  $d_k$  is not equal to  $d_1, d_2, \dots, d_{k-1}$ .
- For the  $k$ -th ( $k > 4$ ) pass-image, a user must press the selection button placing the pass-image in the  $d_{k-4}$ -th row.

The authentication system judges that a user is the authentic user when each  $L$  pass-image on the interface follows the aforementioned rules.

These rules are intended to make the method tolerant to random attacks while satisfying the requirements for the authentication interface. In addition, the aforementioned row restriction does not increase the amount of information that a user needs to memorize. The position of each pass-image is not recorded beforehand. A user selects suitable pass-image positions freely, following the aforementioned rules.

## V. SECURITY EVALUATION

There are different ways to consider authentication using the pass-images. If  $L$  pass-images are selected using a user's favourite story, the order of the pass-image is important and each pass-image has its own order. If the user selects his/her favourite  $L$  pictures randomly, the order of the pass-images is not important.

In evaluation, we assume the following two schemes:

- Ordered pass-image

The number of the pass-images is  $L$ . A user selects  $L$  pass-images and a sequence of pass-images is registered beforehand. The authentication must be achieved using the registered pass-image sequence.

- Non-ordered pass-image

A user selects and registers  $L$  pass-images beforehand. A user can select  $L$  pass-images in random order. The authentication must be achieved  $L$  times with each different pass-image.

### A. Security against random attacks

#### (1) Ordered pass-image

We assume that each pass-image is chosen from  $N$  different images. If  $N$  is large enough, the success probability of random attack is very small.

Figs. 2, 3 and 4 show success probabilities of random attacks. Each value is a mean value of the simulation conducted one million times.

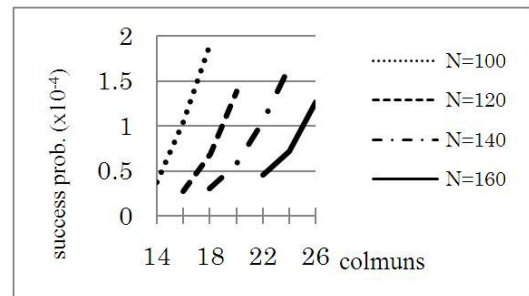


Fig. 2 Success probability of random attacks ( $L = 7$ )

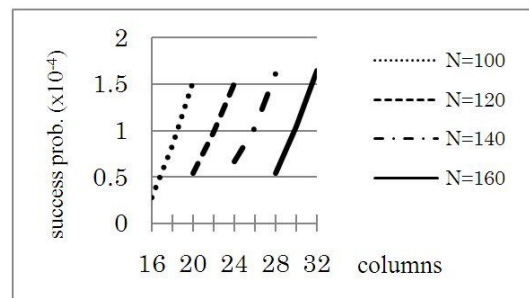


Fig. 3 Success probability of random attacks ( $L = 8$ )

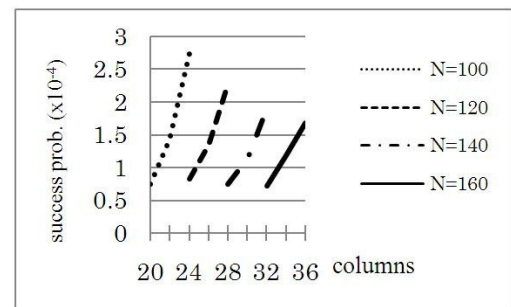


Fig. 4 Success probability of random attacks ( $L = 9$ )

The number of columns in the authentication interface and the pass-image length vary in the evaluation. The number of rows in the interface is fixed to four. When the number of columns increases, the number of images on the display also increases, thus increasing the success probability of random attacks.

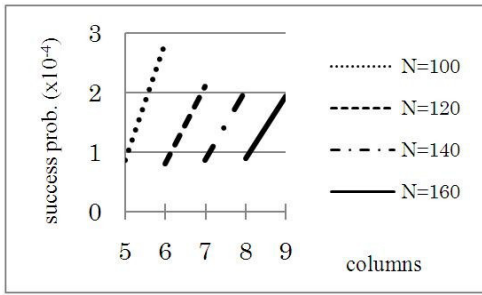
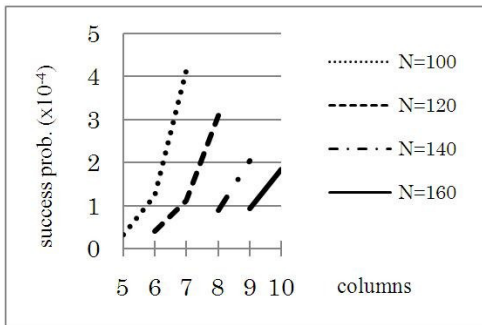
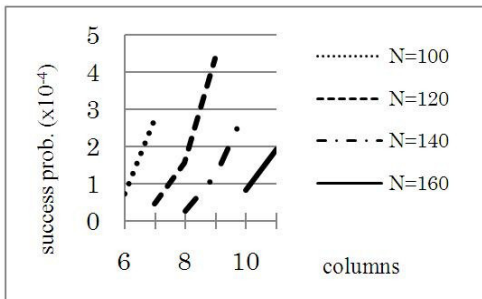
A safe range of the number of columns against random attacks is summarized in Table I. In the table, each value is the maximum number of columns in which the success probability of random attacks does not exceed  $1/10000$ .

TABLE I  
SAFETY RANGE OF COLUMNS

	$L = 7$	$L = 8$	$L = 9$
$N = 160$	$\sim 25$	$\sim 29$	$\sim 35$
$N = 140$	$\sim 20$	$\sim 25$	$\sim 29$
$N = 120$	$\sim 19$	$\sim 22$	$\sim 24$
$N = 100$	$\sim 15$	$\sim 18$	$\sim 20$

## (2) Non-ordered pass-image

Figs. 5, 6 and 7 show success probabilities of random attacks. Each value is also a mean value of the simulation conducted one million times.

Fig. 5 Success probability of random attacks ( $L = 7$ )Fig. 6 Success probability of random attacks ( $L = 8$ )Fig. 7 Success probability of random attacks ( $L = 9$ )

The safe range of the number of columns in the authentication interface against random attacks is summarized in Table II. In the table, each value is the maximum number of columns in which the success probability of random attacks does not exceed  $1/10000$ .

TABLE II.  
SAFE RANGE OF COLUMNS

	$L = 7$	$L = 8$	$L = 9$
$N = 160$	~8	~9	~10
$N = 140$	~7	~8	~8
$N = 120$	~6	~6	~7
$N = 100$	~5	~5	~6

## B. Video-recording attacks

Video-recording attack is very serious; attackers record users' password input operation by using video cameras. In addition, multiple authentication operations are assumed to be recorded. As the first step, we assume that two different authentication operations are recorded in this article. We denote them as  $s_{11}, s_{12}, \dots, s_{1L}$  and  $s_{21}, s_{22}, \dots, s_{2L}$ , where  $L$  is the length of the pass-image sequence. For example, these authentication operations could be recorded yesterday and today, respectively, and  $s_{12}$  would indicate the screen shot of yesterday's second pass-image selection.

Attackers analyse videos and attempt to obtain the pass-images as follows:

## (1) Ordered pass-images

When ordered pass-images are used, the first pass-image must appear in the first authentication. Let the shots of the authentication interface in the first pass-image selection of each authentication be  $s_{11}$  and  $s_{21}$ . In both the shots, the first pass-image must appear. Similarly, the second pass-image must be included in both  $s_{12}$  and  $s_{22}$ . In this way, attackers analyse video and attempt to obtain the pass-images. If a sequence of the images follows the following conditions, it is a possible pass-image candidate:

- 1) The sequence of the pass-images consists of  $L$  images. It is described as  $c_1, c_2, \dots, c_L$ .
- 2) The correct pass-image  $c_k$  must appear in both  $s_{1k}$  and  $s_{2k}$ .
- 3) For each image involved in the sequence of the pass-images, row restriction is satisfied for the first and second set of operations.

Through the aforementioned analysis, attackers obtained several ordered pass-image sequences whose length is  $L$ . When attackers attempt to obtain the correct sequence of the pass-images, they use each possible pass-image sequence. Thus, it is considered to satisfy the requirements described in Chapter II when more than 10000 possible pass-image sequences exist.

## (2) Non-ordered pass-images

When non-ordered pass-images are used, a candidate for the pass-image sequence must satisfy the following conditions:

- 1) The sequence of the pass-images consists of  $L$  images. It is described as  $c_1, c_2, \dots, c_L$ .
- 2) When image  $c_1, c_2, \dots, c_L$  appears in  $s_{11}, s_{12}, \dots, s_{1L}$  in this order, image  $c'_1, c'_2, \dots, c'_L$  must appear in  $s_{21}, s_{22}, \dots, s_{2L}$  in the order where  $c'_1, c'_2, \dots, c'_L$  is a permutation of  $c_1, c_2, \dots, c_L$ .
- 3) For each image involved in  $c_1, c_2, \dots, c_L$ , row restriction is satisfied for the first set of operations  $s_{11}, s_{12}, \dots, s_{1L}$ . In addition, for each image involved in  $c'_1, c'_2, \dots, c'_L$ , row restriction is satisfied for the second set of operations  $s_{21}, s_{22}, \dots, s_{2L}$ .

Through the aforementioned analysis, attackers obtain several ordered pass-image sequences whose length is  $L$ . When attackers attempt to obtain the correct sequence of the pass-images, they use each possible pass-image sequence.

However, assuming two different pass-image sequences  $c_1, c_2, \dots, c_L$  and  $c_1', c_2', \dots, c_L'$ , where  $c_1', c_2', \dots, c_L'$  is a permutation of  $c_1, c_2, \dots, c_L$ , when it is clarified that  $c_1, c_2, \dots, c_L$  is not a correct sequence of the pass-images, attackers need not try with  $c_1', c_2', \dots, c_L'$ , because the order of the pass-images is not important in this case, and  $c_1', c_2', \dots, c_L'$  is not a correct sequence of the pass-images. Thus, it is considered to satisfy the requirements described in Chapter IV when more than 10000 possible pass-image candidates exist.

### C. Security against video-recording attacks

#### (1) Security evaluation for ordered pass-images

Figs. 8, 9 and 10 show the number of the pass-image candidates obtained through video analysis. Each value is a mean value of the simulation conducted 100 times.

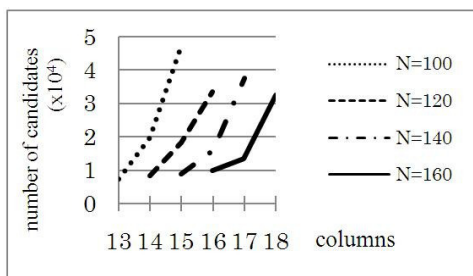


Fig. 8 Number of pass-image candidates (L = 7)

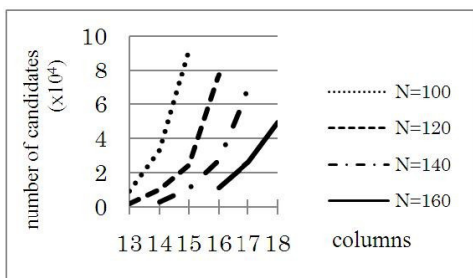


Fig. 9 Number of pass-image candidates (L = 8)

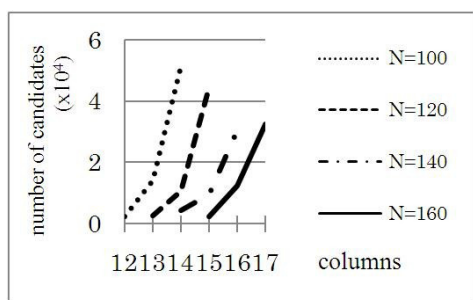


Fig. 10 Number of pass-image candidates (L = 9)

The results are briefly summarized in Table III. For example, when the pass-image length is eight and the total number of images is 160, 16 or more columns are required in the authentication interface for the method to be tolerant to video-recording attacks.

TABLE III.  
SAFE RANGE OF COLUMNS

	L = 7	L = 8	L = 9
N = 160	17~	16~	16~
N = 140	16~	15~	16~
N = 120	15~	14~	14~
N = 100	14~	14~	13~

#### (2) Security evaluation for non-ordered pass-images

Figs. 11, 12 and 13 show the number of the pass-image candidates obtained through video analysis. Each value is a mean value of the simulation conducted 100 times.

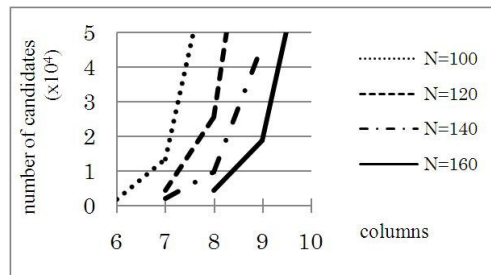


Fig. 11 Number of set of pass-image candidates (L = 7)

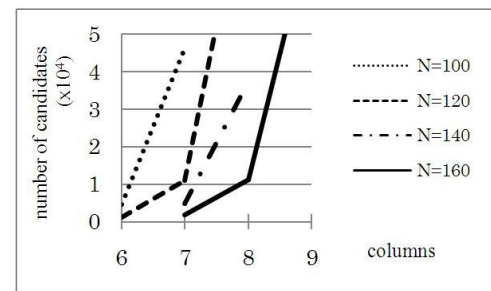


Fig. 12 Number of set of pass-image candidates (L = 8)

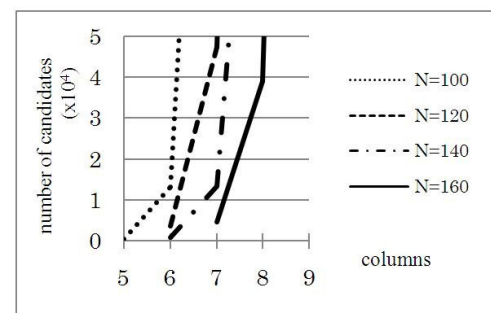


Fig. 13 Number of set of pass-image candidates (L = 9)

The results are summarized in Table IV. For example, when the pass-image length is eight and the total number of images is 160, eight or more columns are required in the authentication interface for the method to be tolerant to video-recording attacks.



TABLE IV.  
SAFE RANGE OF COLUMNS

	L = 7	L = 8	L = 9
N = 160	9~	8~	8~
N = 140	9~	8~	7~
N = 120	8~	7~	7~
N = 100	7~	7~	6~

#### D. Security against both attacks and discussion

The range of columns in the authentication interface for the method to be tolerant to both types of attacks is shown in Tables V and VI.

TABLE V.  
SAFE RANGE OF COLUMNS (ORDERED PASS-IMAGES)

	L = 7	L = 8	L = 9
N = 160	17 ~ 25	16 ~ 29	16 ~ 35
N = 140	16 ~ 20	15 ~ 25	16 ~ 29
N = 120	15 ~ 19	14 ~ 22	14 ~ 24
N = 100	14 ~ 15	14 ~ 18	13 ~ 20

TABLE VI.  
SAFE RANGE OF COLUMNS (NON-ORDERED PASS-IMAGES)

	L = 7	L = 8	L = 9
N=160	-	8 ~ 9	8 ~ 10
N=140	-	8	7 ~ 8
N=120	-	-	7
N=100	-	-	6

The values in Table V are higher than those in Table VI, which implies that authentication with ordered pass-images requires many columns in the interface for the method to be tolerant to video-recording attacks. Many columns indicate a wide interface that is considered unsuitable for use.

Authentication with non-ordered pass-images is considered to be superior to that with ordered pass-images in terms of the interface.

#### VI. USABILITY OF THE METHOD

In this section, we evaluate and discuss the usability of the proposed pass-image authentication method. We prepared several photo images in eight categories, which included photographs of animals, scenery, vehicles and food. We asked our colleagues to use the authentication method. We faced a difficulty when some colleagues found selecting and memorizing eight images to be a difficult task.

If photographs captured by each user are used as pass-images, memorizing them is much easier; but it is known that there is a problem in security because it is possible to narrow down the pass-image candidates paying attention with photographer's preference and conditions of taking photographs.

Another solution is to memorize the images by using a story. However, this was ruled out in the discussion of the evaluation results in this article.

We attempt to solve this problem in the following section.

#### VII. USAGE GUIDELINES

Considering the suggestions of our colleagues, we introduced usage guidelines for the proposed method summarized as follows:

- The length of the pass-image sequence is eight pass-images.
- Four category images are used for authentication. A user selects two pass-images in each category.
- For authentication, two pass-images in the same category are continuously used. Thus,  $2k$ -th and  $(2k + 1)$ -th pass-images must fall in the same category.
- Row restriction for the pass-images must be satisfied.

These usage guidelines are rather easy to follow; however, whether the authentication method following these guidelines is tolerant to both types of attacks is unclear. We assume that attackers have the same information as normal users; for example, they know the correct category of each image.

The results of security evaluation are shown in Table VII. It shows a safe range of columns in the authentication interface.

TABLE VII.  
SAFE RANGE OF COLUMNS

	columns	Number of pass-image candidates	Success rate of random attacks ( $\times 10^{-6}$ )	Number of operations per one selection
N = 160	10	$1.4 \times 10^4$	0	3.8
	11	$5.2 \times 10^4$	2	3.6
	12	$1.6 \times 10^5$	6	3.4
	13	$3.2 \times 10^5$	14	3.2
	14	$1.8 \times 10^6$	17	3.1
	15	$4.0 \times 10^6$	26	3.0
N = 140	16	$9.5 \times 10^6$	61	2.9
	10	$3.0 \times 10^4$	5	3.5
	11	$9.9 \times 10^4$	6	3.3
	12	$3.6 \times 10^5$	26	3.1
	13	$1.3 \times 10^6$	44	3.0
N = 120	14	$3.4 \times 10^6$	57	2.9
	15	$1.1 \times 10^7$	90	2.8
	9	$1.6 \times 10^4$	4	3.4
	10	$1.0 \times 10^5$	18	3.2
N = 100	11	$2.8 \times 10^5$	31	3.0
	12	$8.1 \times 10^5$	71	2.9
	9	$5.1 \times 10^4$	31	3.1
	10	$2.4 \times 10^5$	75	2.9

In the table, the average button operation frequency is shown on the right. In [19], keystrokes per character (KSPC) on a mobile terminal are discussed and the operation frequency is considered as a measure to evaluate the user interface.

For example, when 100 images are used (25 images for each category), authentication is tolerant to both random and video-recording attacks in the case of a 9- or 10-column and 4-row interface. Also, the value of KSPC is relatively small and easy to use.

Conventionally, there is no authentication method which satisfies followings:

- The authentication method, that uses pass-images instead of textual password, is tolerant to random and video-recording attacks even if the operation is video recorded twice.
- Any additional information except pass-images should not be registered beforehand. In a case of other method, sometimes users are required to memorize additional information, such as the correct place in the interface for each pass-image or pass-text.

A user's authentication operation must be satisfied the "row restriction" described in Chapter IV. But it may be described in the authentication interface for user's help. It is no need to memorize it.

In this article, we use random selection and describe statistical data. In the case where  $N = 100$  and 4-row 10-column display is used in the authentication interface, the average number of pass-image candidates is  $2.4 \times 10^3$ , where the standard deviation is  $9.7 \times 10^4$ . Assuming the values as normally distributed, a 95% confidence interval is  $1.9 \times 10^4 \pm$  the mean value. A 99% confidence interval is  $2.6 \times 10^4 \pm$  mean value.

#### VIII. CONCLUSION

This article proposed a user authentication method that uses pass-images instead of textual password. Fundamental characteristics of the authentication method are clarified, and the proposed authentication method is shown to be tolerant to random and video-recording attacks even if the operation is video recorded twice. The method does not require a user to register additional information except pass-images.

In the discussion of usability, usage guidelines for eight pass-images are proposed. In addition, it is shown to be tolerant to both random and video-recording attacks when authentication is used in accordance with the guidelines.

#### REFERENCES

- [1] The Mitsubishi Tokyo UFJ bank, 'A bank report about that the camera was put on secretly at the ATM machine by some person'.  
[http://www.bk.mufg.jp/info/ufj/ufj\\_20051101.html](http://www.bk.mufg.jp/info/ufj/ufj_20051101.html)
- [2] Bank of Yokohama, 'A bank report about that equipment for the sneak shot was installed in the unmanned agency (the ATM out of the store)'.  
<http://www.boy.co.jp/info/pdf/9.pdf>

- [3] M. Une, T. Matsumoto, 'About the fragilitas about the living body authentication: It studies mainly a fragilitas about the counterfeiting of a stigma by the finance', vol.24, no.2, pp.35-84 (2005)
- [4] Banno, 'The recent trend, the forensic science technology of the living body authentication technology', vol.12, no.1, pp.1-12 (2007)
- [5] Secom Co., Ltd., 'It begins' the ATM sneak shot damage prevention service 'by the offer'  
[http://www.secom.co.jp/corporate/release/2006/nr\\_20060814.html](http://www.secom.co.jp/corporate/release/2006/nr_20060814.html)
- [6] NEC, 'The service of the investigation of the detecta-phone and the sneak shot receptacle'  
<http://www.necf.jp/solution-service/office/hidden-mic-camera/>
- [7] V. Roth, K. Richter, R. Freidinger, 'A Pin-Entry Method Resilient Against Shoulder Surfing', CCS'04, pp.236-245 (Oct 2004)
- [8] H. Zhao, X. Li, 'S3PAS: A Scalable Shoulder-Surfing Resistant Textual-graphical Password Authentication Scheme', IEEE Advanced Information Networking and Applications Workshops 2007, pp.467-472 (2007)
- [9] T. Takada, 'fakePointer: The authentication technique which has tolerance to video recording attacks', IPSJ transaction, vol.49, no.9, pp.3051-3061 (Sep 2008)
- [10] T. Takada, 'fakePointer2: The proposal of the user interface to improve safety to the peep attack about the individual authentication', Cryptography and Information Security Symposium, SCIS2007 (2007)
- [11] Sakurai, Yoshida, Bunaka, 'Mobile authentication method', Computer Security Symposium 2004, pp.625-630 (Oct 2004)
- [12] X. Suo, Y. Zhu, G. S. Owen, 'Graphical Passwords: A Survey', 21<sup>st</sup> Annual Computer Security Applications Conference, ACSAC 2005 (2005)
- [13] R. Dhamija and A. Perrig, 'Déjà vu: A User Study Using Images for Authentication', 9<sup>th</sup> Usenix Security Symposium, pp.45-58 (Aug, 2000)
- [14] RealUser: <http://www.realuser.com/>
- [15] L. Sobrado, J. Birget, 'Graphical passwords', The Rutgers Scholar, An Electronic Bulletin for Undergraduate Research, vol. 4 (2002)
- [16] T. Takada, H. Koike, 'Awase-E: Image-Based Authentication for Mobile Phones Using User's Favorite Images', LNCS2795. Human-Computer Interaction with Mobile Devices and Services, pp.347-351 (2003)
- [17] T. Pering, M. Sundar, J. Light, R. Want, 'Photographic Authentication through Untrusted Terminals', IEEE Pervasive Computing, vol.2, no.1, pp.30-36 (2003)
- [18] W. Ku, M. Tsaar, 'A Remote User Authentication Scheme Using Strong Graphical Passwords', IEEE Local Computer Networks, LCN'05 (2005)
- [19] I. S. MacKenzie, 'KSPC (keystrokes per Characters) as a Characteristic of Text Entry Techniques', Proc. Mobile HCI '02, LNCS-2411, Berlin, pp.405-416, Springer-Verlag (2002)





# Risks Awareness and Management through Smart Solutions

**A**RISK is something at a borderline, which does not guarantee the success and cannot be sure of the failure. Risks are present everywhere, in the daily life (e.g., cross the street, drive a car, eat fish), in business (e.g., investments, hiring), or in IT (e.g., adopt a new IT solution, develop a software).

Challenges concerning risks are mainly related to become aware of them, to explicitly consider them, to avoid them if possible, to neutralize them when possible, and to be ready to face them properly.

The aim of this event is to focus on the identification and modeling of risks and their characteristics, as well as on the solutions adopted to address risks in various application domains. In particular, we are interested to explore if and how the advances in the IT domain may provide support to identify and address/manage risks. Risks are strongly related to business aspects. For example, financial risks may change business models, security risks may lower the customers' confidence, or work risks may lead to a high turn-over of the employees.

We consider that a system of any type may work in three different conditions: normal, when the executing conditions are proper for the system functioning, risk, when one or more executing conditions tend to stay or overcome their normal limits, and emergency, when a damage has been verified in the execution conditions. Hence, risks may be seen as pre-emergencies. The challenge is to monitor the system and to identify risks to address them and avoid emergencies. Of course, this is not always possible: emergencies may also occur suddenly without any prelude.

The participants to this event will be encouraged to discuss the IT advances (from design to available technologies) which can be successfully exploited in risk management systems in any application domain. For example, the MAPE (Monitoring, Analyzing, Planning, Executing) loop or Deming (Plan-Do-Check-Act) loop can be used to design risk management systems. Or, sensor networks or wearable technologies can be used to implement risk management systems.

## TOPICS

Topics include (but are not limited to):

- Risk definition, features, and application domains;
- Risk management systems: analysis, design, and implementation;
- Risk management systems: evaluation and testing;
- Case studies;

- Smart environments for risk prevention;
- Legislation and rules;
- Business risk;
- Risk and recovery;
- Security and trust;
- Tools and business intelligence strategies for risk prevention;
- Risk awareness in self-healing systems;
- Semantic enhancement of software for risk awareness;
- Open software for training and education to risk procedures.

## PROGRAM COMMITTEE

**Crespo Adalberto**, Center for Information Technology, Brasil

**Francesca Arcelli Fontana**, University of Milano-Bicocca, Italy

**João Camargo**, São Paulo University, Brasil

**Mirko Cesarini**, University of Milano-Bicocca, Italy

**Nicoletta Dessi**, University of Cagliari, Italy

**MariaGrazia Fugini**, Politecnico di Milano, Italy

**George C. Hadjichristofi**, University of Cyprus, Cyprus

**Ronald Israels**, Quint Wellington Redwood, The Netherlands

**Alexander Kipp**, University of Stuttgart, Germany

**Nabuco Olga**, CTI, Brazil

**Claudia Raibulet**, University of Milano-Bicocca, Italy

**Filippo Ramoni**, Politecnico di Milano, Italy

**Thamarai Selvi**, Anna University Chennai, India

**Francesco Tisato**, University of Milano-Bicocca, Italy

**Luigi Ubezio**, I3B, Italy

**Ramon Salvador Valles**, Universitat Internacional de Catalunya, Spain

## ORGANIZING COMMITTEE

**MariaGrazia Fugini**, Politecnico di Milano, Dipartimento di Elettronica e Informazione, Italy  
fugini@elet.polimi.it

**George C. Hadjichristofi**, Department of Computer Science and Engineering Frederick University, Cyprus  
com.hg@fit.ac.cy

**Ronald Israels**, Quint Wellington Redwood, Netherlands  
r.israels@quintgroup.com

**Claudia Raibulet**, Università degli Studi di Milano-Bicocca, Dipartimento di Informatica, Sistemistica e Comunicazione, Italy raibulet@disco.unimib.it



## Enhancing DNS Security using Dynamic Firewalling with Network Agents

Joao Afonso

Foundation for National Scientific Computing  
Lisbon, Portugal  
e-mail: joao.afonso@fccn.pt

Pedro Veiga

Department of Informatics  
University of Lisbon  
Lisbon, Portugal  
e-mail: pedro.veiga@di.fc.ul.pt

**Abstract**—In this paper we propose a solution to strengthen the security of Domain Name System (DNS) servers associated with one or more Top Level Domains (TLD). In this way we intend to be able to reduce the security risk when using major internet services, based on DNS. The proposed solution has been developed and tested at FCCN, the TLD manager for the .PT domain. Through the implementation of network sensors that monitor the network in real-time, we are capable to dynamically prevent, detect or limit the scope of attempted intrusions or other types of occurrences to the DNS service. The platform relies heavily on cross-correlation allowing data from a particular sensor to be shared with the others. Administration tasks such as setting up alarms or performing statistical analysis are made through a web-based interface.

**Index Terms**—DNS; risk; security; intrusion detection system; real-time; monitoring.

### I. INTRODUCTION

**O**BSERVING internet usage and world population statistics [1] updated on March 2011, there are 30.2% internet users – of the estimated world population of 6.8 billion. If we take a closer look to Europe this value increase to 58.3 % (with a growth rate of 353.1% between 2000 and 2011) and in North America, there are 78.3 % of internet users (growth rate of 151.7% at same period), as shown in Fig. 1.

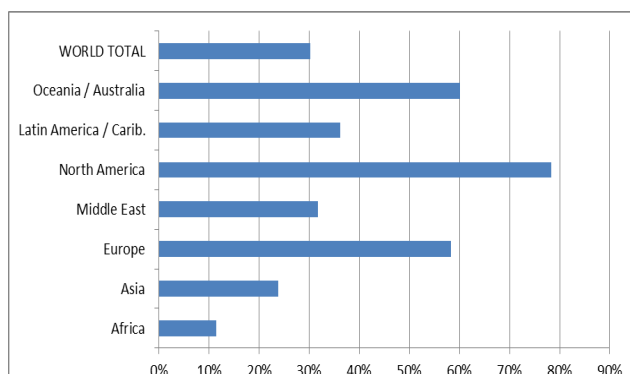


Figure1. Internet penetration (% population)

The DNS service is required to access e-mail, browse Web sites, and is needed for normal operation in all major services in the Internet (most of them use critical information, like e-banking).

Taking care of the huge number of internet users, and the risk associated with the fact that all major applications requires the DNS service, there is a security risk needed to be reduced.

DNS servers assume a pivotal role in the regular running of IP networks today and any disruption to their normal operation can have a dramatic impact on the service they provide and on the global Internet.

Although based on a small set of basic rules, stored in files, and distributed hierarchically, the DNS service has evolved into a very complex system [2].

According to other recent studies [3], there are nearly 11.7 million public DNS servers available on the Internet.

It is estimated that 52% of them allow arbitrary queries (thus allowing the risks of denial of service attacks or “poisoning” of the cache).

They are still nearly 33% of the cases where the authoritative nameservers of an area are on the same network, which facilitates the attacks of Denial of Service (DOS).

Furthermore, the type of attacks targeting the DNS are becoming more sophisticated, making them more difficult to detect and control on time.

Examples are the attacks by Fast Flux (ability to quickly move the DNS information about the domain to delay or evade detection) and its recent evolution to Double Flux [4].

A central aspect of a security system is the ability to collect statistically useful information about network traffic. This information can be used to monitor the effectiveness of the protective actions, to detect trends in the collected data that might suggest a new type of attack or simply to record important parameters to help improve the performance of the service.

The fact that the DNS is based on an autonomous database, distributed by hierarchy, means that whatever solution we use to monitor, it must respect this topology. In this paper we propose a distributed system using a network of sensors, which operate in conjunction with the DNS servers of one or more TLDs, monitoring in real-time the data that passes through them.

The ability to perform real-time analysis is crucial in the DNS area since it may be necessary to act in case of abuse, by blocking a particular access, and notifying the other sensors on the origin of the problem, since several types of attacks are directed to other DNS components

The use of a Firewall solution whose triggering rules are dynamically generated by the network sensors is a fundamental component of the system, to filter attacking systems and returning to the initial situation when the reason to filter different traffic patterns has ceased to exist, guarantees an autonomous functioning of the platform. Special care was taken to minimize the detection of false negatives and positives.

The remaining of the paper is structured as follows: Section 2 provides background information regarding related work. Section 3 introduces System Requirements. In section 4 we describe the proposed solution. Section 5 presents a case study for validation of the proposal. In Section 6 the results gathered in the case study are analyzed. Finally, Section 7 presents some conclusions and directions for further work.

## II. RELATED WORK

One of the first studies that can be witnessed in this area has the authorship of Guenter and Kolar, with a tool entitled *sqljbdns* [5]. Their application uses a modified version of the traditional BIND [6] working together with a Structured Query Language (SQL) version inside a Relational database management system (RDBMS). For DNS clients, this solution is transparent and there is no difference from classic BIND.

Zdrnja presented a system for Security Monitoring of DNS traffic [7], using network sensors without interfering with the DNS servers to be monitored. This is a transparent solution that does not compromise the high availability needed for the DNS service.

Vixie proposed a DNS traffic capture utility called, *DNSCap* [8]. This tool is able to produce binary data using pcap format, either on standard output or in successive dump files. The application is similar to *tcpdump* [9] – command line tool for monitoring network traffic, and has finer grained packet recognition tailored for DNS transactions and protocol options, allowing for instance to see the full DNS message when *tcpdump* only shows a one-line summary.

Another tool available is *DSC - DNS Statistics Collector* [10]. *DSC* is an application for collecting and analyzing statistics from busy DNS servers. Major features include the ability to parse, summarize and search inside DNS queries detail. All data is stored in an SQL database. This tool, can work inside a DNS server or in another server that "captures" bi-directional traffic for a DNS node.

Kristoff also proposed an automated incident response system using BIND query logs [11]. This particular system, besides the common statistical analysis, also provides information regarding the kind of consultations operated. All information is available through the Web based portal. Each security incident can result in port deactivation.

## III. METHODOLOGY

### A. Architecture

To be able to reduce the incident risk in DNS operation, the architecture of the system that we have developed aims to improve the security, performance and efficiency of the DNS protocol, removing all unwanted traffic and reinforce the resilience of a Top Level Domain. We propose an architecture comprising an integrated protection of multiple DNS servers, working together with several network sensors that apply live rules to a dedicated firewall, acting as a traffic shaping element.

Sensors carefully located in the network monitor all the traffic going to the DNS infrastructure, identify potentially harmful traffic using an algorithm that we have developed and use this information to isolate traffic that has been identified as security threats.

Several networks sensor monitor different parts of the infrastructure and exchange information related to security attacks. In this way, as shown in Fig. 2, it should also be possible to exchange critical security information between the sensors. In addition to an increase in performance, this operation should prevent an attack on a server from a source, identified by another sensor as malicious. This scenario is relevant since some kinds of attacks are directed to several components of the DNS infrastructure.

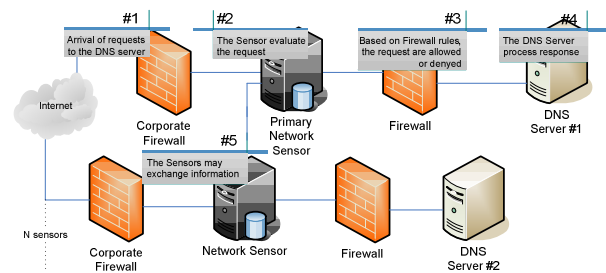


Figure2. Diagram of the desired solution

### B. Heuristic

One of the crucial parts of our work is the algorithm to identify traffic harmful to the DNS. In order to implement the stated hypothesis in the architecture and keep the DNS protocol as efficient as possible, it is necessary to apply a heuristic, which in real time, evaluates all the information collected from different sources and applies convenient weights to each component and act accordingly.

The components that we have chosen to have impact in the security incidents of DNS are: the number of occurrences, analysis of type of queries been made, the amount of time between occurrences, the number of probes affected and information reported from intrusion detection systems.

Our system uses the following formula to evaluate a parameter that measures the likelihood of the occurrence of a security incident:

$$f(x) = O \cdot 0,2 + C \cdot 0,2 + G \cdot 0,15 + N \cdot 0,25 + I \cdot 0,20$$

Are factors considered in applying this formula:

- Occurrences (O) - Represents the number of times (instances) that have given malicious source was blocked, so that the distributed then depicted in Table I.

TABLE I – CONTRIBUTION OF THE NUMBER OF OCCURRENCES OF A SOURCE IN MALICIOUS HEURISTIC

<i>Occurrences</i>	<i>Weight</i>
1	25%
2	50%
3	75%
4 or more	100%

- Analysis (C) - Real-time evaluation of the deviation of the values recorded in relation to the average observed statistics, based on the criteria and weights identified below in Table II.

TABLE II – CONTRIBUTION OF EVENTS TYPIFIED A POTENTIALLY MALICIOUS SOURCE GIVEN IN HEURISTIC

<i>Event</i>	<i>Weight</i>
Entire zone transfer attempt (AXFR)	100%
Partial transfer zone attempt (IXFR)	50%
Incorrect query volume, 50 to 75% on average per source	75%
Incorrect query volume exceeding 75%	100%
Query volume, up 50%, the average number of access by origin	50%

Note that the estimates apply the moving average, for the determination of reference values, given the ongoing development of data collected.

- Time between occurrences (G) - time since last occurrence of a given source, distributed with the weights associated to the times below is obeisant.

TABLE III – WEIGHT OF DIFFERENT TIME BETWEEN EACH OCCURRENCE

<i>Time</i>	<i>Weight</i>
Less than 1 Minute	100%
Less than 1 Hour	75%
Less than 1 Day	50%
Less than 1 Week	25%

- Incidence (N) - Number of probes that report blocks in the same address source.  
For the calculation, we observed expression:

$$\frac{1}{\#Total\_Sensors - \#Sensors\_Attacked}$$

In the above expression the factor *#Total\_Sensors* represents the number of sensors running together in the infrastructure and able to exchange information among themselves.

The other factor *#Sensors\_Attacked* stands for the number of Sensors that are current reporting security incidents.

- Intrusion Detection Systems (I) - We considered the use of the Snort platform, being free to use, and gather a large number of notarized signatures of security incidents relating to the DNS service.

TABLE IV – INTERCONNECTION WITH TEMPORAL DATA GATHERED FROM INTRUSION DETECTION SYSTEMS

<i>Metric: Common Vulnerability Scoring System (CVSS)</i>	<i>Weight</i>
Low level	34%
Middle level	67%
High level	100%

For the activation of a rule in Firewall occurs will require:

1. The formula shown above take values equal to or greater than 0.25;
2. The combination of two or more criteria of the formula.

Exception: when receiving information from all the other sensors, in which case a single criteria is sufficient;

3. It respected the existing white list in the repository, allowing considered privileged sources that are not blocked.

In this way we avoid compromising the Internet service, considering the key role played by DNS, the White List protects key addresses from being blocked in case of false positives events.

This list is created from a record of trusted sources, allowing all addresses listed here to be protected from being added to the Firewall rules.

One example is the list of internal addresses, and the DNS servers of ISPs.

Instead, for the removal of a rule in the firewall will need to occur simultaneously on the following assumptions:

1. Exceeded the quarantine period, based on the parameters in use;
2. The expression of activation (heuristic) does not (still) check the referenced source.

IV. PROPOSED SOLUTION

A. Diagram

As shown in Fig. 3, this solution is based on a network of sensor engines that analyze all traffic flowing into the DNS server in the form of valid or invalid queries, process the information received from other probes and issue restrictions for specific network addresses. In case an abnormal behavior is detected or there is suspicious behavior from a certain network address, it will be blocked in the firewall and the other probes notified so they can act accordingly. The system can also calculate the response time for each operation to evaluate the performance of the server.

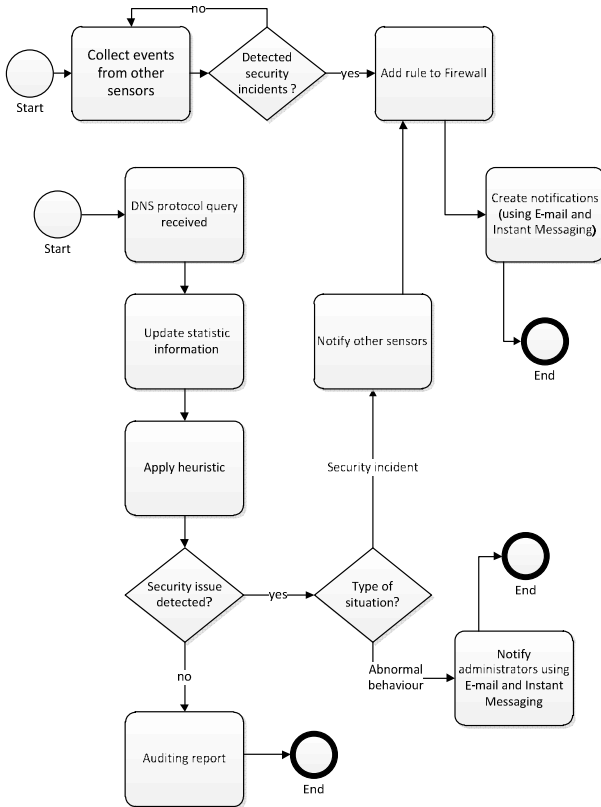


Figure 3. Block Diagram of proposed solution

For each rule inserted in the sensor firewall, there will be a period of quarantine and, at the end of this time, the sensor will evaluate the behavior of that source, to evaluate the needed to remove the rule, as shown in Fig. 4.

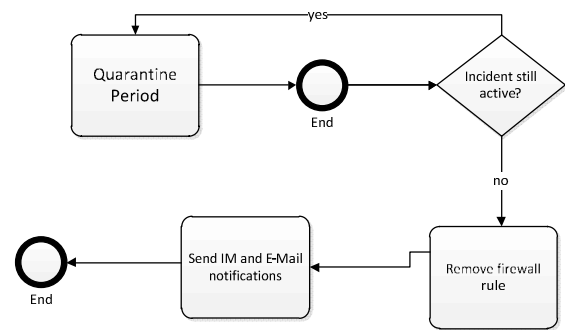


Figure 4. Quarantine procedure over the Firewall

B. Network data flow

According to our design, all data that flows through the probe heading for the DNS server is treated according to a standard set of global firewall rules, followed by specific local rules regarding to the addresses that are being blocked in real time. The queries are then delivered to the parser to be analyzed and stored in the RDBMS. At the top is the system of alarms and the Web portal (Fig. 5).

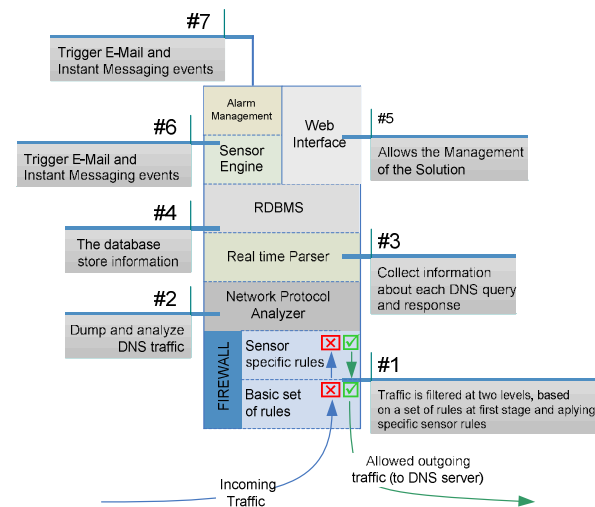


Figure 5. Network data flow

All information collected is stored in a database implemented in MySQL [12]. Taking into consideration the need to optimize the performance of the queries and to reduce the volume of information stored, the data is divided into a number of different tables.

The conversion of the IP address of source and destination (DNS server) into an integer format, has allowed for much more efficient data storage, and significant improvements in the overall performance of the solution.



The information regarding all queries made, is stored daily into a log, and kept available during the next 30 days.

Two tables containing the set of rules that are dynamically applied – add or removed, based on situations that have been triggered - control the correct operation of the firewall. For auditing purposes every action is registered.

The information required for auditing and statistical tasks never expires.

C. Statistical analysis and performance evaluation

The statistical information collected and stored in the database has a significant amount of detail. It is possible, for example, to calculate, for each sensor, the evolution of queries per unit of time (hour, day, etc) badly formatted requests, DNS queries of rare types and determine the sources that produce the larger number of consultations. It is also possible to see the standard deviation of a given measure so we can relate it to that is seen with the other hits [15].

The performance of the DNS protocol responses is permanently measured, regarding the response time per request. Data is constantly registered and an alarm is raised in case normal response times are exceeded.

V. CASE STUDY

Our proposal have been under development since September 2006 at FCCN – who has the responsibility to manage, register and maintain the domains under the .PT TLD.

At present time, there are two sensors running attached to the DNS servers (one at the primary DNS and another working together with a secondary DNS server).

The network analyzer is tshark [16], and the firewall used is IPFilter [13]. The real time parser was programmed in Java, collecting the information received from the tshark. The Web server is running Apache with PHP.

Regarding the Xmpp server [14] we choose the Jive messenger platform.

All modules are integrated together.

The entire sensor solution, as described above, as well as the web platform we developed went on-line on the 1st of January 2007, and the data from the various agents was collected from the 10th of May 2008 till now (Fig. 6).



Figure 6. Web portal

In addition to the usual operations of monitoring and collection of statistics relating to the operation of DNS service, as shown above, the solution proposed here can easily be adapt to specific situations, given the fact that it is fully configurable.

One of the applications was the event North Atlantic Treaty Organization (NATO) Summit Lisbonne 2010 at the period of 15 to 21 November 2010.

To reduce security risks in the area of the Internet involved in event, a number of areas considered most vulnerable were selected, and made a daily monitoring over them.

They were classified into four categories: law enforcement, banking, government and industry. The data collected by each sensor created a pattern of consultations for each of the categories and detect abnormal situations occurred when deviations happens from pattern (Fig. 7)

Alarmist notifications were programmed using SNMP traps.

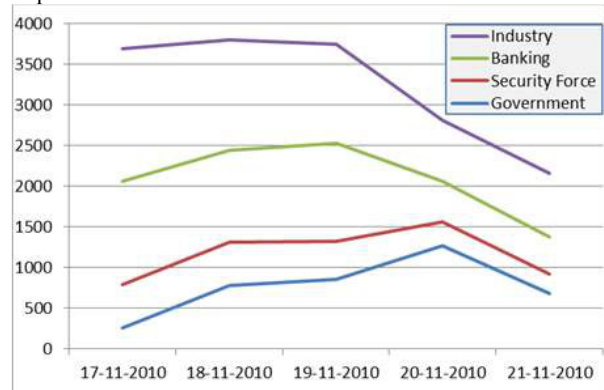


Figure 7. Monitoring DNS service at NATO event/Lisbon 2010

VI. RESULTS

We present here the results of the last 12 months of data collection (between 1st of May 2009 and 31st May 2010). The Average number of requests to the primary DNS server is up to 19,769,946 per day (228 per sec.) using last records collect on 7<sup>th</sup> August 2011.

The performance of the data analysis program is above 1240 requests processed per sec. (filtered, validated and inserted in the database).

Using the data collected by the sensors, during this time period, we were able to collect useful statistical information, e.g.:

- Daily statistics by type of DNS protocol registers accessed;
- Number of Internationalized domain name (IDN) queries;
- Number of daily queries to IPV6 AAAA DNS type (Fig. 8).

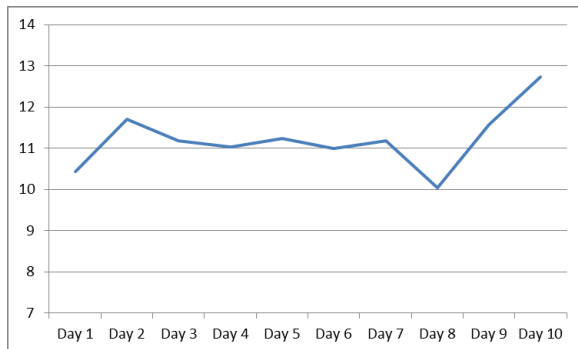


Figure 8. Statistical analysis by IPv6 records accessed (AAAA)

- Detect examples of abnormal use (that are not security incidents). For example we were able to detect that a given IP was using the primary .PT DNS server as location resolver.  
The number of queries made was excessive when compared with the average value per source, reaching values close to some Internet Service Providers that operate under the .PT domain.
- Detect situations of abuse, including denial of service attacks, with the execution of massive queries. In last 12 months of analysis there are 17 DOS attacks triggered.  
They were instantly blocked, and addresses placed in quarantine (Table V).

TABLE V. EXAMPLES WHEN THE SENSOR DETECTED SITUATIONS THAT REQUIRED THE FIREWALL RULES TO CHANGE.

Source Address	Date / Time	Operation	Sensor
xx.xx.200.45	2011-08-05 02:15:44	Add rule	xx.xx.44.63
xx.xx.17.122	2011-08-05 03:25:12	Remove rule	xx.xx.44.63
xx.xx.129.51	2011-08-05 04:47:14	Add rule	xx.xx.44.62
xx.xx.14.239	2011-08-05 05:27:29	Add rule	xx.xx.44.62
xx.xx.14.131	2011-08-05 08:35:38	Remove rule	xx.xx.44.63

## VII. CONCLUSION AND FUTURE WORK

This article has presented a novel approach to reduce the security risk on the internet applications that use DNS service. Our solution builds upon the existing solutions that collect statistical information regarding DNS services, by adding the ability to detect and control security incidents in real time. It also adds the advantage of operating in a distributed way, allowing the exchange of information between cooperating probes, and the reinforcement of its own security, even before it is threatened.

Currently, the solution presented does not allow the processing of addresses in the IPv6 format. The technical aspects that led to this situation are linked to the need to optimize the performance of the data recorder application making it possible to store the data from all consultations. One possible option for solve this issue is to change the database engine to other solution, for instance, a commercial one. Nevertheless, all queries made to IPv6 addresses are contained in this solution (AAAA types).

We are also working on extending the data correlation capabilities of the system by adding information collected from other sources (intrusion detection systems for instance). We anticipate that this could be a valuable approach to reduce considerably the number of false positives and negatives [17].

## REFERENCES

- [1] Internet Usage and World Population Statistics website, [http://www.internetworldstats.com/stats.htm]. Last accessed on 7 August 2011.
- [2] P. Vixie, "DNS Complexity", ACM Queue vol. 5, no. 3, April 2007.
- [3] D. Wessels, "A Recent DNS Survey", DNS-OARC, November 2007.
- [4] Dave Piscitello, "Conficker Summary and Review", ICANN, May 2010.
- [5] SQLDNS website, [http://home.tiscali.cz:8080/~cz210552/sqldns.html]. Last accessed on 7 August 2011.
- [6] BIND website, [http://www.isc.org/products/BIND]. Last accessed on 7 August 2011.
- [7] Bojan Zdrnja, "Security Monitoring of DNS traffic", May 2006.
- [8] Paul Vixie, D. Wessels, "DNSCAP – DNS traffic capture utility", CAIDA Workshop, July 2007.
- [9] Duane Wessels, "Whats New with DSC", DNS-OARC, November 2007.
- [10] Lawrence Berkeley National Laboratory. Tcpcdump website [http://www.tcpdump.org].
- [11] John Kristoff, "An Automated Incident Response System Using BIND Query Logs", June 2006.
- [12] MySQL website – (Open Source Database), [http://www.mysql.com]. Last accessed on 7 August 2011.
- [13] IP FILTER – TCP/IP Firewall/NAT Software, [http://coombs.anu.edu.au/~avalon]. Last accessed on 7 August 2011.
- [14] P. Saint-Andre, Ed., Extensible Messaging and Presence Protocol (XMPP): Core, RFC 3920, 2004.
- [15] João Afonso, Edmundo Monteiro, "Development of an Integrated Solution for Intrusion Detection: A Model Based on Data Correlation", in Proc. of the IEEE ICNS'06, International Conference on Networking and Services - ICNS'06, Silicon Valley, USA, July 2006.
- [16] Tshark website – The Wireshark Network Analyzer, [http://www.wireshark.org]. Last accessed on 7 August 2011.
- [17] João Afonso, Pedro Veiga, "Protecting the DNS Infrastructure of a Top Level Domain: Real-Time monitoring with Network Sensors", WSNS 2008, 4<sup>th</sup> IEEE – International Workshop on Wireless and Sensor Networks Security, Atlanta, USA, 29 September – 2 October 2008.

# Enhanced CakES representing Safety Analysis results of Embedded Systems

Yasmin I. Al-Zokari, Daniel Schneider, Dirk Zeckzer, Liliana Guzman, Yarden Livnat, Hans Hagen  
Kaiserslautern University  
Computer Graphics and HCI, Software Engineering: Processes and Measurement  
Kaiserslautern, Germany

Email: (alzokari,zeckzer,guzman,hagen)@informatik.uni-kl.de,danielschneider84@gmail.com,yarden@sci.utah.edu

**Abstract**—Nowadays, embedded systems are widely used. It is extremely difficult to analyze safety issues in embedded systems, to relate the safety analysis results to the actual parts, and to identify these parts in the system. Further, it is very challenging to compare the system's safety development and the different safety metrics to find their most critical combinations. Due to these fundamental problems, a large amount of time and effort is spent for analyzing the data and for searching for important information. Until now, there is a lack of visualization metaphors supporting the efficient analysis of safety issues in embedded systems. Therefore we present “Enhanced CakES”, a system that combines and links the existing knowledge of the safety analysis and the engineering domain and improves the communication between engineers of these domains. The engineers can directly explore the most safety critical parts, retaining an overview of all critical aspects in the actual model. A formal empirical evaluation was performed and showed the increase of accuracy from ESSaRel 28.7% to 83% for CakES .

**Index Terms**—Safety analysis, fault tree analysis, minimal cutset, embedded systems, visualization.

## I. INTRODUCTION

THE COMPLEXITY of embedded systems is currently a major problem. Cars, trains, airplanes, etc. contain an increasing number of these systems. Their safety is one very important aspect. Interactive graphical representations of the data can significantly help to ease the analysis, exploration, and fast comprehension of this complex information. We present “Enhanced CakES” (Enhanced Cake metaphor for safety analysis of Embedded Systems), a visualization system to solve these problems. It provides a new research direction combining the system engineering and the safety analysis domains. This approach comprises different types of data that are aggregated, visualized, and interacted with. We extract and visualize the most important features of the fault tree analysis of an embedded system and display them together with the parts of the physical model. For our approach, a multi-application framework was developed that enables us to combine various applications and their views for different environments (standard monitors and tiled walls [15]).

We performed an empirical evaluation comparing Enhanced CakES, our new safety analysis tool, to ESSaRel (Embedded Systems Safety and Reliability Analyser) [10], the standard tool for safety analysis using component fault tree analysis. Further, we used questionnaires and asked the users to think

aloud to obtain additional information. The evaluation was performed on real data from the embedded systems domain (robotics). We found that our method had a huge positive impact on the participants. The accuracy increased significantly from 28.7% for ESSaRel to 83.1% for Enhanced CakES, while the time for searching and exploring the data increased only slightly from 24.6 minutes for ESSaRel to 29.8 minutes for CakES (not statistically significant  $p = 0.4875$ ). Additionally, we compared the “standard monitors” and the “tiled wall” environments and found that the participants preferred the standard monitors for their work.

The paper is structured as follows: Section II provides the problem statement, including definitions, tasks, measurements, and related work. Our approach, the visual metaphors and the interaction, is presented in Section III, where we illustrate its usage using a real world data set. The empirical evaluation of our approach is presented in Section IV. The future work is presented in Section V. We close this paper with conclusions in Section VI.

## II. PROBLEM STATEMENT AND RELATED WORK

*a) Definitions:* *Fault Tree Analysis* (FTA) is a widely used analytical technique in all fields of safety [5], p. 9, 54. According to [16], “The *Fault Tree* (FT) itself is a graphic model of the various parallel and sequential combinations of faults that will result in the occurrence of the predefined undesired event”. An event is any proposition that is true with a certain failure probability (FP) [12]. *Basic Events* (BEs) are the lowest-level influence factors in the FT and they are represented as the leaves. One of the methods to support the system design, is to provide all possible smallest combinations (*Minimal CutSet* (MCS)) of failures (BE) that lead to a *top level event* (TLE, the undesirable event) [5], p. 58. The authors of [20], [17] point to the importance of MCSs. Almost all FTA tools and methods can be used to generate the MCSs of the system being analyzed.

*b) Tasks:* There are some tasks to be fulfilled by the analysts and the engineers (in our case: the robotics engineers) to improve the safety of any system for a specified cost (e.g., time, effort, money). These tasks—regardless to the tool/method—are as follows: 1-Build FTs, to analyze the safety

of the system (performed by both analysts and engineers). 2-From the FTs, compute the MCSs (safety analysts). 3-Find the most critical MCSs, to start the analysis while spending as little time and effort as possible (safety analysts). 4-Find the BEs of these MCSs (safety analysts). 5-Therefore, there will be a need to have the relations between the safety data (e.g., BEs' ID) and the engineering data (e.g., actual parts in the model). 6-Improve/replace the BEs or add redundancy (decisions made by analysts and engineers). 7-Update the FT accordingly (This is done by exchanging a considerable amount of knowledge between the analysts and the engineers).

c) *Measurements*: The analysts consider one or more measurements for the MCS criticality. Most commonly considered are: the MCSs with the highest *Failure Probability* (FP) or with the highest *Failure Rate* (FR) and the MCSs with the smallest order. The order is also called size, which is the smallest number of BEs in it. For example, single points of failures are very critical.

In addition to the MCSs measurements there are some BEs measurements that needs to be considered, e.g., its id, name, location, shape, FP, and/or the number of occurrence of BEs in the system.

d) *Problem Statement and Related Work*: Visualization of safety data in embedded systems is an emerging research topic. With the growth of embedded systems and fault tree analysis data, visualization is becoming an important method to ease and speedup the developer's analysis. From the IBM & Industry Studies, Customer Interviews it was found that 30% of people's time is spent for searching for relevant information [23]. Moreover, as this exploration depends on humans, this leads to an increasingly high amount of human errors, specially when the data being analyzed is presented in a simple way (e.g., text) and this increases when the amount of the data. When the number of BEs increases, the effort and time needed to analyze the FT information increases significantly, especially when the FT is complex.

In [2], tools and methods related to FTA were examined to assess their support of the tasks outlined above. It was found that almost all tools and methods provide the MCSs' information—if provided—in textual formats. This textual representation is usually not complete. To obtain these information, the analysts have to deal with some difficulties. Most safety analysis tools support the analysts powerfully in the tasks 1 and 2. However, while tasks 3-7 are considered to be difficult and time consuming tasks for the analysts, they are not supported very well. For more details please refer to [2].

*ESSaRel* [10], [13], [11] is the standard tool for component fault tree generation used by, e.g., Siemens, that visualizes component fault trees, models, gates, BEs, and others in a 2D representation. We choose it as model for the tools reviewed in [2], because it is a freely available standard tool and because we had experts whom we could consult. Fig. 1 shows a screen shot of the *ESSaRel* tool.

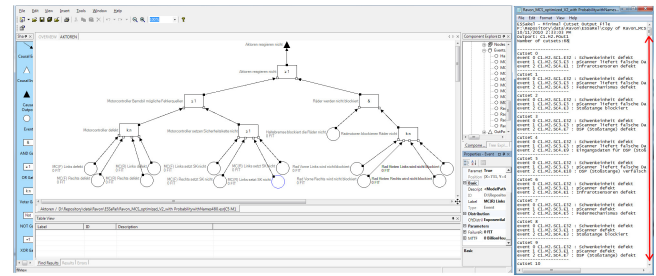


Fig. 1. *ESSaRel*: FT representation and textual results of the FTA (68 MCSs).

### III. OUR APPROACH

In this work, we extend [1], where we focus on the tasks that supports the analysts in a basic way (3-7). This is done by representing the results of the analysis in a different way to ease tasks 3 and 4 without navigating through the FTs or through the text file. Next, we link the data from both domains (safety analysis and engineering) to ease tasks 5-7. Additionally, we added some extra functionalities, such as: -Providing information about the FP distribution over the system. -Ability to trace the temporal safety development of a system. -Providing information about the parts: their shape in 3D and their actual location in the system. -Ability to compare between BEs by their number of occurrence and the quality of the MCSs that contains it. -Supporting users who have color vision deficiency by providing color schemes. -Providing one slider that performs filtering for different levels of MCSs' FP and which adapts its coloring to the selected color scheme. -The system can be applied on different environments such as standard monitors for daily usage and tiled-wall for demonstrations and discussions [15].

The system was developed in close cooperation with the domain experts to reduce the gap of knowledge between safety analysts and embedded systems engineers. This paper presents the enhanced visualization of [1] based on the results and on the feedback of an informal evaluation. Further, it provides many additional utilities:

- Enhancements in the menu:
  - multi-selection of MCSs, BE selection
  - multi range slider
  - added another type of color vision deficiency
  - number of BEs' occurrence in the visualized MCSs
- Enhancements in the MCS view:
  - anti aliasing
  - arranging the MCSs by FP from center
  - visualization of the order of the MCS
  - included the ghost and rotation of the BEs in the MCS when selected to give a 3D non-occluded view
  - automatic zooming towards the selected MCS
  - Three different saturation levels not two, for more distinction
- Enhancements in the BE view:
  - shows the most important BE by FP of the selected MCS



- Enhancements in the interaction:
  - faster interaction, added two different speeds
- performed the evaluations

#### A. Real World Scenario and Setup

We envisioned four safety analysis scenarios based on fault tree analysis data that were created to assess the safety of the embedded system RAVON:

- 1) An engineer wants to find which are the most critical parts in the system to improve them.
- 2) Both analyst and engineer want to discuss which critical parts should be improved by reducing their criticality.
- 3) An engineer and an analyst present the system to managers. They show the improvements by comparing the cake before improving the critical parts with the cake after enhancing the system.
- 4) A safety expert wants to point out the most critical parts to an engineer, i.e., a specialist from the robotics group by showing the most critical BE by both FP and number of occurrence.

Before we introduce our system, we describe RAVON, the safety analysis tool ESSaRel, and the safety data obtained from the analysis.

RAVON is a Robust Autonomous Vehicle for Off-road Navigation [18], [25]. The original model was converted from ProE to open inventor with a size of 162MB. The model is hierarchical and therefore each part can be accessed individually.

The safety of this complex embedded system was analyzed using fault tree analysis. ESSaRel (Section II, Fig. 1) was used to perform the fault tree analysis. We used the results of this analysis, a textual description of the MCSs and the BEs in the FT, as input to our system. This FT contains 540 MCSs and 29 unique BEs. FPs are associated with each BE and each part of RAVON is linked to a BE of the FTA data obtained from this analysis.

#### B. Overview of the System

Our system provides four views: the *MCS View* displays the collection of MCSs and their BEs (safety related data), the *Menu View* displays a menu and additional safety related data, the *Model View* displays the model, and the *BE View* displays the most important BE in the MCS that is selected by the user.

These different views allow the user to have different levels of focus and context at the same time finding the most critical MCSs. The user is directly involved in the data analysis process without the need to navigate through the system's fault trees. We use the first scenario described above to explain our system for single selection and the fourth for multi-selection.

1) *MCS View* Fig. 2: The first step for finding the critical parts is to get an overview over the MCSs (the safety of the system). Therefore, the analyst starts with the MCS view.

In Enhanced CakES, the MCSs are visualized using the cake metaphor. A cake consists of three different levels. Each visible level holds a number of MCSs, which are represented

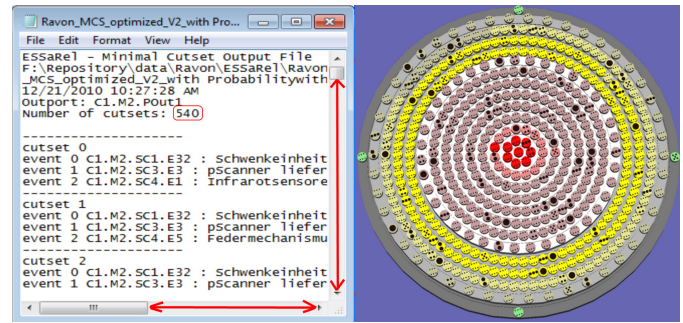


Fig. 2. The MCSs' information. (left) ESSaRel. (right) CakES showing the failure probability distribution of the whole system, the MCSs' failure probability (cylinders color), and their size (number of dots on the cylinders), in 2D. This system is unsafe because it has mostly red cylinders (high failure probability).

as cylinders. Each MCS contains a certain number of BEs and has a specific FP. Fig. 2 (right) shows the MCSs FP and size in the MCS view.

Each *level* (also called *holder*) is represented by a cylinder. The MCSs that are placed on each level are determined according to their FP. The FP of an MCS is computed as the product of the FPs of its BEs. However, any other function could be also used to calculate the FP of MCS. There are four FP values that influence the placement of the MCSs and the number of levels displayed: minimum, border between lowest and middle risk acceptance range (lower border), border between middle and highest (upper border), and maximum. The user can change them using the multi-thumb slider in the menu (Fig. 3 B).

The innermost cylinder (the first level) corresponds to the highest range of FPs, those between the upper border and the maximum. It is the upper part of the cake and includes the most important MCSs. The middle cylinder (the second level) corresponds to the middle range between lower and upper border. The outermost cylinder (the third level) corresponds to the range between the minimum and the lower border.

Three different saturation levels are available. Therefore nine different levels of FP are provided, which can be seen in one blink. A high saturation is assigned to MCSs having the highest FP, a medium saturation level is assigned to MCSs having a high FP, and a low saturation level is assigned to MCSs having a low FP in each specific holder. Fig. 2 (right) shows the MCSs of the first and the second holder (red, yellow MCSs) having three different saturation values, whereas the MCSs in the third holder (green MCSs) all have the same saturation. The order of each MCS is presented as dots [3], importance is reflected by the size of the dots. The most important MCSs are the ones with smallest order (e.g., single BE). Usually, most analysts examine MCSs with order in the range of [1-6], because the larger the number of BEs in a MCS the less critical it is. Therefore, we represented six BEs as a maximum of any MCS containment. The dots visibility can be set or unset by the user. The parts are sized according to the space available inside the MCS.

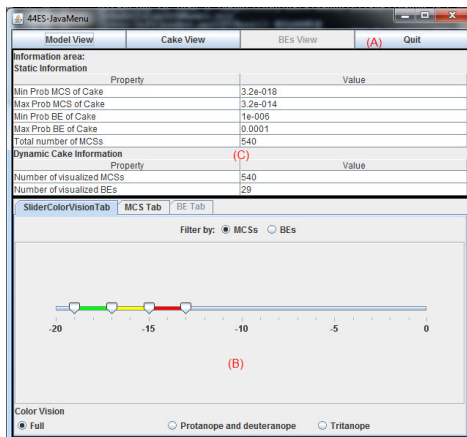


Fig. 3. Information and interaction (FP range slider and color vision deficiency radio buttons) area in the menu.

As 10% of males and 1% of females have color vision deficiency [22], [24], we included two more coloring schemes for different types of color vision deficiency (Protanopia, Deuteranopia, and Tritanopia) in addition to the normal coloring scheme [14]. We used the “Vischeck” tool [9], [21] for simulating the color vision deficiency types and assessing the quality of our choice. When the color schemes are changed, the colors of the multi-thumb slider in the menu also change.

2) *Menu View (Fig. 3)*: The menu view is primarily used for interaction. There are three interaction areas in the menu.

The first area consists of four buttons (Fig. 3 A). Three of them can be used to select the views in focus that the user would like to interact with (MCS, BE, and Model view), one, if pressed, terminates the application.

The second area consists of a multi-thumb slider for setting the FP ranges of each levels (Fig. 3 B, top) and three radio buttons (Fig. 3 B, bottom) for choosing the color scheme. The multi-thumb slider is also used as a filter determining the minimum, lower border, upper border, and maximum FPs of the holders (Section III-B1). We adapted it from [19].

MCSs are assigned to different holders when the border FPs are changed. A logarithmic scale is used for the slider. The default values of the minimum and maximum thumbs are provided directly when loading the data, by taking the minimum and maximum FPs of the MCSs in the data. The second function of the menu view is to display additional information (Fig. 3 C). On the one hand, static quantitative information for the data set is provided (Fig. 3 C, top). The static quantitative information includes the total number of MCSs and the minimum and maximum FPs of both the MCSs and the BEs of the system, while the dynamic information shows the visualized number of MCSs and BEs in the chosen FP ranges. This information changes whenever the user manipulates the thumbs of the slider. On the other hand, dynamic quantitative and qualitative information about selected elements is displayed in the MCS and the BE tab of the menu (Fig. 4 and Fig. 5).

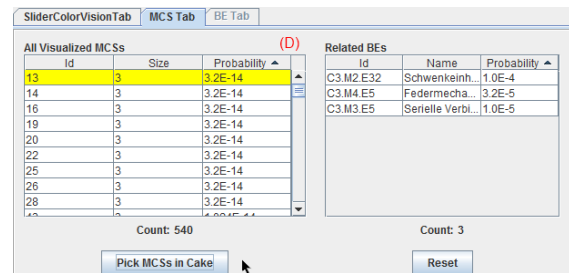


Fig. 4. MCS area in the menu, view after picking an MCS.

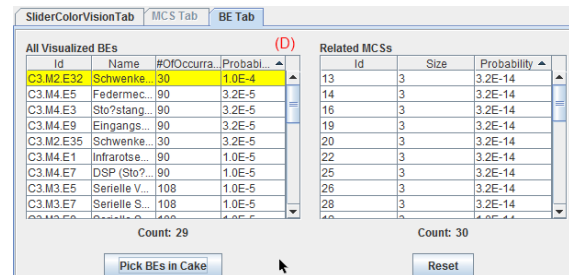


Fig. 5. BE area in the menu, view after selecting a BE which leads to multi-selection of MCSs.

### C. Single Selection

Now, let's get back to our first scenario. The user can identify the most critical MCSs in the MCS view of CakES. Those are the ones in the innermost holder having the highest saturation. If the user suffers from one of the color vision deficiency types, he can change the color scheme to one adapted to his color vision. From the overview, he also sees the distribution of the MCSs criticality. So, he directly gets an insight about the criticality of the system and can compare it with the previous state of the system if available. If he wants to further investigate an MCS, he selects it either in the MCS view or from the menu view. Then, the entry of the selected MCS (its ID, its size, and its FP) is highlighted in the MCS area of the menu (Fig. 4) and information (ID, Name, FP) about its BEs are shown.

Further, this leads to the following automatic changes in the views. First, in the MCS view, the selected MCS becomes translucent and its BEs become visible inside it. These BEs rotate to show the user the 3D shapes of the physical parts related to its BEs. Fig. 6 (left) shows the effect. Until now, the BEs of an MCS are only related to hardware parts of the system. The physical parts related to the BEs of the fault tree analysis are visualized inside each MCS to give the user the relationship between the MCS view, the BE view, and the model view. The positioning of the BEs in the MCSs depends on the number of the BEs in an MCS.

Second, the system displayed in the model view changes to a translucent model with the physical parts related to the BEs of the selected MCS being opaque. This shows the user the parts, the shape of the parts, and additionally their location in the system as shown in Fig. 6 (right). Third, the

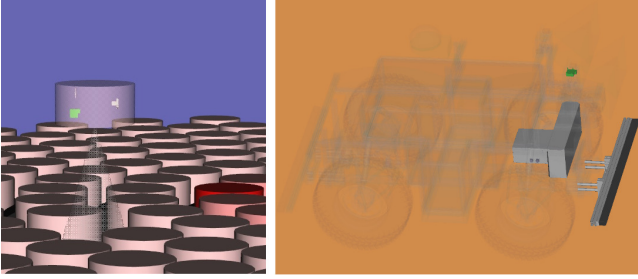


Fig. 6. (left) The MCS view, MCS raising (ghost effect). (right) Rotated view of the model view in 3D.

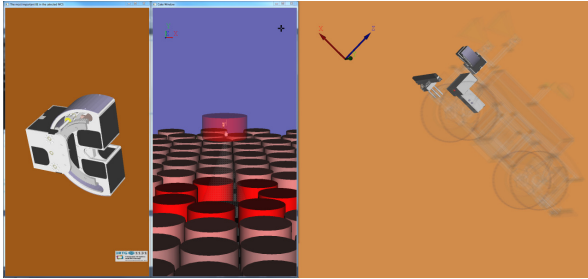


Fig. 7. After selecting an MCS.

BE view visualizes the most important BE in the selected MCS (Fig. 7, left) and facilitates the detailed examination of each BE.

Now, the engineer explores and understands the relation between the BEs of the MCS in the model. Thus, he can identify the parts that are most critical in the system and that should be improved with respect to safety.

The MCS view shows the system's criticality and additionally provides the ability for temporal comparison of the system's evolution. Fig. 8 shows how the system is developing after enhancements.

#### D. Multi Selection

For our fourth scenario, the analyst explores the BEs in the BE area of the menu. The BE's ID, name, FP, and number of occurrence (the number of the MCSs influenced by this BE) are shown. When he selects an interesting BE (Fig. 5), the BE entry is highlighted and all visualized MCSs containing it are listed together with their information (ID, size, FP). At the same time, these MCSs are highlighted in the MCS view making the selected MCSs distinguishable (Fig. 9 and Fig. 10). In this view, he can immediately see the distribution of the MCSs over the different FP levels (critical, tolerable, negligible) without searching the list and thus an overview over the system's criticality is provided that eases comparison between different BEs as shown in Fig. 10. Assume, that there are two different BEs having the same number of occurrence, namely 30. The result of selecting these BEs is shown in Fig. 10: the set of MCSs in the left image is more critical than the set of MCSs in the right image, which means that the first BE (left image) influences the safety of the system more negatively than the second (right image) and should be considered for

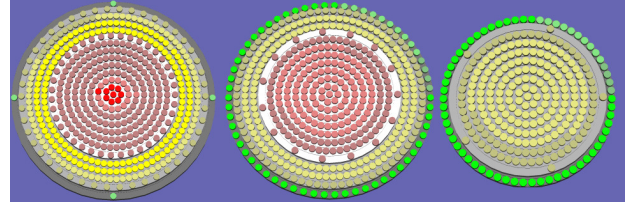


Fig. 8. The safety of the system is increasing during development (MCS view).



Fig. 9. Multiple MCSs show their BEs.

improvement. Fig. 11 shows the BE view, the MCS view, and the model view after selecting a BE.

#### E. Visualization Environments

We used two visualization environments: a two monitor system, where one monitor could display the model in mono or stereo view, and a tiled wall. In both environments, we had multiple coordinated views. More details are described in [15], [1].

## IV. EMPIRICAL EVALUATION

We performed a preliminary empirical evaluation for assessing the impact of CakES on the performance of safety analysis for analyzing a given system based on a fault tree model. Since an evaluation requires a reference for comparison, we compare the results between CakES and ESSaRel for tasks that are supported by both tools. Consequently, we define the statistical hypotheses as follows:

- $HT_1: \mu_{T,CakES} \neq \mu_{T,ESSaRel}$
- $HT_0: \mu_{T,CakES} = \mu_{T,ESSaRel}$

with  $\mu_T$  = mean time required for performing a set of tasks

- $HAcc_1: ACC_{CakES} > ACC_{ESSaRel}$
- $HAcc_0: ACC_{CakES} \leq ACC_{ESSaRel}$

with  $Acc$  = accuracy

Additionally, we evaluated the usability and the usefulness of both tools for supporting the safety analysis of embedded system. In this context, usability means "degree to which a person believes that using a particular system would be free of effort" and usefulness means "the degree to which a person believes that using a particular system would enhance his or her job performance" [7]. We measured usability and



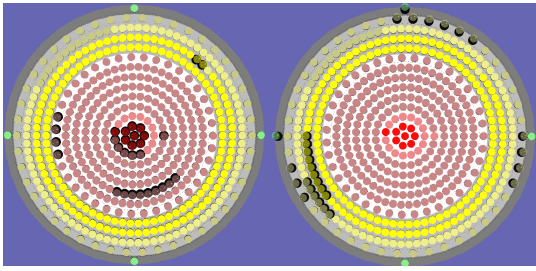


Fig. 10. Selecting a BE which causes selecting multiple MCSs.

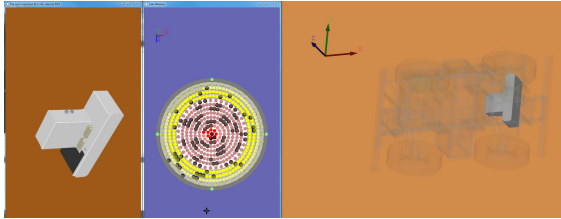


Fig. 11. After selecting a BE (multi-selection of MCSs).

usefulness using a questionnaire with closed questions and statements using a 5-point likert-scale.

#### A. Design

We conducted an experiment with non-probability sampling and one control group according to the recommendations of [6]. We used a convenient sample including software engineers from the Software Research Groups Dependability and Process and Measurement of the University of Kaiserslautern and from the Fraunhofer Institute for Experimental Software Engineering. All subjects were assigned randomly to the experimental group (CakES) and the control group (ESSaRel).

The experimental treatment was applied separately to each subject in the presence of two researchers, one moderator and one observer. During the experiment, the observer registered all questions and comments of the subject. The experimental treatment includes the following steps:

*a) Training:* First, the moderator briefly introduced the purpose of the experiment and the confidentiality and the anonymity of the responses. Then, a structured questionnaire was used for eliciting demographic information including subject age, gender, profession, and experience in safety analysis. Additionally, we conducted a color deficiency test according to [9], [14], [8], [4], [21].

Finally, each subject received the corresponding tutorial of either CakES or ESSaRel. The tutorials were prepared by an expert of each tool. Additionally, the moderator provided the corresponding tool and a data set example to give the subject the opportunity to explore the corresponding tool. The moderator instructed the subject to comment any doubt, uncertainty, or difficulty she or he had about the use of the corresponding tool. All questions of the subject were resolved. We registered the time that each subject required for the training.

*b) Safety analysis:* After the training, the moderator loaded the data set used for the evaluation. For that purpose, we selected a real problem in the robotics domain. The corresponding system includes 540 MCSs and 29 distinct BEs. The moderator also gave the subject a list of tasks to be accomplished using the corresponding tool without time limit. The tasks to be performed by the subject are based on the task list presented in Section II (3-5): 1. Determine how many MCSs should be improved. 2. Provide the identification of the MCSs you want to explore. 3. Give the failure probabilities of these MCSs. 4. Provide the identification, failure probability, and name of the BEs that could cause a failure to the system of each MCS of the previous question. 5. If the subject used CakES: describe the location of the objects associated to the BEs in the model. 6. Judge, if the system you analyzed is safe or critical and if it is worth analyzing and spending time and effort on.

The accuracy of the tasks described above was measured against the baseline defined by a safety expert. For the first and second task, we considered the rate of correctly identified MCSs. Defining  $M_C$  as the set of correct MCSs according to the expert judgment and  $M_F$  the set of identified MCSs by the user, we determined the accuracy as  $\frac{|M_F \cap M_C|}{|M_C|}$ . For the third task, accuracy took a binary value (1: correct; 0: incorrect) considering the value computed by 3 safety experts. For the fourth and fifth task, we considered the rate of correctly identified BEs. Specifying  $B_C$  as the set of correct BEs  $B_C = \{BE | BE \in MCS, MCS \in M_C\}$  according to the safety expert and  $B_F$  as the set of identified BEs by the user ( $B_F$ ), the accuracy was determined as:  $\frac{|B_F \cap B_C|}{|B_C|}$ . For the last tasks we also used a binary value.

*c) Evaluation of usability and usefulness:* A structured questionnaire with closed questions and statements using a 5-point Likert-scale was used for eliciting the impressions of the subjects regarding the usability and usefulness of the corresponding tool. According to [7], we refined usability into: easy to understand, easy to learn, and the aesthetic value of the tool. We also decomposed usefulness into: allows to work faster, increase productivity, and the subjects confidence in his or her results.

It is important to remark that we performed 3 pilot tests for reviewing the experimental instruments (i.e., training material, problem definition, instructions, templates, and questionnaires) and the experimental treatments. The goals of the pilot tests were to identify on time possible confounding variables and important omissions and to prevent misunderstandings and mistakes regarding the experimental instruments. The pilot tests were conducted with one software engineer, one safety expert, and one robotic engineering expert.

#### B. Results

*d) Sample:* The experiment took place during August 2010. As we choose people working on a high level in safety related areas, we had only 12 participants, who were split up into the experimental and the control group, i.e., CakES and ESSaRel respectively. The subjects of the CakES group were

between 22 and 33 years old. Considering a seven point Likert-scale (1: extremely low and 7: extremely high), they have on average rather low experience in safety analysis and neutral experience in visualization. Out of six subjects, two had used FTA tools before. The subjects of the ESSaRel group were between 26 and 36 years old, they have on average rather low experience in safety analysis and neutral experience in visualization. Out of six subjects, 3 had used a FTA tools before.

e) *Training*: The training for CakES took on average 34 minutes ( $\sigma = 8.1$ ) and the training for ESSaRel 13.6 minutes ( $\sigma = 9.5$ ). The difference is explained because 3 subjects in the ESSaRel group had worked before with the tool. Therefore, they neither read the tutorial nor used the tool during the training. The training was conducted according to the experimental plan.

f) *Performance in safety analysis*: Whereas all the subjects in the CaKES group completed the assigned tasks and spent on average 35 minutes ( $\sigma = 0.8$ ) on solving those tasks, only 5 subjects of the ESSaRel group completed the assigned tasks and they spent on average 32 minutes ( $\sigma = 5.3$ ). The subject who did not finish the tasks claimed that he or she did not want to explore all views and to compute manually something that should be supported by the tool.

Considering the size of the sample, Lilliefors test shows that the time is normally distributed for both groups. Consequently, we used ANOVA to test  $HT_0$  (i.e.,  $\mu_{T,ESSaRel} = \mu_{T,CakES}$ , with  $\mu_T$  being the mean time to complete all tasks) with a significance level of 0.05. It shows that we can not reject the null hypothesis ( $p = 0.4875$ ). This means that there is no significant difference between the time that the subjects spent in solving the assigned tasks in both tools.

g) *Accuracy*: The results indicate that subjects using CakES achieved more accurate results than using ESSaRel. For the accuracy variables of task 1 to 4, Lilliefors test shows that they are not normally distributed for both groups (significance level = 0.05;  $p < 0.001$ ). Therefore we conducted a Wilcoxon rank sum test for testing the null hypotheses  $H_{Acc_0}$  (i.e.,  $Acc_{CakES} = Acc_{ESSaRel}$ , with  $Acc = accuracy$ ) for tasks 1 to 4. The corresponding results show that  $H_{Acc_0}$  can be rejected with significance level 0.1 for tasks 1 ( $p = 0.06$ ) and 2 ( $p = 0.08$ ) and with significance level 0.05 for tasks 3 ( $p = 0.03$ ) and 4 ( $p = 0.03$ ). So, subjects using CakES were significantly more accurate.

Consequently, the results of this empirical evaluation show that the participants achieved better performance using CakES than ESSaRel. But considering the sample size and composition, these results can not be interpreted as being conclusive. More empirical studies with larger samples, including safety analysts and engineers from several domains are necessary to obtain more reliable conclusions regarding the performance of CakES. Table I shows the descriptive statistics for the accuracy of each task for both tools. \*<sup>1</sup> One participant did not know where is the front or the back of the robot is, so he/she said that the BEs are in the back of the robot, but pointed the correct location by hand. \*<sup>2</sup> One participant did not understand the question, he thought the system means the visualization system

TABLE I  
ACCURACY. TS:TOTAL NUMBER OF CORRECT ANSWERS FROM THE PARTICIPANTS

Task id	Accuracy Statistics	Measure	ESSaRel	CakES
1	$\frac{ M_F \cap M_C }{ M_C }$ , mean, std		0.38, 3.5, 4.03	0.94, 8.5, 1.11
2	$\frac{ M_F \cap M_C }{ M_C }$ , mean, std		0.27, 2.5, 3.09	0.79, 7.16, 2.85
3	$\frac{TS, TS}{ M_C }$ , mean, std		0.20, 1.83, 3.23	0.88, 8, 1.15
4	$\frac{ B_F \cap B_C }{ B_C }$ , mean, std		0.36, 9.83, 9.02	0.92, 25, 4.47
5	$\frac{ B_F \cap B_C }{ B_C }$ , mean, std		Not applicable	0.83, 0.83, 0.37 * <sup>1</sup>
6	1: correct; 0: incorrect, number of correct occurrences		2 out of 6	5 out of 6 * <sup>2</sup>

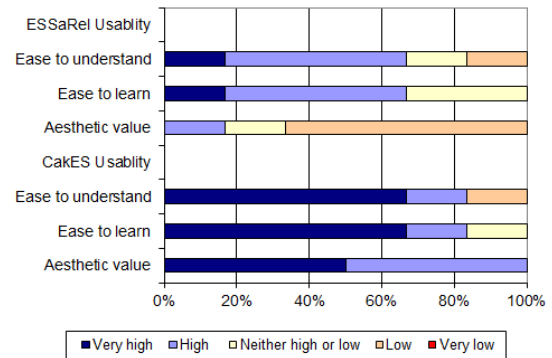


Fig. 12. The usability of ESSaRel and CakES.

(CakES), not the system being analyzed. So, he/she provided his/her positive opinion.

h) *Perception on usability and usefulness*: The preliminary results shows that the participants tend to perceive CakES as being more easy to understand, easy to learn, and with a greater aesthetic value than ESSaRel. Participants tend also to consider CakES more suitable for working faster than ESSaRel. They believe that they increased their productivity and confidence in the produced results more by using CakES. Even though the results provide positive feedback, since we measured usability and usefulness only based on participants perception, it is important to conduct more empirical studies including more objective measures related to usability and usefulness in order to get more reliable results. Fig. 12 and Fig.13 show the usability and the usefulness of both tools.

## V. FUTURE WORK

In the future, we want to test our approach using other examples from other domains. Additionally, we would like to get larger data to assess the scalability of our approach. As mentioned in the evaluation section, more empirical studies should be performed, layering the results separately with respect to the peoples' safety-analysis-experience level. Further, we will allow selecting parts in the model view highlighting all related MCSs in the MCS view and in the Menu. Adding software will be the next major task to achieve. Finally, adding labels on the parts shown in the model view would be nice to have.

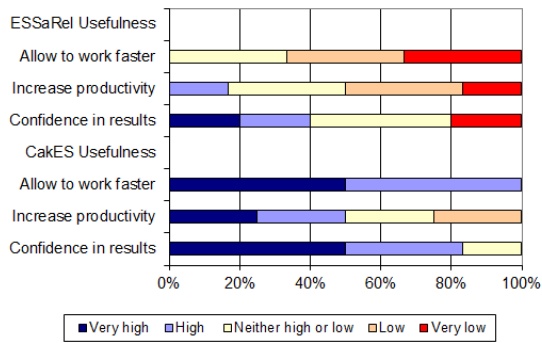


Fig. 13. The usefulness of both ESSaRel and CakES.

## VI. CONCLUSION

CakES is an easy to learn and intuitive to use visual environment providing the most critical factors that are required in the safety analysis domain. It facilitates the exploration of the data and alleviates the comparison and the understanding of the system's safety. It combines safety analysis results and the models of embedded systems enabling the user to directly relate safety issues with the corresponding parts of the embedded system. As it is applicable on different screen configurations, the user can choose the most suitable one for his or her needs.

First of all, Enhanced CakES provides the minimal cutsets, their FPs and size, their basic events, and the overall FP distribution of the system. Further, it relates the basic events to the actual parts (shape) of the system (in 3D) and its location in the system, and provides its information: id, name, and FP. It shows, which is the most critical basic event of a minimal cutset. It provides different color schemes and it works in different visualization environments. It is interactive and provides 3D stereoscopic views of the model and is also applicable on tiled-wall environments, which could be used for users representations and discussions [15]. In addition to visualizing the results of fault tree analysis, Enhanced CakES also allows to easily compare the criticality of different systems or different versions of the same system. Finally, after gaining experience in either domain the expert can work efficiently alone exploring and judging how to choose the parts to improve and the parts to replace depending on their safety and importance. Even though we performed the evaluation only on the single selection mechanism, the CakES performed significantly better than the standard tool.

## VII. ACKNOWLEDGMENTS

We would like to thank our colleagues at the TU Kaiserslautern, Martin Proetsch Lisa Kiekbusch, Zhensheng Guo, and all the participants in our evaluation. This project was partially supported by the DAAD, the BMBF project ViERforES, and the IRTG 1131.

## REFERENCES

- [1] Y. Al-Zokari, T. Khan, D. Schneider, D. Zeckzer, and H. Hagen. CakES: Cake Metaphor for Analyzing Safety Issues of Embedded Systems. In

- H. Hagen, editor, *Scientific Visualization: Advanced Concepts*, volume 2 of *Dagstuhl Follow-Ups*, Wadern, Germany, 2010. Schloss Dagstuhl–Leibniz Center for Informatics.
- [2] Y. I. Al-Zokari, Y. Yang, D. Zeckzer, P. Dannenmann, and H. Hagen. Towards Advanced Visualization and Interaction Techniques for Fault Tree Analyses, Comparing existing methods and tools, 2011. to be submitted to proceedings of the IRTG 2011.
- [3] Y. I. Al-Zokari, D. Zeckzer, and H. Hagen. Safety-Domino Representing Criticality of Embedded Systems. In *EuroVis 2011, Bergen, Norway, Eurographics / IEEE Symposium on Visualization 2011, poster proceedings*, page 21, 2011.
- [4] Archimedes' Lab. Color Blindness or Color Vision Deficiency, 2010. <http://www.archimedes-lab.org/colorblindnesstest.html>; Online; accessed 31-March-2011.
- [5] M. Bozzano and A. Villaforita. *Design and Safety Assessment of Critical Systems*. CRC Press (Taylor and Francis), an Auerbach Book, 2010.
- [6] J. Creswell. *Research design: qualitative, quantitative, and mixed methods approaches*. Sage Publications, 2009.
- [7] F. D. Davis. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340, 1989.
- [8] S. Deeb and A. Motulsky. Red-Green Color Vision Defects, 2005. <http://www.ncbi.nlm.nih.gov/bookshelf/br.fcgi?book=gene&part=rgcb>; Online; accessed 31-Mar-2011.
- [9] B. Dougherty and A. Wade. Vischeck simulates colorblind vision, 2008. <http://www.vischeck.com/>; Online; accessed 31-March-2011.
- [10] ESSaRel. Background information — ESSaRel, 2002. <http://www.essarel.de/index.php?site=backgroundtext>; Online; accessed 31-March-2011.
- [11] B. Kaiser, C. Gramlich, and M. Förster. *Computer Safety, Reliability, and Security*, volume 3219/2004, chapter State-Event-Fault-Trees - A Safety Analysis Model for Software Controlled Systems, pages 195–209. Springer Berlin, Heidelberg, 2004. <http://www.springerlink.com/content/j886uwajl8tnu9y6>.
- [12] B. Kaiser, C. Gramlich, and M. Förster. State/event fault trees - A Safety Analysis model for software-controlled systems. *Reliability engineering & systems safety*, 92:1521–1537, 2007.
- [13] B. Kaiser, P. Liggesmeyer, and O. Mäkel. A new component concept for fault trees. In *Proceedings of the 8th Australian Workshop on Safety Critical Systems and Software (SCS'03), Adelaide*, pages 37–46, 2003.
- [14] M. Kalloniatis and C. Luu. *Psychophysics of Vision: The Perception of Color*, 2007.
- [15] T. Khan, D. Schneider, Y. Al-Zokari, D. Zeckzer, and H. Hagen. Framework for Comprehensive Size and Resolution Utilization of Arbitrary Displays. In H. Hagen, editor, *Scientific Visualization: Advanced Concepts*, Dagstuhl Follow-Ups, Wadern, Germany, 2010. Schloss Dagstuhl–Leibniz Center for Informatics.
- [16] NASA Office of Safety and Mission Assurance. *Fault Tree Handbook with Aerospace Applications*. Technical report, NASA Headquarters, Washington, DC, USA, August 2002.
- [17] F. Ortmeier, W. Reif, and G. Schellhorn. Formal safety analysis of a radiobased railroad crossing using deductive cause-consequence analysis (DCCA). In *Proceedings of 5th European Dependable Computing Conference EDCC, volume 3463 of LNCS*. Springer, 2005.
- [18] RAVON. AG Robotersysteme: Ravon, 2009. <http://agrosy.informatik.uni-kl.de/en/robots/ravon/>; Online; accessed 31-Mar-2011.
- [19] Swing. JQuery Examples 1. [http://en.pudn.com/downloads3/sourcecode/java/detail6037\\_en.html](http://en.pudn.com/downloads3/sourcecode/java/detail6037_en.html); Online; accessed 31-Mar-2011.
- [20] A. Thums and G. Schellhorn. Formal safety analysis in transportation control. In *Proceedings of the Workshop on Software Specification of Safety Relevant Transportation Control Tasks*, VDI Verlag GmbH, 2002.
- [21] B. Wandell, B. Dougherty, and A. Wade. Try Vischeck on Your Image Files. <http://vischeck.com/vischeck/vischeckImage.php>; Online; accessed 31-Mar-2011.
- [22] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers Inc., 2004.
- [23] M. Weber. A survey of Semantic Annotations for Knowledge Management, 2008. <http://www.mendeley.com/profiles/markus-weber/>; Online; accessed 31-Mar-2011.
- [24] Wikipedia. Color blindness. [http://en.wikipedia.org/wiki/Color\\_blindness](http://en.wikipedia.org/wiki/Color_blindness); Online; accessed 31-Mar-2011.
- [25] Wikipedia. Ravon — Wikipedia, The Free Encyclopedia, 2009. <http://en.wikipedia.org/wiki/Ravon>; Online; accessed 31-Mar-2011.

# Integrated management of risk information

José Barateiro  
INESC-ID, LNEC

Rua Alves Redol 9, 1000-029, Lisboa, Portugal  
Email: jose.barateiro@ist.utl.pt

José Borbinha  
INESC-ID

Rua Alves Redol 9, 1000-029, Lisboa, Portugal  
Email: jlb@ist.utl.pt

**Abstract**—Today’s competitive environment requires effective risk management activities to create prevention and control mechanisms to address the risks attached to specific activities and valuable assets. One of the main challenges in this area is concerned with the analysis and modeling of risks, which increases with the fact that current efforts tend to operate in silos with narrowly focused, functionally driven, and disjointed activities. This leads to a fragmented view of risks, where each activity uses its own language, customs and metrics. The lack of interconnection and holistic view of risks limits an organization-wide perception of risks, where interdependent risks are not anticipated, controlled or managed. In order to address the Risk Management interoperability and standardization issues, this paper proposes an alignment between Risk Management, Governance and Enterprise Architecture activities, providing a systematic support to map and trace identified risks to enterprise artifacts modeled within the Enterprise Architecture, supporting the overall strategy and governance of any organization. We propose an architecture where risks are defined through a XML-based domain specific language, and integrated with a Metadata Registry to handle risk concerns in the overall organization environment.

## I. INTRODUCTION

**R**ISK always exists, whether or not it is detected or recognized by an organization. Several areas involve risks that should be treated to provide significant benefits to an organization, like business risks, market risks, credit risks, operational risks, IT risks, engineering, etc. Thus, risk strategies vary from generic approaches, project management, IT (including information security), safety engineering, etc.

Depending on the knowledge area, several definitions of risk can be found in the literature. For instance, in [1] risk is defined as: “An undesirable outcome that poses a threat to the achievement of some objective. A process risk threatens the schedule or cost of a process; a product risk is a risk that may mean that some of the system requirements may not be achieved.” Similarly, the ISO Guide 73:2009 [2] defines risk as: “...the combination of the probability of an event (threat<sup>1</sup>) and its consequences when exploiting any vulnerability<sup>2</sup>”.

Risk Management (RM) is a continuously developing arena whose ultimate goal is to define prevention and control mechanisms to address the risks attached to specific activities and

valuable assets. The early identification of potential problems allows the creation of plans to reduce their potential adverse impact [3]. A RM process describes a set of systematic activities to support the proactive identification and mitigation of risks within a specific environment.

In this paper, we consider that a risk exists when a threat with the potential to cause loss or harm occurs and is able to exploit a vulnerability/weakness associated with an asset that has a value to be protected. The type of assets depends on the nature of the organization, but might include physical entities (e.g., person, office), information entities and processes. When the vulnerability is exploited, it causes an impact on the achievement of the organization objectives. The goal of RM is to manage risks by defining a set of adequate controls to block threats, eliminate vulnerabilities or reduce the impact of the risk occurrence.

Analyzing and modeling risks is one of the most critical tasks in the overall process of RM. Traditional approaches, such as Fault Tree Analysis, Event Tree Analysis, Failure Mode Effect and Criticality Analysis are commonly used to model risks in the safety community [4], [5]. However, these approaches are not suitable to address the imminent risks that today’s organizations face at multiple dimensions (both internally and externally).

Several models have been proposed to address risks at the organizational level, integrating the different views of the related stakeholders, such as the COSO Enterprise RM framework (see Section II), KAOS [6], GBRM [7] and the Tropos Goal Risk Model [8]. Risks at the organizational level are covered by Enterprise Risk Management (ERM), which provides a framework to manage the uncertainty and the associated risks and opportunities in the global scope of an organization. Thus, ERM should be seen as an enabler to the organizations, being impossible to operate on silos. In fact, ERM is part of the corporate governance, providing risk information to the board of directors and audit committees. It is also related to the performance management by providing risk adjustment metrics, with internal control, and with external audit firms. This increases the requirement to be able to exchange risk information, supporting the interoperability of risk information.

It is currently recognized that RM activities must be aligned with the business processes of the organization [9]. When organization business processes and strategic planning are aligned with proactive RM activities, a well-defined path and

<sup>1</sup>Threat is any circumstance or event with the potential to adversely impact an asset through unauthorized access, destruction, disclosure, modification of data, and/or denial of service [2].

<sup>2</sup>Vulnerability is the existence of a weakness, design, or implementation error that can lead to an unexpected, undesirable event compromising the security of the computer system, network, application, or protocol involved [2].



strategy to attain business value is achieved. However, no known business processes have the capability to formally define the sources and dependencies of risks [10]. Moreover, obtaining value through risk assessment can only be achieved through appropriate reporting and communication mechanisms. Due to a complete view of organization's risks, overall risk information becomes visible to executives and management boards, making it possible to incorporate this information to strategic and operational planning.

In fact, one of the main problems of RM is the fact that several efforts operate in silos with narrowly focused, functionally driven, and disjointed activities [9]. This leads to a fragmented view of risks, each using their own language, customs and metrics. The lack of interconnection and unified view of risks hampers an organization-wide view of risks, where interdependent risks are not anticipated, controlled or managed. On the other hand, there is an increasing requirement to exchange risk and control information between organizations and external audit firms. Mapping risk and control information, both internally and to external organizations is highly expensive and inefficient. The lack of interoperability mechanisms between applications used to support different techniques also impedes the analysis of interrelated risks.

This paper proposes an alignment between RM, Governance and Enterprise Architecture (EA) activities, in order to provide a systematic support to map and trace identified risks to artifacts modeled within an EA, supporting the overall strategy of any organization. We formalize the risk management concepts and propose an architecture to manage risk information in an integrated way. This architecture is built on top of three main ideas: (i) risks should be mapped into EA artifacts to support an organization-wide view of risks (from the multiple viewpoints defined in the EA), better assess the spread of a risk from a systematic analysis of the EA related components, and improving the monitoring of risks, using the monitoring activities and tracking of changes in the EA; (ii) the risks models should be decoupled from the EA representation in order to not depend on a specific representation (e.g., if we propose an extension to a specific notation to include risk information, we would be limited to scenarios where this notation is used); and (iii) risk information should be represented in a format that simplifies the interoperability and exchange of information to both internal and external stakeholders or systems.

The remainder of this paper is organized as follows. First, in Section II we describe the related work in the areas of IT Governance, RM and EA. Section III shows the proposed approach to address risks through the EA. Section IV formalizes the risk management concepts, while Section V details the architecture view for the management of risk information. Finally, Section VI presents the main conclusions of this work.

## II. RELATED WORK

### A. Risk Management

RM frameworks are especially concerned with the definition of a set of principles and foundations to guide the design and

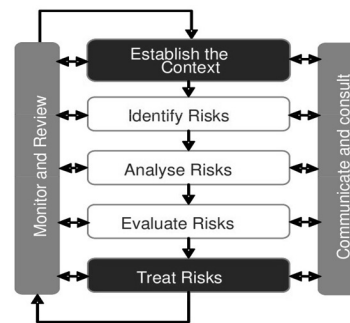


Fig. 1. Risk Management Process

implementation of RM processes in any type of organization. Since they are not focused on any specific area of implementation, it is not possible to find any recommendation about adequate methods to execute within the RM process or even a previous knowledge base with common risks and suitable treatment plans for the identified risks.

The ISO 31000:2009 RM standard [11] is based on the principle that RM is a process operating at different levels, as shown in Figure 1. The RM process is characterized by the combination of policies and procedures applied to the activities of establishing the context, assessing (identifying, analyzing and evaluating), treating, communicating, consulting, monitoring and reviewing the risks.

First, defining the context is crucial to identify strategic objectives and define criteria (both internal and external parameters) to determine which consequences are acceptable to this specific context. Second, today's organizations are continuously exposed to several threats and vulnerabilities that may affect their normal behavior. The identification recognizes the existence of risks; the analysis examines the nature and severity of the identified risks; and the evaluation compares the severity of risks with the defined risk criteria, to decide if the risks are acceptable, tolerable or define the appropriate techniques/controls to handle them.

The identification of threats, vulnerabilities and risks is based on events that may affect the achievement of the goals identified in the first phase. After that, the risk analysis and evaluation estimates the likelihood and impact of risks to the strategic goals, in order to be able to decide on the appropriate techniques to handle these risks (Treat Risks).

The RM process requires a continuous monitor and review activity to audit the behavior of the whole environment allowing, for instance, the identification of changes in risks, or the suitability of implemented risk treatment procedures and activities. Finally, the communication and consultation activities are crucial to engage and dialog with stakeholders.

Enterprise Risk Management (ERM) is the process of identifying and analyzing risks, from an integrated and organization-wide perspective [12].

The Committee of Sponsoring Organizations of the Treadway Commission (COSO) view of ERM is that "Every entity exists to provide value for its stakeholders" [13]. In fact, all entities can face several types of uncertainty, raising a

challenge to the management on how to deal with such uncertainty in a way that maximizes the values of those entities for the interested stakeholders.

In 2004, COSO issued the COSO ERM Framework [13] to provide a common accepted model for evaluating and aligning effective enterprise-wide approaches to RM. This framework defines essential ERM components; discusses key ERM principles and concepts, and suggests a common ERM language.

The COSO ERM Framework analyzes ERM from three different dimensions: Objectives, Organization (and organization units) and components of ERM. Within the context of an organization vision, management establishes objectives for several levels. The COSO ERM framework organizes objectives in four categories:

- Strategic: high-level goals to support the organization's mission.
- Operations: effective use of the organization operational resources.
- Reporting: reliability of reporting (both for internal and external stakeholders).
- Compliance: compliance with applicable law and regulations.

The proposed categories might overlap, since a specific objective can fall into more than one category, but support the focus on distinct issues of ERM. The organization dimension considers ERM activities at all levels of the organizational architecture (e.g., Organization-level, Division, and Business Unit). Finally, the framework is composed by eight interrelated components:

- Internal Environment - encompasses the tone of an organization, and establishes the basis for how RM is viewed and addressed.
- Objective Setting - the definition of objectives is required to allow the identification of potential events affecting their achievement.
- Event Identification - identification of events that may affect the achievement of objectives. Events that may cause a negative impact represent risks, while events that may have a positive impact represent opportunities.
- Risk Assessment - understand the extent of incidents, analyzing their likelihood and impact. It is used to assess risks and also to measure the related objectives. Assessment can be qualitative or quantitative.
- Risk Response - identifies and evaluates potential responses (avoiding, accepting, reducing or sharing) to risk.
- Control Activities - set of policies and procedures to ensure that risk responses are effectively carried out.
- Information and Communication - relevant information concerning risks is captured and communicated to stakeholders to carry out their responsibilities.
- Monitoring - the effectiveness of other ERM components is monitored through continuous monitoring activities or separate evaluations.

Note that ERM is not a series of independent processes,

but a multidimensional and iterative discipline where each component can influence another.

### B. IT Governance

IT Governance is a key discipline for making effective decisions and communicating the results within IT-supported organizations. Its main purpose is to identify potential managerial and technical problems before they occur, so that actions can be taken to reduce or eliminate the likelihood and/or impact of these problems. *Control Objectives for Information and related Technology (COBIT)* [14] is a set of best practices, measures and processes to assist the management of IT systems. COBIT is not specific to a technological infrastructure nor business area, and intends to fill the gap between requirements, technical issues and risks. It includes a framework, a set of control goals, audit maps, tools to support its implementation and, especially, a guide for IT management. The latter is organized in the domains of (i) Planning and Organization; (ii) Acquisitions and Implementation; (iii) Delivery and Support; and (iv) Monitoring and Evaluation. These processes address the areas of strategic alignment (alignment of IT with the business) [15]; value delivery (creation of business value); resource management (proper management of IT resources); risk management; and performance management.

The "ISO/IEC 27000 series" [16] include a set of standards developed for information security matters. This family of standards specifies the Information Security Management Systems (ISMS) Requirements, proposing a process approach to design, implement, operate, monitor, review, maintain and improve an ISMS. The *design* process follows a risk management approach, including the definition of the risk assessment approach, risk identification, risk analysis, evaluation of risk treatment options and selection of controls to treat risks. The requirements proposed in these standards intend to be generic and applicable to all types of organizations, independent of type, size and nature.

### C. Enterprise Architecture

Architectural descriptions provide rigorous descriptions of complex systems with diverse concerns, and are a recommended approach to tackle the dynamic and increasing complexity of those systems. According to the IEEE Std. 1471-2000, which has also become ISO/IEC 42010:2007, architecture is "the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution" [17]. It considers that a system has a mission and inhabits an environment which influences it. It also has one or more stakeholders that have concerns regarding the system and its mission. Concerns are "those interests that pertain to the system's development, its operation, or any other aspects that are critical or otherwise important to one or more stakeholders".

A system has an architecture described by an architecture description which includes a rationale for the architecture. The architecture description is also related with the stakeholders

Perspective	DATA What	FUNCTION How	NETWORK Where	PEOPLE Who	TIME When	MOTIVATION Why
<b>Planner (Objective/Scope - Contextual)</b>	Things important for the business	Business Processes	Business Locations	Important Organizations	Events	Business Goals and Strategies
<b>Owner (Enterprise Model - Conceptual)</b>	Conceptual Data / Object Model	Business Process Model	Business Logistics System	Workflow Model	Master Schedule	Business Plan
<b>Designer (System Model - Logical)</b>	Logical Data Model	System Architecture Model	Distributed Systems Architecture	Human Interface Architecture	Processing Structure	Business Rule Model
<b>Builder (Technology Model - Physical)</b>	Physical Data/Class Model	Technology Design Architecture	Technology Architecture	Presentation Architecture	Control Structure	Rule Design
<b>Programmer (Detailed Representation - Out of Context)</b>	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Speculation
<b>User (Functioning Enterprise)</b>	Usable Data	Working Definition	Usable Network	Functioning Organization	Implemented Schedule	Working Strategy

Fig. 2. The Zachman framework

of the system and deals with several views according to the viewpoints of the stakeholder. This includes functional and non-functional aspects of stakeholders' concerns.

Accurate architecture descriptions provide a "complete picture" of the overall system. However, any system (especially a complex system made of software, people, technology, data and processes) is continuously subject to changes, usually driven by the evolution of the system environment [18].

Enterprise Architecture is a holistic approach to systems architecture with the purpose of modeling the role of information systems and technology in the organization, aligning enterprise-wide concepts and information systems with business processes and information. It supports planning for sustainable change and provides self-awareness to the organization [19].

The Zachman framework is a "way of defining an enterprise's systems architecture" with the purpose of "giving a holistic view of the enterprise which is being modeled" [20]. It can also be described as a "classification theory about the nature of an enterprise" and the kinds of entities that exist within. As shown in Figure 2, the Zachman framework presents itself as a table where each cell can be related to the set of models, principles, services and standards needed to address the concerns of a specific stakeholder. The rows depict different viewpoints of the organization (Scope, Business, System, Technology, Components, and Instances), and the columns express different perspectives on each of the viewpoints (Data, Function, Network, People, Time, Motivation). Due to its visually appealing nature almost resembling a "periodic table of the elements" of descriptive representations of the organization, it is very useful in analyzing the scope of specific models and frameworks, and in reconciling potentially conflicting viewpoints.

The Open Group Architecture Framework (TOGAF) [21] provides methods and tools to support architecture development. It comprises seven modules which can be partly used independently of each other. The core of TOGAF is the Architecture Development Method (ADM), which consists of a cyclical process that starts with a preliminary phase in which the context, relevant guidelines, standards, and goals are identified, the main process begins with the elaboration of an architecture vision and the principles that should guide the architecture work. This architecture vision phase provides

the basis for developing the business architecture, information systems architecture, and technology architecture. On this basis, solutions are developed (opportunities and solutions phase), and migration and implementation are planned and governed (migration planning and implementation governance phases).

Finally, the architecture change management phase ensures that the architecture continues to be fit for purpose. All of the phases are executed concurrently with a Requirements Management activity, which drives the other phases. The ADM can be adapted for various purposes, and in more complex situations, the architecture can be scoped and partitioned so that several architectures can be developed and later integrated using an instance of the ADM to develop each one of them.

### III. APPROACH

In other to provide a systematic support to the overall strategy of any organization, being able to map and trace identified risks to enterprise artifacts, this paper proposes an alignment between Risk Management (RM), Governance and Enterprise Architecture (EA) activities. Governance processes intend to ensure the comprehensive control when moving from strategic planning to operative implementation. This task demands orientation and transparency that can be supported by the EA processes. In fact, EA can be used to reveal deficiencies, show complex interactions between strategies, business processes, services and infrastructure, providing a foundation for complex analysis (either by Governance or RM activities). We propose an integrated view of Governance, Risk and EA to support organizations to be efficient, effective and reliable. In other words, decision making must be able to do the right things in the right way with a controlled risk.

Organizations can be described in terms of their architecture. The existence of a description of EA artifacts (e.g., data models, business models, strategies, infrastructure plans, hardware, functions, organizational structure, etc) denotes awareness of the organization concerning its architecture. Like in buildings, the architecture always exist, either it is recognized, planned and supported by accurate models, but also in scenarios where EA is not recognized by organizations. When we consider the relation between Governance and EA, EA provides transparent information as a basis for decision making and control activities (Governance). However, this should not be seen as a static relation, since it is also about the continuous provision of updated and accurate information that enables governance, bridging the gap between strategic planning and real operations (strategic alignment).

The interaction between Governance and Risk is already recognized by the broader area of Governance, Risk and Compliance (GRC). In fact, the increasing spread of regulations like Basel II and the Sarbanes-Oxley Act, along with the ultimate series of global economic and financial events, raised the awareness to effectively address the GRC activities of today's organizations [22]. The concepts involved in GRC are not new, but are traditionally addressed as separate concerns inside the organizations. However, these concepts share a set



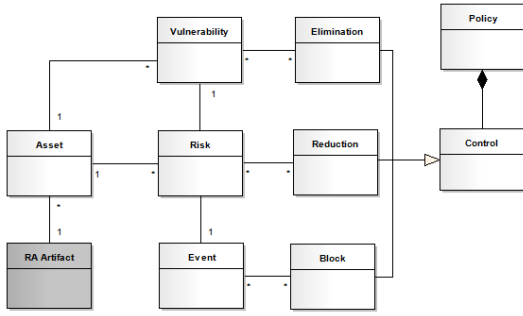


Fig. 3. Domain model of the risk concepts

of knowledge, methodology and processes, which allows an optimal and common view where GRC activities are addressed in an integrated way to improve decision making, strategy setting and performance. This avoids conflicts, overlaps and gaps between the GRC activities.

The main rationale to propose a connection between Risk and Enterprise Architecture is based on the fact that risk activities are usually performed in silos and without a clear mapping between risks and potentially affected organization components. Extending risk activities to map risks to EA components supports the analysis of the spreading of risks that can directly affect only one component but contaminate a larger set of valuable assets. On the other hand, updates to the EA will also be reflected in the risk information, which improves the precision and updatability of risk information.

#### IV. RISK CONCEPTS

In order to address the interoperability and standardization issues in RM and between RM and the related activities of Governance and EA, we propose a XML-based Domain Specific Language for RM (Risk-DL), supported by a formal definition of the RM concepts. For mathematical clarity, in this paper we formalize the RM concepts covered in the proposed framework.

To formalize the RM concepts, we use the notation proposed in the relational model [23], where a **relation schema** describes the attributes of each concept, and a **relation instance** is composed by a set of instances of the concepts (tuples) defined in the relation schema. More formally, let  $R(f_1 : D_1, \dots, f_n : D_n)$  be a relation schema, and for each  $f_i, 1 \geq i \leq n$ , let  $Dom_i$  be the set of values with the domain named  $Di$ . An instance of  $R$  is a set of tuples, where:

$$\langle f_1 : d_1, \dots, f_n : d_n \rangle \mid d_1 \in Dom_1, \dots, d_n \in Dom_n$$

Also, we define functions as  $f : D \rightarrow R$ , where  $f$  is the name of the function;  $D$  is the domain and  $R$  is the range of the function. Note that relations can be used to represent the allowed domains or the range of the functions.

Using this notation, the RM concepts are defined in Table I. These concepts are visually modeled by an UML Domain diagram [24] represented in Figure 3.

An **Asset** ( $A$ ) is any entity which has a value to the organization. Using the proposed language, an asset should

be represented by EA artifacts. For instance, an asset can correspond to an entity represented by a cell of the Zachman framework (e.g., business process, program, server). The **Asset Value** is determined by the function  $A_{Val}$ . The asset value estimation is decoupled from the asset concept to better integrate distinct types of valuation. We consider three types of valuation: (i) quantitative: where the value is estimated by a real number, allowing mathematical calculus to process this values; (ii) qualitative: where the value is a qualitative representation, and thus not supporting mathematical calculus; and (iii) semi-quantitative: where an initial qualitative value is transformed into a quantitative value to allow mathematical calculus.

Depending on the type of scenario, the asset value function ( $A_{Val}$ ) can be quantitative, semi-quantitative or qualitative, having  $D_{Aval}$  as the admissible range. On the other hand, the value of an asset is not an intrinsic property of the asset "per se", but a result of its integration in a specific environment. Indeed, the same asset can have completely different values if considered in different scenarios (or even if evaluated by stakeholders with different concerns).

A **Vulnerability** ( $V$ ) identifies a specific characteristic of an asset that exposes its value through a quantifiable **Vulnerability Exposure** ( $V_E$ ). Again, the fact that the vulnerability exposure can be determined by quantitative, semi-quantitative and qualitative methods, explains the decoupling of this function from the concept of vulnerability.

An **Event** ( $E$ ) represents any uncontrolled circumstance that has the ability to produce consequences on the value of assets. Again, the event is quantified by an **Event Likelihood** ( $E_L$ ) that can be determined by quantitative, semi-quantitative or qualitative methods. Considering quantitative values, the likelihood of the event cannot be 0% neither 100%. In fact, events that will never occur do not introduce any type of risks, while events that are certain to occur are known facts (if we know that an important technician will retire next month, we can not say that we have a risk of losing that technician, since it is a fact).

A **Risk** ( $R$ ) is determined by a triple composed by the event that can exploit a vulnerability of a specific asset. The **Risk Severity** is modeled by a function ( $R_S$ ) to quantify the impact that occurs if the event is able to exploit the vulnerability of the asset defined in this risk. Once again, this function can produce quantitative, semi-quantitative and qualitative results.

A **Control** ( $C$ ) is used to manage risks, trying to mitigate them. We propose three types of controls. First, a **Block Control** ( $C_B$ ) is a control to limit the probability of an event to occur. This way, a block control represents a function that determines a new likelihood for an event. This function has the same range ( $D_{EL}$ ) of the event likelihood function. Second, an **Elimination Control** ( $C_E$ ) intends to reduce the exposure of a vulnerability. This way, it determines a new exposure, using the same range ( $D_{VE}$ ) of the vulnerability exposure function. Finally, the **Reduction Control** ( $C_R$ ) assumes that the risk occurs and pretends to reduce its consequences, producing a result on

TABLE I  
FORMALIZATION OF RISK MANAGEMENT CONCEPTS

Concept	Formalization	Description
Asset	$A(aName : string, aType : D_{atype}, aRef : D_{aRef}, a_1 : D_1, \dots, a_n : D_n)$	$D_{atype}$ determines the domain of asset types; $D_{aRef}$ determines the reference of the asset to the EA.
Asset Value	$A_{Val} : A \rightarrow D_{Aval}$	Asset value (quantitative, semi-quantitative or qualitative).
Vulnerability	$V(name : string, vType : D_{vType}, asset : A)$	Identifies a vulnerability in an asset defined in $A$ . An asset can have several vulnerabilities.
Vulnerability Exposure	$V_E : V \rightarrow D_{VE}$	Function that determines the exposure of the vulnerability.
Event	$E(name : string, eType : D_{eType})$	it can be a threat (bad event) or an opportunity (positive event).
Event Likelihood	$E_L : E \rightarrow D_{EL}$	Initial estimation of the probability of occurrence of an event.
Risk	$R(E, A, V)$	Consequences that an event produce when exploiting a vulnerability.
Risk Severity	$R_S : R \rightarrow D_{RS}$	Severity of the impact produced by the occurrence of the risk.
Block Control	$C_B : E \rightarrow D_{EL}$	Control to block the event (reducing its probability).
Elimination Control	$C_E : V \rightarrow D_{VE}$	Control to eliminate a vulnerability (reducing its exposure).
Reduction Control	$C_R : R \rightarrow D_{RS}$	Control to reduce the severity of the impact produced by a risk.
Control	$C \equiv C_B \cup C_E \cup C_R$	Actions that can be taken to mitigate risks.
Cost	$Cost : C \rightarrow D_C$	Cost of implementing a control.
Policy	$P \equiv C_1, C_2, \dots, C_n$	where $C_i \in C$

the range ( $D_{RS}$ ) has happens in the risk severity function.

The adoption of a specific control has a **Cost** ( $Cost$ ) to the organization, which can also be determined in a quantitative, semi-quantitative or qualitative way.

Finally, the concept of **Policy** ( $P$ ) defines the set of controls that are managing the risks identified in a specific organization. Ideally, organizations procure an optimal policy to effectively handle risks at a minimum cost.

Note that the ranges:  $D_{atype}, D_{aRef}, D_{Aval}, D_{vType}, D_{eType}, D_{EL}, D_{VE}, D_{RS}, D_C$  have to be defined (in qualitative assessment, these ranges define the risk matrices). For instance,  $D_{Aval}$  can be defined as:  $D_{Aval} \equiv \{low, medium, high\}$ , meaning that the asset values can be qualitatively quantified by low, medium or high.

## V. MANAGING RISKS USING RISK-DL

Risk-DL is a XML<sup>3</sup> based vocabulary and schema to represent the risk concepts defined in Section IV. In fact, the Risk-DL defines the XML Schema<sup>4</sup>, in the form of a *.xsd* file, that should be used to create XML files defining risks.

The main objectives of Risk-DL include, but are not limited to: support interoperability between distinct sources of risk information; support of sharing, discovery, reuse and processing of risk information; enable the alignment between risks and organization artifacts, by linking assets to records (e.g., business processes) managed within an organization EA; reduce inconsistencies by formalizing the risk concepts; provide an open specification that enables risk information to be categorized and support human-machine and machine-machine interoperability, either internally when different units produce risk information or externally across multiple organizations. Also, XML uses a human language that can be easily understood by people and computers, being highly portable and platform independent. Moreover, this solution also takes

advantage of common XML properties, like its extensibility, which simplifies the evolution of Risk-DL, as well as the assurance of compatibility between different versions of the same language.

Figure 4 shows an excerpt of the Risk-DL definition *.xsd* file (left side) and an example of XML file defining an asset. In this example, the *BPI.2.3 Central data validation* is a business process defined in the EA. Both the asset type and value are previously defined in the XML file (omitted in the paper due to space limitations).

The use of a formalized XML representation for risk information, facilitates the automatic definition of risk information. For instance, an organization that has an *Asset Management* system, or a *Configuration Management Database*, can define mappings to automatically generate the Risk-DL structure to represent assets. This fact, not only simplifies the Risk Management process, but also increases the quality and alignment between risk activities and other organization processes.

The proposed overall solution to manage risk information is detailed in Figure 5. An Operator represents the business worker that is responsible to interact with the system. First, the Operator provides a Risk Description that is transformed into the Risk-DL Specification of these risks, using the Risk Modeling component. The transformation into the Risk-DL Specification is supported by the Metadata Registry (MDR) component. The use of a MDR intends to ensure interoperability between different risk representations, as proposed by ISO/IEC 11179 [25], where an information system is responsible for managing and publishing descriptive information about resources (risk information). A MDR promotes interoperability by using a common reference model to register the descriptions of the data (semantic interoperability) and the context where it should be used (pragmatic interoperability), while registering version information about the data object (dynamic interoperability) and the corresponding relations (conceptual interoperability), whether related to relationships between different versions of the same or different data

<sup>3</sup><http://www.w3.org/XML>

<sup>4</sup><http://www.w3.org/XML/Schema>

```

<xs:element name="asset">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="riskdl:name"/>
      <xs:element ref="riskdl:asset-Type"/>
      <xs:element ref="riskdl:description"/>
      <xs:element ref="riskdl:asset-value"/>
      <xs:element ref="riskdl:properties"/>
      <xs:element ref="riskdl:vulnerabilities"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="asset-Type" type="xs:NCName"/>
<xs:element name="asset-value" type="xs:NCName"/>
<xs:element name="properties">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="riskdl:property"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="property">
  <xs:complexType>
    <xs:attribute name="name" use="required"/>
    <xs:attribute name="value" use="required"/>
  </xs:complexType>
</xs:element>
</asset>
<asset>
  <name>BP1.2.3 Central data validation</name>
  <asset-Type>Process</asset-Type>
  <description>GestBarragens Central data validation</description>
  <asset-value>medium</asset-value>
  <properties>
    <property name="BP name" value="Data valiadation"></property>
    <property name="Inputs" value="BP1, BP2"></property>
    <property name="Specification" value="bpel"></property>
    <property name="File" value="centralDataVal.bpel"></property>
  </properties>
  <vulnerabilities>
    <vulnerability>
      <name>Lack of metadata</name>
      <vulnerability-Type>Process</vulnerability-Type>
      <vulnerability-Exposure>medium-high</vulnerability-Exposure>
    </vulnerability>
    <vulnerability>
      <name>Lack of qualified staff</name>
      <vulnerability-Type>Process</vulnerability-Type>
      <vulnerability-Exposure>very-high</vulnerability-Exposure>
    </vulnerability>
  </vulnerabilities>
</asset>

```

Fig. 4. Risk-DL example

objects. This way, the syntactic representation of the Risk-DL language is irrelevant for the overall purpose of this solution.

Consequently, the architecture supports different versions of Risk-DL, as well as other risk representations. The Operator can manually define the risk information (using a web interface) or automatically transform its specific format to represent risk information (e.g., the asset list) into Risk-DL. Automatic transformations are supported by the MDR component if specific risk format is registered into the MDR. The mapping between a specific format and Risk-DL are partially inferred by the MDR (if the mapping is not complete, the MDR component provides an interface to specify schema mappings). The rationale for this approach is based on the separation of concerns between the risk information and the services processing it.

The Risk Analyzer parses a Risk-DL Specification (XML file) and generates an internal Risk Representation (a set of *Java* objects) to be used and processed by the Plan Generator, which is responsible to produce options to manage risks (Risk Plans), based on previous knowledge stored in the Risk Library. The Plan Generator proposes controls based on the Risk Library, but other controls can be specified through Risk-DL. Based on the available controls, the set of Risk Plans is generated.

The Risk Library represents a risk knowledge base, locally storing validated risk information as, for instance, risks used in previous scenarios, risk matrices, threats, vulnerabilities, assets, controls, plans, etc.

In order to support the complex decision of the most suitable risk treatment plan for a specific scenario, the Plan Evaluator produces a set of statistics that can be used to compare plans. When risks were defined according to different types of scores (quantitative, qualitative, semi-quantitative, or different scales), the Risk Normalizer is responsible to normalize scores, turning it possible to compare and rank risks defined using different methods.

Finally, the Report Generator produces Risk reports to support the decision on the optimal plan to apply. Also, risk information must be delivered to different stakeholders (with different concerns). Having this in consideration, the Report Generator is connected to the MDR to be able to provide different representations to view the risk information from the perspective of the concerns of every stakeholder.

Note that the proposed solution focuses on the risk dimension of the approach described in Section III. The relation to EA and Governance is expressed on the fact that Risk-DL maps risks to artifacts defined in the EA. Also, the interoperability supported by the way that risks are defined, allows the integration of risks delivered by different organization units (usually done in silos without any connection to other risks identified in the organization), supporting a common view and integrated management of risks. Finally, the reporting mechanisms provide metrics and reports to support an effective decision making, based on risk and optional paths to deal with them

## VI. CONCLUSION

Risks exist everywhere and everyday, whether or not it is recognized by the stakeholders affected by them. One of the main challenges that the risk community has to address is on the modeling of risk information. In fact, among other issues, risks involve a highly heterogeneous set of assets, events, methods, stakeholders and responsibilities, requiring adaptable methods and tools to support the exchange and interoperability of risk information. On the other hand, RM in general and risk assessment in particular, tend to be done in silos, by distinct teams with potential different views on the same risks.

This type of issues are commonly addressed by the EA community, where organizations are modeled from multiple views and different concerns of the involved stakeholders (viewpoints). In this paper, we propose to take advantage of the EA methods and best practices to facilitate the exchange

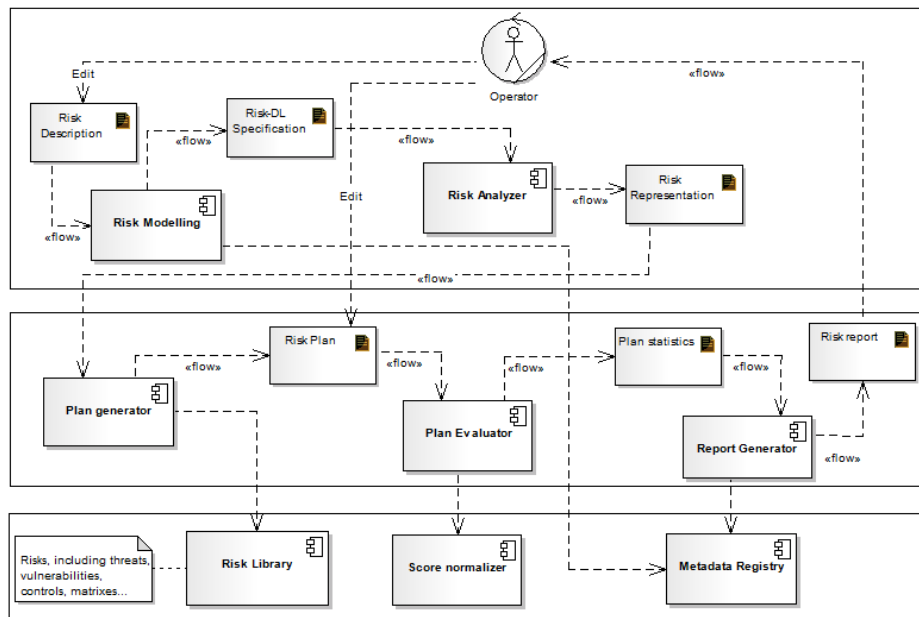


Fig. 5. Solution Overview

of risk information, as well as providing an organization-wide view on risks. As a consequence, risks can be tracked back to EA artifacts, allowing a detailed and precise analysis of the spreading of a specific risk.

We propose a solution that is decoupled from any risk or EA representation, so that it does not depend on any formalism. The proposed solution includes a XML-based language to formalize risks and integrate a Metadata Registry to support the communication with different risk representations, as well as providing different views to communicate risk information to the stakeholders.

#### ACKNOWLEDGMENT

This work was supported by FCT (INESC-ID multi-annual funding) through the PIDDAC Program funds and by the projects SHAMAN and TIMBUS, partially funded by the EU under the FP7 contracts 216736 and 269940.

#### REFERENCES

- [1] I. Sommerville, *Software Engineering (7th Edition)*. Addison Wesley, 2004.
- [2] "ISO Guide 73:2009. Risk management – Vocabulary," 2009.
- [3] "Software Engineering Institute. Capability Maturity Model Integration for Development. Version 1.3," Carnegie Mellon University, November 2010.
- [4] M. Stamatelatos, W. Vesely, J. Dugan, J. Fragola, J. Minarick, and J. Railsback, "Fault tree handbook with aerospace applications," NASA, 2002.
- [5] DoD: *Military Standard, Procedures for Performing a Failure Mode, Effects, and Critical Analysis (ML-STD-1692A)*. US Department of Defense, 1980.
- [6] A. Dardenne, A. van Lamswerde, and S. Fickas, "Goal-directed requirements acquisition," in *Science of Computer Programming*, vol. 20, 1993, pp. 3–50.
- [7] A. Anton, "Goal-based requirements analysis," in *International Conference on Requirements Engineering*, IEEE Computer Society. Washington DDC, USA: IEEE Computer Society, 1996.
- [8] Y. Asnar and P. Giorgini, "Modelling risk and identifying countermeasure in organizations," in *CRITIS*, J. Lopez, Ed. Springer-Verlag, 2006, pp. 55 – 66.
- [9] S. Maziol, "Risk management: Protect and maximize stakeholder value," Oracle Governance, Risk, and Compliance, White Paper, February 2009.
- [10] S. Biazzo, "Process mapping techniques and organisational analysis: Lessons from sociotechnical system theory," *Business Process Management Journal*, vol. 8, no. 1, pp. 42–52, 2002.
- [11] "ISO 31000:2009. Risk management – Principles and guidelines," 2009.
- [12] "Managing Risk from the Mailroom to the Boardroom," June 2003.
- [13] "Committee of Sponsoring Organizations of the Treadway Commission. Enterprise Risk Management – Integrated Framework," 2004.
- [14] "IT Governance Institute. CobiT 4.1. Framework – Control Objectives – Management Guidelines – Maturity Models," 2007.
- [15] W. V. Grembergen, *Strategies for information technology governance*. Idea Group Publishing, 2003.
- [16] "ISO/IEC 27001:2005. Information technology – Security techniques – Information security management systems – Requirements," 2005.
- [17] "IEEE 1471:2000 - Recommended Practice for Architectural Description for Software-Intensive Systems," 2001.
- [18] T. Mens, J. Magee, and B. Rumpe, "Evolving software architecture descriptions of critical systems," *Computer*, vol. 43, pp. 42–48, 2010.
- [19] P. Sousa, A. Caetano, V. A., C. Pereira, and J. Tribolet, "Enterprise architecture modeling with the unified modeling language," in *IGI Global*, 2006.
- [20] J. Zachman, "A framework for information systems architecture," *IBM Systems Journal*, vol. 12, no. 6, pp. 276–292, 1987.
- [21] "The Open Group. TOGAF Version 9. Zaltbommel, Netherlands: Van Haren Publishing," 2009.
- [22] M. Frigo and R. Anderson, "A strategic framework for governance, risk, and compliance," *Strategic Finance*, vol. 90, no. 8, February 2009.
- [23] R. Ramakrishnan and J. Gehrke, *Database Management Systems (Second Edition)*. McGRAW-HILL International Editions, 2000.
- [24] "ISO/IEC 19501:2005. Unified modeling language specification, v. 1.4.2 formal/05-04-01," January 2005.
- [25] "ISO/IEC 11179-1:2004. Information Technology – Metadata Registries (MDR) – Part 1: Framework," 2004.

# 3<sup>rd</sup> Workshop on Advances in Programming Languages

**P**ROGRAMMING languages are programmers' most basic tools. With appropriate programming languages one can drastically reduce the cost of building new applications as well as maintaining existing ones. In the last decades there have been many advances in programming languages technology in traditional programming paradigms such as functional, logic, and object-oriented programming, as well as the development of new paradigms such as aspect-oriented programming. The main driving force was and will be to better express programmers' ideas. Therefore, research in programming languages is an endless activity and the core of computer science. New language features, new programming paradigms, and better compile-time and run-time mechanisms can be foreseen in the future.

The aims of this event is to provide a forum for exchange of ideas and experience in topics concerned with programming languages and systems. Original papers and implementation reports are invited in all areas of programming languages.

## TOPICS

Major topics of interest include but are not limited to the following:

- Automata theory and applications
- Compiling techniques
- Domain-specific languages
- Formal semantics and syntax
- Generative and generic programming
- Grammarware and grammar based systems
- Knowledge engineering languages, integration of knowledge engineering and software engineering
- Languages and tools for trustworthy computing
- Language theory and applications
- Language concepts, design and implementation
- Markup languages (XML)
- Metamodeling and modeling languages
- Model-driven engineering languages and systems
- Practical experiences with programming languages
- Program analysis, optimization and verification
- Program generation and transformation
- Programming paradigms (aspect-oriented, functional, logic, object-oriented, etc.)
- Programming tools and environments
- Proof theory for programs

- Specification languages
- Type systems
- Virtual machines and just-in-time compilation
- Visual programming languages

## WAPL 2011 CHAIR

**Ivan Luković**, University of Novi Sad, Serbia  
ivan@uns.ac.rs

## PROGRAM COMMITTEE

**Fei Cao**, Microsoft, USA

**Haiming Chen**, Chinese Academy of Sciences, China

**Tom Dinkelaker**, Technische Universität Darmstadt, Germany

**Johan Fabry**, Universidad de Chile, Chile

**Krešimir Fertalj**, University of Zagreb, Croatia

**Rémi Forax**, Université de Marne-la-Vallée, France

**Pedro Henriques**, University of Minho, Portugal

**Zoltán Horváth**, Eotvos Lorand University, Hungary

**Mirjana Ivanović**, University of Novi Sad, Serbia

**Jan Janousek**, Czech Technical University in Prague, Czech Republic

**Ján Kollár**, Technical University Kosice, Slovakia

**Tomaž Kosar**, University of Maribor, Slovenia

**Shih Hsi "Alex" Liu**, California State University, USA

**Pablo Martínez López**, Universidad Nacional de Quilmes, Argentina

**Ivan Luković**, University of Novi Sad, Serbia

**Federica Mandreoli**, University of Modena, Italia

**Marjan Mernik**, University of Maribor, Slovenia

**Hanspeter Mössenböck**, Johannes Kepler Universität Linz, Austria

**Nikolaos Papaspyrou**, National Technical University of Athens, Greece

**Maria João Varanda Pereira**, Instituto Politecnico de Braganca, Portugal

**Jaroslav Porubán**, Technical University Kosice, Slovakia

**José Luis Sierra Rodríguez**, Universidad Complutense de Madrid, Spain

**Vladimir Safonov**, St.Petersburg University, Russia

**Boštjan Slivnik**, University of Ljubljana, Slovenia

**Zdzisław Sławski**, Wrocław University of Technology, Poland



# Implementation of the Domain-Specific Language EasyTime using a LISA Compiler Generator

Iztok Jr. Fister<sup>\*</sup>,  
Marjan Mernik,<sup>†</sup> Iztok Fister<sup>‡</sup> and Dejan Hrnčič<sup>§</sup>  
<sup>\*</sup>Faculty of Electrical Engineering and Computer Science  
University of Maribor  
Smetanova ul. 17, SI-2000 Maribor, SLOVENIA

Email: iztok.fister@guest.arnes.si

<sup>†</sup> Email: marjan.mernik@uni-mb.si

<sup>‡</sup> Email: iztok.fister@uni-mb.si

<sup>§</sup> Email: dejan.hrnccic@uni-mb.si

**Abstract**—A manually time-measuring tool in mass sporting competitions cannot be imagined nowadays because many modern disciplines, such as IronMan, take a long time and, therefore, demand additional reliability. Moreover, automatic timing devices, based on RFID technology, have become cheaper. However, these devices cannot operate stand-alone because they need a computer measuring system that is capable of processing the incoming events, encoding the results, assigning them to the correct competitor, sorting the results according to the achieved times, and then providing a printout of the results. In this article, the domain-specific language EasyTime is presented, which enables the controlling of an agent by writing the events in a database. In particular, we are focused on the implementation of EasyTime with a LISA tool that enables the automatic construction of compilers from language specifications using Attribute Grammars. By using of EasyTime, we can also decrease the number of measuring devices. Furthermore, EasyTime is universal and can be applied to many different sporting competitions in practice.

**Index Terms**—domain-specific language, parser, code generator, time-measuring, RFID technology

## I. INTRODUCTION

**I**N THE past, timekeepers measured the time manually. The time from a timer was assigned to competitors based on their starting number and these competitors were then ordered according to their achieved results and category. Later, the manual timers were replaced by the timers with an automatic time register that was capable of capturing and printing out registered times. However, an assigning the time to a competitor based on their starting number was still done manually. This work could be avoided by using the electronic measuring technology which, in addition to registering the time, also enabled the registering of the competitors' starting number. An expansion of RFID (Radio Frequency Identification) technology has helped this measuring technology become less expensive ([2], [12]) and accessible to a wider range of users (e.g., sport clubs, organizers of sporting competitions). Moreover, they were able to compete with time-measuring monopolies at smaller competitions.

In addition to measuring technology, a flexible computer system is also needed to monitor the results. The proposed computer system enables the monitoring of different sporting competitions with a various number of measuring devices and measuring points, the online recording of events, the writing of results, as well as efficiency and security. The measuring device is dedicated to the registration of events and is triggered either automatically, when the competitor crosses the measuring point that acts as an electromagnetic antenna fields with an appropriate RFID tag, or manually, when an operator presses the suitable button on a personal computer that acts as a timer. The control point is the place where the organizers want to monitor results. Until now, each control point required its own measuring device. However, modern electronic measuring devices now allow for the handling of multiple control points simultaneously. Moreover, each registered event can have a different meaning, depending on the situation in which it is generated. Therefore, the event is handled by the measuring system according to the rules that are valid for the control point. As a result, the number of control points (and measuring devices) can be reduced with more complex measurements. Fortunately, the rules controlling events can be described easily with the use of a domain-specific language (DSL) [7]. With this DSL, the measurements of different sporting competitions can be accomplished with the easy pre-configuration of rules.

A DSL is suited to an application domain and has certain advantages over general purpose languages (GPL) in a specific domain [6], [7]. The GPL is dedicated to writing software in a wider range of application domains. With these languages general problems are usually solved. However, to change the behavior of a program written in a GPL, a programmer is necessary. On the other hand, the advantages of DSL are reflected in its greater expressive power and hence increased productivity, ease of use (even for domain experts that are not programmers), and easier verification and optimization [7]. In this article, a DSL called EasyTime and its implementation is presented. EasyTime is intended to control the agents that are responsible for recording events from the measuring devices



into a database. Therefore, the agents are crucial elements of the proposed measuring system. Finally, EasyTime was successfully employed in practice as well. For instance, it measured times in a World championship for the ultra double triathlon in 2009 [4] and a National Championship in the time trials for bicycle in 2010 [4].

The structure of the rest of the article is as follows; In the second section, the problems that are accompanied with time-measuring at sporting competitions are illustrated. We focus primarily on triathlon competitions, because they contain three disciplines that need to be measured and also because of their long duration. The design of DSL EasyTime is briefly shown in section three. In the fourth section, the implementation of the EasyTime compiler is described, while in the fifth section the execution of the program written in EasyTime is explained. Finally, the article is concluded with a short analysis of the work performed and a look at future work.

## II. MEASURING TIME IN SPORTING COMPETITIONS

In practice, the measuring time in sporting competitions can be performed manually (classically or with a computer timer) or automatically (with a measuring device). The computer timer is a program that usually runs on a workstation (personal computer) and measures in real time. Thereby, a processor tact is exploited. The processor tact is the velocity with which the processor's instructions are interpreted. A computer timer enables the recording of events that are generated by the competitor crossing the measure points (MP) similar to the measuring device. In that case, however, the event is triggered by an operator pressing the appropriate button on the computer. The operator generates events in the form of  $\langle \#, MP, TIME \rangle$ , where  $\#$  denotes the starting number of a competitor,  $MP$  is the measuring point and  $TIME$  is the number of seconds since 1.1.1970 at 0:0:0 (timestamp). One computer timer represents one measuring point.

Today, the measuring device is usually based on RFID (Radio Frequency Identification) technology [3], where an identification is performed with electromagnetic waves in the range of radio frequencies and consists of the following elements:

- readers of RFID tags,
- primary memory,
- LCD monitor,
- numeric keyboard, and
- antenna fields.

More antenna fields can be connected on the measuring device. One antenna field represents one measuring point. Each competitor generates an event by crossing the antenna field with passive RFID tags that include an identification number. This number is unique and differs from the starting number of the competitor. The event from the measuring device is represented in the form of  $\langle \#, RFID, MP, TIME \rangle$ , where the identification number of the RFID tag is added to the previously mentioned triplet.

The measuring devices and workstations running the computer timer can be connected to the local area network.

Communication with devices is performed by a monitoring program, i.e. an agent, that runs on the database server. The agent communicates with the measuring device via the TCP/IP sockets and appropriate protocol. Usually, the measuring devices support a protocol *Telnet* that is character-stream oriented and, therefore, easy to implement. The agent employs the file transfer protocol to communicate with the computer timer.

### A. Example: Time Measuring Times at Triathlons

Special conditions apply for triathlon competitions, where one competition consists of three disciplines. In this article, therefore, we will devote the most of our attention to this problem.

The triathlon competition was first held in the USA in the year 1975. Today, the competition has become an Olympic discipline as well. The triathlon competition is performed as follows: first, the athletes swim, then they ride a bike and finally they run. In practice, all these activities are performed continuously. However, the transition times, i.e. the time that elapses when the competitor shifts from swimming to bicycling and from bicycling to running, are added to the summary result. There are various types of triathlon competitions that differ according to the length of various courses. To make things easier, the organizers will often employ the round courses (laps) of shorter lengths instead of one long course. Therefore, the difficulty of measuring time is increased because the time for each lap needs to be measured.

Measuring time in triathlon competitions can be divided into nine control points (Fig. 1). The control point (CP) is a location on the triathlon course, where the organizers need to check the measured time. This can be intermediate or final. As can be seen in Fig. 1, when dealing with a double triathlon there are 7.6 km of swimming, 360 km of bicycling and 84 km of running, while the swimming course of 380 meters consists of 20 laps, the bicycling course of 3.4 kilometers contains 105 laps and the running course of 1.5 kilometers has 55 laps.

Therefore, the final result of each competitor in a triathlon competition (CP8) consists of five final results: the swimming time SWIM (CP2), the time for the first transition TA1 (CP3), the time spent bicycling BIKE (CP5), the time for the second transition TA2 (CP6), the time spent running RUN (CP8), and three intermediate results: the intermediate time for swimming (CP1), the intermediate time for bicycling (CP4) and the intermediate time for running (CP7). However, the current time INTER\_x and the number of remaining laps LAPS\_x are measured by the intermediate results, where  $x = \{1, 2, 3\}$  denotes the appropriate discipline (1=SWIM, 2=BIKE and 3=RUN).

Suppose a measuring device with two measuring places (MP3 and MP4) is available to measure a triathlon competition as illustrated in Fig. 1 and the competition is performed in one location. In that case, the last crossing over the MP3 denotes the time of CP5, the first crossing over the MP4 denotes the time CP6 and the last crossing over the MP4 is the final time (CP8). The measuring places MP1 and MP2

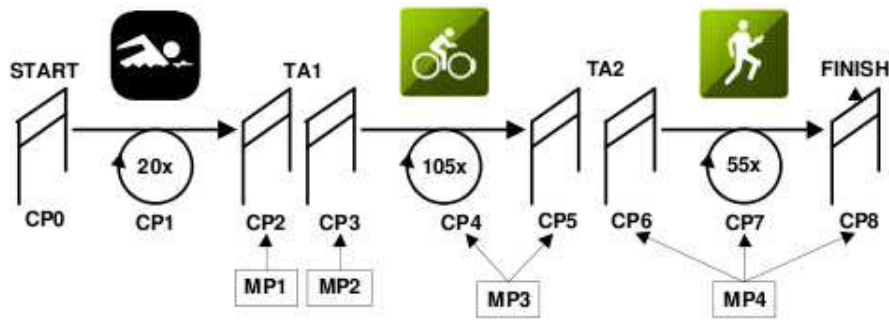


Fig. 1. Definition of control points in the triathlon.

are conducted manually by the computer timers. The number of control points, thereby, can be reduced by three if the measuring system and the appropriate setting of control points are employed. In fact, the 162 events pro competitor (87.6% of all) can be measured by this device. Moreover, because the technology for measuring swimmers in lakes and seas is very expensive and, therefore, usually recorded by referees manually, this real-world competition would be covered by measuring the 98% of all events.

In order to achieve this goal, however, the DSL EasyTime was developed and employed in practice by conducting measurements at the World Championship in Double Triathlon in 2009. Note that measurements were realized according to Fig. 1. In the next sections, a design, an implementation, and an operation of EasyTime is presented.

### III. THE DESIGN OF THE DOMAIN-SPECIFIC LANGUAGE EASYTIME

Typically, the development of a DSL consists of the following phases:

- a domain analysis,
- a definition of an abstract syntax,
- a definition of a concrete syntax,
- a definition of a formal semantics, and
- an implementation of the DSL.

Domain analysis provides an analysis of the application domain, i.e. measuring time in sporting competitions. The results of this analysis define the concepts of EasyTime that are typically represented in a feature diagram [13]. The feature diagram describes also dependencies between the concepts of the DSL. Thus, each concept can be broken down into features and sub-features. In the case of EasyTime, the concept *race* consists of the sub-features: *events* (e.g., *swimming*, *bicycling* and *running*), *control points*, *measuring time*, *transition area*, and *agents*. Each *control point* is described by its *starting* and *finish* line and at least one *lap*. In addition, the feature *transition area* can be introduced as the difference between the finish and start times. Both *updating time* and *decrementing laps* are sub-features of *measuring time*. However, an *agent* for the processing of

events received from the measuring device is needed. It can act either *automatically* or *manually*.

Domain analysis identifies several concepts in the application domain that needs to be mapped into EasyTime syntax and semantics [7]. At first, the abstract syntax is defined (context-free grammar). Each concept obtained from the domain analysis is mapped to a non-terminal in the context-free grammar; additionally, some new non-terminal and terminal symbols are defined. The translations of the EasyTime domain concepts to non-terminals are presented in Table I, while an abstract syntax is presented in Table II. Interestingly, a description of agents and measuring places cannot be found in other DSLs or GPLs. While attribute declaration is similar to variable declaration in many other programming languages there is the distinction that variables are actually database attributes allocated for every competitor. Some statements, such as assignment, conditional statement, and compound statement can be found in many other programming languages, while decrement attributes and update attributes are domain-specific constructs.

TABLE II  
THE ABSTRACT SYNTAX OF EASYTIME

$P \in \mathbf{Pgm}$	$A \in \mathbf{Adec}$
$D \in \mathbf{Dec}$	$M \in \mathbf{MeasPlace}$
$S \in \mathbf{Stm}$	$b \in \mathbf{Bexp}$
$a \in \mathbf{Aexp}$	$n \in \mathbf{Num}$
$x \in \mathbf{Var}$	$file \in \mathbf{FileSpec}$
$ip \in \mathbf{IpAddress}$	
$P$	$::= A D M$
$A$	$::= n \text{ manual } file \mid n \text{ auto } ip \mid A_1; A_2$
$D$	$::= \text{var } x := a \mid D_1; D_2$
$M$	$::= \text{mp}[n_1] \rightarrow \text{agt}[n_2] S \mid M_1; M_2$
$S$	$::= \text{dec } x \mid \text{upd } x \mid x := a \mid (b) \rightarrow S \mid S_1; S_2$
$b$	$::= \text{true} \mid \text{false} \mid a_1 = a_2 \mid a_1! = a_2$
$a$	$::= n \mid x$

In the formal semantics phase, a meaning of the EasyTime language constructs is prescribed. Each language construct, belonging to the syntax domain, is mapped into an appropriate semantic domain (Table III) by semantic functions  $CP$ ,  $CM$ ,  $CS$ ,  $A$  (Table IV). In addition, semantic functions  $\mathcal{A}$  and  $CM$  are illustrated by Table V.

TABLE I  
TRANSLATION OF THE APPLICATION DOMAIN CONCEPTS TO A CONTEXT-FREE GRAMMAR

Application domain concepts	Non-terminal	Formal semantics	Description
Race	P	$\mathcal{CP}$	Description of agents; control points; measuring places.
Events (swimming, cycling, running)	none	none	Measuring time is independent from the type of an event. However, good attribute's identifier in control points description will resemble the type of an event.
Transition area times	none	none	Can be computed as difference between events final and starting times.
Control points (start, number of laps, finish)	D	$\mathcal{D}$	Description of attributes where start and finish time will be stored as well as remaining laps.
Measuring places (update time, decrement lap)	M	$\mathcal{CM}$	Measuring place id; agent id, which will control this measuring place; specific actions which will be performed at this measuring place (e.g., decrement lap).
Agents (automatic, manual)	A	$\mathcal{A}$	Agent id; agent type (automatic, manual); agent source (file, ip).

TABLE III  
SEMANTIC DOMAINS

<b>Integer</b> ={... -3, -2, -1, 0, 1, 2, 3...}	$n \in \mathbf{Integer}$
<b>Truth-Value</b> ={true, false}	
<b>State</b> =Var $\rightarrow$ Integer	$s \in \mathbf{State}$
<b>AType</b> ={manual, auto}	
<b>Agents</b> =Integer $\rightarrow$ AType $\times$ (FileSpec $\cup$ IpAddress)	$ag \in \mathbf{Agents}$
<b>Runners</b> =(Id $\times$ RFID $\times$ LastName $\times$ FirstName)*	$r \in \mathbf{Runners}$
<b>DataBase</b> =(Id $\times$ Var <sub>1</sub> $\times$ Var <sub>2</sub> $\times$ ... $\times$ Var <sub>n</sub> )*	$db \in \mathbf{DataBase}$
<b>Code</b> =String	$c \in \mathbf{Code}$

TABLE IV  
TRANSLATION OF THE SYNTAX DOMAIN TO SEMANTIC DOMAINS BY SEMANTIC FUNCTIONS

Syntax Domain	Semantic Function	Semantic Domain
<b>Pgm</b>	$\mathcal{CP}$	<b>Code</b> $\times$ <b>Integer</b> $\times$ <b>Database</b>
<b>MeasPlace</b>	$\mathcal{CM}$	<b>Code</b> $\times$ <b>Integer</b>
<b>Stm</b>	$\mathcal{CS}$	<b>Code</b>
<b>Adecs</b>	$\mathcal{A}$	<b>Agents</b>

The sample program written in EasyTime that covers the measuring time in the double ultra triathlon as illustrated by Fig. 1 is presented by Algorithm 1.

More details of EasyTime syntax and semantics are presented in [4]. In this article, we are focused on the implementation phase as presented in the next section.

#### IV. THE IMPLEMENTATION OF THE DOMAIN-SPECIFIC LANGUAGE EASYTIME

Our motivation was to automatize an implementation phase as much as possible. Therefore, we use a compiler generator that can convert a formal description of a programming language into a compiler/interpreter for that language. Several recent compiler generators accept descriptions in terms of attribute grammars or denotational semantics [11]. Although

TABLE V  
TRANSLATION OF AGENTS AND MEASURING PLACES

$\mathcal{A}:\mathbf{Adecs} \rightarrow \mathbf{Agents}$	$\rightarrow$	<b>Agents</b>
$\mathcal{A}[n \text{ manual } file]ag$	=	$ag[n \rightarrow (manual, file)]$
$\mathcal{A}[n \text{ auto } ip]ag$	=	$ag[n \rightarrow (auto, ip)]$
$\mathcal{A}[A_1; A_2]ag$	=	$\mathcal{A}[A_2](\mathcal{A}[A_1]ag)$
$\mathcal{CM}:\mathbf{MeasPlace} \rightarrow \mathbf{Agents}$	$\rightarrow$	<b>Code</b> $\times$ <b>Integer</b>
$\mathcal{CM}[\mathbf{mp}[n_1] \rightarrow \mathbf{agnt}[n_2]S]ag$	=	(WAIT $i : \mathcal{CS}[S](ag, n_2), n_1)$
$\mathcal{CM}[M_1; M_2]ag$	=	$\mathcal{CM}[M_1]ag : \mathcal{CM}[M_2]ag$

**Algorithm 1** EasyTime program for measuring time in a triathlon competition as illustrated in Fig. 1

```

1: 1 manual "abc.res";
2: 2 auto 192.168.225.100;
3:
4: var ROUND1 := 20;
5: var INTER1 := 0;
6: var SWIM := 0;
7: var TRANS1 :=0;
8: var ROUND2 := 105;
9: var INTER2 :=0;
10: var BIKE := 0;
11: var TRANS2 :=0;
12: var ROUND3 := 55;
13: var INTER3 := 0;
14: var RUN := 0;
15:
16: mp[1]  $\rightarrow$  agnt[1] {
17:   (true)  $\rightarrow$  upd SWIM;
18:   (true)  $\rightarrow$  dec ROUND1;
19: }
20: mp[2]  $\rightarrow$  agnt[1] {
21:   (true)  $\rightarrow$  upd TRANS1;
22: }
23: mp[3]  $\rightarrow$  agnt[2] {
24:   (true)  $\rightarrow$  upd INTER2;
25:   (true)  $\rightarrow$  dec ROUND2;
26:   (ROUND2 == 0)  $\rightarrow$  upd BIKE;
27: }
28: mp[4]  $\rightarrow$  agnt[2] {
29:   (true)  $\rightarrow$  upd INTER3;
30:   (ROUND3 == 55)  $\rightarrow$  upd TRANS2;
31:   (true)  $\rightarrow$  dec ROUND3;
32:   (ROUND3 == 0)  $\rightarrow$  upd RUN;
33: }

```

many compiler generators exist today, we selected a LISA compiler-compiler that was developed at the University of Maribor in the late 1990s [8]. The LISA tool produces highly efficient source code for: scanner, parser, interpreter or compiler in Java. The lexical and syntactical parts of a language specification in LISA supports various well known formal methods, like regular expressions and BNF. LISA provides two kinds of user interfaces:

- a graphic user interface (GUI) (Fig. 2), and
- a Web-Service user interface.

The main features of LISA are as follows:

- since it is written in Java, LISA works on all Java platforms,
- a textual or a visual environment,
- an Integrated Development Environment (IDE), where users can specify, generate, compile and execute programs on the fly,
- visual presentations of different structures, such as finite-state-automata, BNF, a dependency graph, a syntax tree, etc.,
- modular and incremental language development [9].

LISA specifications are based on Attribute Grammar (AG) [10] that has been introduced by D.E. Knuth [5]. The attribute grammar is a triple  $AG = \langle G, A, R \rangle$ , where  $G$  denotes a context-free grammar,  $A$  a finite set of attributes and  $R$  a finite set of semantic rules. In line with this, LISA specifications include:

- lexical regular definitions,
- attribute definitions,
- semantic rules, and
- operations on semantic domains.

Lexical specifications for EasyTime in LISA (Figure 2) are similar to those used in other compiler generators. While LISA automatically infers whether an attribute is inherited or synthesized [5], the type of an attribute must be specified (Figure 2). For example, the attribute *code* represents generated code using translation functions, the attribute *outAG* is the synthesized attribute and *inAG* the inherited attribute representing agents (*ag* from semantic specifications). The correspondence between attributes in LISA specifications and a semantic description of EasyTime is shown in Table VI.

TABLE VI

TRANSLATING OF SEMANTIC FUNCTION TO LISA SPECIFICATIONS		
Semantic Spec.	Semantic Domain	LISA Attributes
<i>c</i>	<b>Code</b>	String *.code
<i>ag</i>	<b>Agents</b>	Hashtable *.inAG, *.outAG
<i>s</i>	<b>State</b>	Hashtable *.inState, *.outState
<i>n</i>	<b>Integer</b>	*.n

Essentially, we are focused on LISA specifications of semantic rules, which consists of generalized syntax rules that also encapsulate semantic rules. The semantic rules of EasyTime, as presented in Section III, were translated into the LISA specifications according to Table VII.

TABLE VII

TRANSLATING OF SEMANTIC FUNCTIONS TO LISA SPECIFICATIONS

Semantic Function	LISA Specification
$\mathcal{CP}$	Start
$\mathcal{CM}$	Mes_Places
$\mathcal{CS}$	Stmts
$\mathcal{A}$	Agents

Due to page limitations, only some part of mapping of EasyTime semantic specifications into LISA specifications are explained in this paper.

During the conversion from the abstract syntax to the concrete syntax, the production in the abstract syntax  $A_1; A_2$  denoting a sequence of agents, is translated to the following production in the concrete syntax:

$$AGENTS ::= AGENTS AGENT | \epsilon.$$

The semantic function  $\mathcal{A}[A_1; A_2]ag = \mathcal{A}[A_2](\mathcal{A}[A_1]ag)$  constructs  $ag \in Agents$ , which is a function from an integer, denoting an agent, into an agent's type (manual or auto) and an agent's ip or agent's file. This function is described in LISA for non epsilon cases as:

$$AGENTS[1].inAG = AGENTS[0].inAG;$$

$$AGENTS[0].outAG = insert(AGENTS[1].outAG, new$$

$$Agent(AGENT.number, AGENT.type, AGENT.file_ip));$$

and for epsilon cases as:

$$AGENTS.outAG = AGENTS.inAG;.$$

The net effect is that we are constructing a list, more precisely a hash table, of agents where we are recording the agent's number ( $AGENT.number$ ), the agent's type ( $AGENT.type$ ), and the agent's ip or file ( $AGENT.file_ip$ ). Those attributes are defined in the productions:

$$AGENT ::= \#Int auto \#ip; | \#Int manual \#file;$$

(Algorithm 2) and implements semantic functions:

$$\mathcal{A}[n \text{ manual } file]ag = ag[n \rightarrow (manual, file)]$$

and

$$\mathcal{A}[n \text{ auto } ip]ag = ag[n \rightarrow (auto, ip)].$$

---

### Algorithm 2 Translation of Agents into LISA specifications

---

```

1: rule Agents {
2:   AGENTS ::= AGENTS AGENT compute {
3:     AGENTS[1].inAG = AGENTS[0].inAG;
4:     AGENTS[0].outAG = insert(AGENTS[1].outAG,
5:     new Agent(AGENT.number, AGENT.type, AGENT.file_ip));
6:   }
7:   | epsilon compute {
8:     AGENTS.outAG = AGENTS.inAG;
9:   };
10: }
11: rule AGENT {
12:   AGENT ::= \#Int manual \#file compute {
13:     AGENT.number = Integer.valueOf(\#Int[0].value()).intValue();
14:     AGENT.type = "manual";
15:     AGENT.file_ip = \#file.value();
16:   };
17:   AGENT ::= \#Int auto \#ip compute {
18:     AGENT.number = Integer.valueOf(\#Int[0].value()).intValue();
19:     AGENT.type = "auto";
20:     AGENT.file_ip = \#ip.value();
21:   };
22: }
    
```

---

During the conversion from the abstract syntax to the concrete syntax, the production in the abstract syntax  $M_1; M_2$  denoting a sequence of measuring places is translated to the following production in the concrete syntax:

$$MES_PLACES ::= MES_PLACE MES_PLACES | MES_PLACE.$$

The translation function:

$$\mathcal{CM}[M_1]ag : \mathcal{CM}[M_2]ag$$

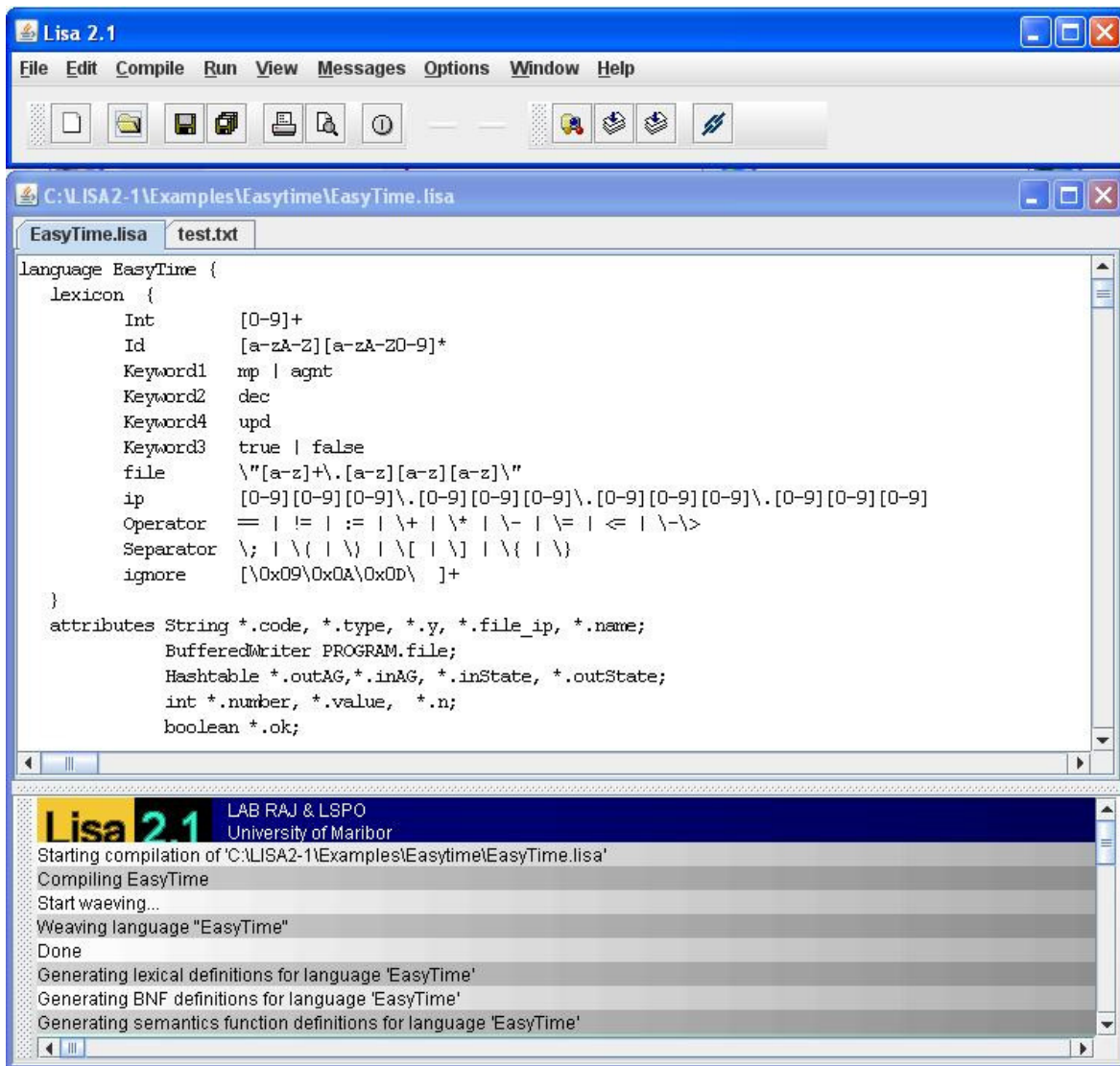


Fig. 2. LISA GUI.

translates into code the first construct  $M_1$  before the translation of the second construct  $M_2$  is performed. This function is described in LISA as:

```

MES_PLACES[0].code = MES_PLACE.code+
  "\n" + MES_PLACES[1].code;

```

with the following meaning: The code for the first construct  $MES\_PLACE$  is simply concatenated with the code from the second construct  $MES\_PLACES[1]$ . While the abstract syntax for the definition of the measuring place:

$$mp[n_1] \rightarrow agnt[n_2]S$$

is translated to the following production in the concrete syntax:

$$MES\_PLACE ::= mp [ \#Int ] \rightarrow agnt [ \#Int ] \{ STMTS \}.$$

The translation function:

$$(WAIT \ i : CS[S](ag, n_2), n_1)$$

is described in LISA as:

$$MES\_PLACE.code = "WAIT \ i" + STMTS.code +$$

$$", " + \#Int[0].value() + " " ;$$

Note, that in the implementation of this semantic function (Algorithm 3) many other attributes need to be defined. For example, a list of agents need to be propagated into statements ( $STMTS.inAG$ ), as well as a list of database attributes ( $STMTS.inState$ ).

Attributes that represent semantic information belong to various semantic domains (Figure 2). The attributes in LISA can be objects of classes specified in the library with already defined behavior (e.g., Hashtable) or can be objects of user-defined classes. For example, the previously mentioned semantic domain *Agents*, can be implemented as a hash table, where each element is an instance of the class Agent (Algorithm 4), where the agents's number, type and ip or file are stored. Moreover, various operations over semantic domain (e.g., insert into hash table - Algorithm 5) can be easily implemented using object-oriented programming. In Algorithm 5, it first

**Algorithm 3** Translation of MES\_PLACE into LISA specifications

```

1: rule Mes_places {
2:   MES_PLACES ::= MES_PLACE MES_PLACES compute {
3:     MES_PLACE.inAG = MES_PLACES[0].inAG;
4:     MES_PLACES[1].inAG = MES_PLACES[0].inAG;
5:     MES_PLACE.inState = MES_PLACES[0].inState;
6:     MES_PLACES[1].inState = MES_PLACES[0].inState;
7:     MES_PLACES[0].ok = MES_PLACE.ok && MES_PLACES[1].ok;
8:     MES_PLACES[0].code = MES_PLACE.code + "\n" +
        MES_PLACES[1].code;
9:   };
10: MES_PLACES ::= MES_PLACE compute {
11:   MES_PLACE.inAG = MES_PLACES.inAG;
12:   MES_PLACE.inState = MES_PLACES.inState;
13:   MES_PLACES.ok = MES_PLACE.ok;
14:   MES_PLACES.code = MES_PLACE.code;
15: };
16: }
17: rule MES_PLACE {
18:   MES_PLACE ::= mp \[ #Int \] \- \>
        agnt \[ #Int \] \{ STMTS \} compute {
19:     STMTS.inAG = MES_PLACE.inAG;
20:     STMTS.inState = MES_PLACE.inState;
21:     STMTS.n = Integer.valueOf(#Int[1].value()).intValue();
22:     MES_PLACE.ok = STMTS.ok;
23:     MES_PLACE.code = "(WAIT i " + STMTS.code + ", " +
        #Int[0].value() + ")";
24: };
25: }
    
```

checks if the agent is already defined. If this condition is not met a new agent is put into hash table.

**Algorithm 4** LISA definition of the semantic domain Agents

```

1: method M_Agent {
2:   class Agent {
3:     int number;
4:     String type;
5:     String file_ip;
6:     Agent ( int number, String type, String file_ip) {
7:       this.number = number;
8:       this.type = type;
9:       this.file_ip = file_ip;
10:    }
11:    public String toString() {
12:      return "(" + this.number + ", " + this.type + ", " + this.file_ip + ")";
13:    }
14:    public int getNumber() {
15:      return this.number;
16:    }
17:    public String getType() {
18:      return this.type;
19:    }
20:    public String getFile_ip() {
21:      return this.file_ip;
22:    }
23:  } // Java class
24: } // Lisa method
    
```

## V. OPERATION

Local organizers of sporting competitions were faced with two possibilities before the developing of EasyTime:

- to rent a specialized company to measure time,
- to measure time manually.

**Algorithm 5** Definition of the method Insert

```

1: method M_Insert ( {
2:   import java.util.*;
3:   Hashtable insert (Hashtable aAgents, Agent aAgent) {
4:     aAgents = (Hashtable)aAgents.clone();
5:     Agent hAgent=(Agent)aAgents.get(aAgent.getNumber());
6:     if (hAgent==null)
7:       aAgents.put(aAgent.getNumber(), aAgent);
8:     else
9:       System.out.println("Agent" + aAgent.getNumber() + "is already
        defined");
10:    return aAgents;
11:  } // Java method
12: } // Lisa method
    
```

The former possibility is expensive, while the latter can be very unreliable. However, the both objectives (i.e. inexpensiveness and reliability), can be fulfilled by EasyTime. On the other hand, producers of measuring devices usually deliver these units with software for collecting of events into a database. Then these events need to be post-processed (batch processed) to get the final results of competitors. Although this batch processing can be executed whenever the organizer desires each real-time application requests online processing. Fortunately, EasyTime enables both kinds of event processing.

In order to use the source program written in EasyTime by the measuring system, it needs to be compiled. Note that the code generation [1] of a program in EasyTime is performed only if the parsing is finished successfully. Otherwise the compiler prints out an error message and stops. For each measuring places individually, the code is generated by strictly following the rules, as defined in section III. An example of the generated code from the Algorithm 1 for controlling of the measurements, as illustrated by Fig. 1, is presented in Table VIII. Note that the generated code is saved into a database.

TABLE VIII  
TRANSLATED CODE FOR THE EASYTIME PROGRAM IN ALGORITHM 1

```

(WAIT i FETCH accessfile("abc.res") STORE SWIM
FETCH ROUND1 DEC STORE ROUND1, 1)

(WAIT i FETCH accessfile("abc.res") STORE TRANS1, 2)

(WAIT i FETCH connect(192.168.225.100) STORE INTER2
FETCH ROUND2 DEC STORE ROUND2
PUSH 0 FETCH ROUND2 EQ BRANCH( FETCH
connect(192.168.225.100) STORE BIKE, NOOP), 3)

(WAIT i FETCH connect(192.168.225.100) STORE INTER3
PUSH 55 FETCH ROUND3 EQ BRANCH( FETCH
connect(192.168.225.100) STORE TRANS2, NOOP)
FETCH ROUND3 DEC STORE ROUND3
PUSH 0 FETCH ROUND3 EQ BRANCH( FETCH
connect(192.168.225.100) STORE RUN, NOOP), 4)
    
```

As a matter of fact, the generated code is dedicated to the control of an agent by writing the events received from the measuring devices into the database. Typically, the program



code is loaded from the database only once. That is, only an interpretation of code could have any impact on the performance of a measuring system. Because this interpretation is not time consuming, it cannot degrade the performance of the system. On the other hand, the precision of measuring time is handled by the measuring device and it is not changed by the processing of events. In fact, the events can be processed as follows:

- batch: manual mode of processing, and
- online: automatic mode of processing.

The agent reads and writes events that are collected in a text file when the first mode of processing is assumed. Typically, events captured by a computer timer are processed in this mode. Here, the agent looks for the existence of the event text file that is configured in the agent statement. If it exists, the batch processing is started. When the processing is finished, the text file is archived and then deleted. The online processing is event oriented, i.e. each event that is generated by the measuring device is processed in time.

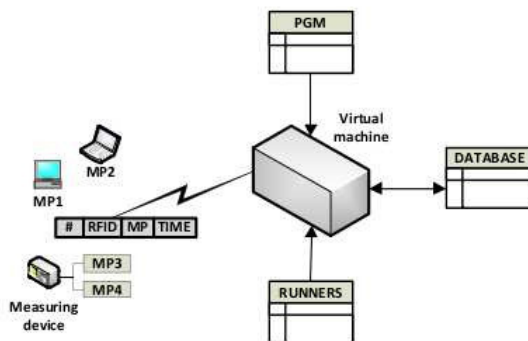


Fig. 3. Executable environment of a program in EasyTime.

In both modes of processing, the agent works with the program PGM, the runner table RUNNERS, and the results table DATABASE, as can be seen in Fig. 3. An initialization of the virtual machine is performed when the agent starts. The initialization consists of loading the program code from PGM. That is, the code is loaded only once. At the same time, the variables are initialized on starting values. A recording of events that are processed by the agent can be divided into the following phases:

- Reconstruction of the event: the competitor is identified by a starting number (#) or *RFID* tag, *MP* determines a virtual machine on which an interpretation of code will be run and the *TIME* represents the timestamp of the event.
- Reading of results: the number (#) or *RFID* tag determines the competitor whose results are read from the table RUNNERS in the database.
- Mapping of the result: the read results are mapped into the data segment of the virtual machine that is identified by the *MP*. In addition, the program register is loaded with the timestamp *TIME* of the event.
- Interpretation of code: the instruction counter is set to zero and the program loaded in the program segment of the virtual machine is started.

- Writing of results: after the interpretation of code, the results from the data segment are saved into the table DATABASE.

## VI. CONCLUSION

The flexibility of the measuring system is a crucial objective in the development of universal software for measuring time in sporting competitions. Therefore, the domain-specific language EasyTime was formally designed, which enables the quick adaptation of a measuring system to the new requests of different sporting competitions. Preparing the measuring system for a new sporting competition with EasyTime requires the following: changing a program's source code that controls the processing of an agent, compiling a source code and restarting the agent. Using EasyTime in the real-world had shown that when measuring times in a small sporting competitions, the organizers do not need to employ specialized and expensive companies any more. On the other hand, EasyTime can reduce the heavy configuration tasks of a measuring system for larger competitions as well. In this paper, we explained how the formal semantics of EasyTime are mapped into LISA specifications from which a compiler is automatically generated. Despite the fact that mapping is not difficult, it is not trivial either, as some additional rules must be defined for attribute propagation. Moreover, we need to take care of error reporting (e.g., multiple definitions of agents). In future work, EasyTime could be replaced by the domain-specific modeling language (DSML) that could additionally simplify the programming of a measuring system.

## REFERENCES

- [1] A.V. Aho and J.D. Ullman. *The theory of parsing, translation, and compiling*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1972.
- [2] The ChampionChip website, 2010.
- [3] K. Finkenzeller. *RFID Handbook*. John Wiley & Sons, Chichester, UK, 2010.
- [4] I. Jr. Fister, I. Fister, M. Mernik, and J. Brest. Design and implementation of domain-specific language Easytime. *Computer Languages, Systems & Structures*, 2011. Article in press.
- [5] D. Knuth. Semantics of context-free languages. *Mathematical Systems Theory*, 2(2):127–145, 1968.
- [6] T. Kosar, N. Oliveira, M. Mernik, M.J. Varanda Pereira, M. Črepinšek, D. da Cruz, and P.R. Henriques. Comparing general-purpose and domain-specific languages: An empirical study. *Computer Science and Information Systems*, 7(2):247–264, 2010.
- [7] M. Mernik, J. Heering, and A. Sloane. When and how to develop domain-specific languages. *ACM computing surveys*, 37:316–344, 2005.
- [8] M. Mernik, M. Lenič, E. Avdičaušević, and V. Žumer. Lisa: an interactive environment for programming language development. In N. Horspool, editor, *11th International Conference Compiler Construction*, volume 2304 of *Lecture Notes in Computer Science*, pages 1–4. Springer, 2002.
- [9] M. Mernik and V. Žumer. Incremental programming language development. *Computer Languages, Systems and Structures*, 31(1):1–16, 2005.
- [10] J. Paakki. Attribute grammar paradigms - a high-level methodology in language implementation. *ACM Computing Surveys*, 27(2):196–255, 1995.
- [11] L. Paulson. A semantics-directed compiler generator. In *Proceedings of the 9th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, POPL '82, pages 224–233, New York, NY, USA, 1982. ACM.
- [12] The RFID Timing System website, 2010.
- [13] A. van Deursen and P. Klint. Domain-specific language design requires feature descriptions. *Journal of Computing and Information Technology*, 10:1–17, 2002.



# Using Aspect-Oriented State Machines for Resolving Feature Interactions\*

Tom Dinkelaker

Software Technology Group  
Technische Universität Darmstadt, Darmstadt, Germany  
Email: dinkelaker@cs.tu-darmstadt.de

Mohammed Erradi

Networking & Distributed Systems Research Group, TIES, SIME Lab,  
ENSIAS, Mohamed V-Souissi University, Rabat, Morocco  
Email: erradi@ensias.ma

**Abstract**<sup>1</sup>—Composing different features in a software system may lead to conflicting situations. The presence of one feature may interfere with the correct functionality of another feature, resulting in an incorrect behavior of the system. In this work we present an approach to manage feature interactions. A formal model, using Finite State Machines (FSM) and Aspect-Oriented (AO) technology, is used to specify, detect and resolve features interactions. In fact aspects can resolve interactions by intercepting the events which causes troubleshoot. Also a Domain-Specific Language (DSL) was developed to handle Finite State Machines using a pattern matching technique.

## I. INTRODUCTION

**A**N important problem in modeling and programming languages is handling *Feature Interactions*. When composing different features in a software system, these may interact with each other. This can lead to a conflicting situation, where the presence of one feature may interfere the correct functionality of another feature, resulting in an incorrect behavior of the system. Various techniques have been explored to overcome this problem. Among them, formal approaches have received much attention as a means for detecting feature interactions in communication service specifications.

In Software Product-Line (SPL) engineering [1], [2], the designer decomposes a software system into functional features by creating a feature model [1], [3]. But a feature model can only define a set of features and known interactions between them. Feature models do not help, when the designer overlooks a feature interaction – especially at the implementation level.

Aspect-Oriented Programming (AOP) [4] uses a special kind of modules called aspects that supports localization of code from crosscutting features. AOP has been extended with special language concepts for controlling aspect interactions [5], [6], but AOP does not support controlling

feature interactions with modules that are not aspects in particular objects.

To address the above problems, in this work we propose a formal approach which uses an extension to finite state machines as the formalism for behavioral specification. The central idea behind using finite state machines as specification models is to have a strong mean to envision feature interactions. The formalism defines a process, which consists of the following steps: First, the developer gives a formal specification of each feature that extends the system's core feature, even partial specifications are allowed. Second, using the FSM's synchronized cross-product [7], the developer makes a parallel composition of the selected feature specifications and analyzes this composition. Third, the developer can identify conflicting states by analyzing the composed specification of the global system. Forth, to resolve feature interactions, the approach uses aspect-oriented state machines to intercept, prevent, and manipulate events that cause conflicts. We suggest a new formalism for aspect oriented state machines (AO-FSM) where pointcuts and advices are used to adopt Domain-Specific Language (DSL) [8] state machine artifacts. The advice defines a state and transition pattern that it applies at the selected points, i.e. it may insert new states and transitions as well as it may delete existing ones.

## II. PROBLEM DOMAIN: TELECOMMUNICATION SYSTEMS

### A. Plain Old Telephone Service (POTS)

Features in Telecommunication systems are packages providing services to subscribers. The Plain Old Telephone System (POTS) is considered as a feature providing basic means to set up a conversation between subscribers. In the following we provide the design and the specification of the basic service of a telephone system (POTS). We assume that a phone is identified by a unique number, and it can be either calling or being called.

In this specification, there are three objects that constitute the telephone system: the "user", the "agent" and the "call" as shown in Figure1. According to our semantics, the

\* This work was partially supported by: the EMERGENT project (01IC10S01N), Federal Ministry of Education and Research (BMBF), Germany, and the DAAD (German Academic Exchange Service) program.

instantiation of these objects provides three objects running in parallel. The communication between objects is based on operation calls using a rendezvous mechanism. Note that the behavior part of these objects is specified using a finite state machine model.

This system works as follows (Fig. 1): Once the caller (user-1) picks up (offhook) his phone (Agent-1), the network (designated by the object "call") responds by sending a tone. This user is then ready to dial the telephone number of the called party (using the operation "dial") using a standard telephone interface. Then the network sends back a signal (operation "Ring") which causes a ring on the called phone (Agent-2). An Echo\_ring is then sent to the caller (operation Echo\_ring). We assume that the called user is always ready to answer a call. When the called user picks up (offhook) his phone, the ring is then interrupted and the two users engage in a conversation.

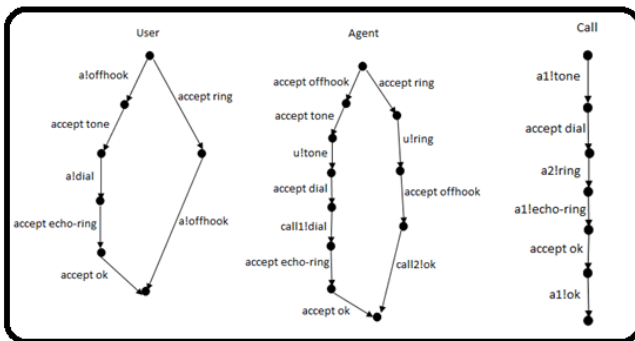


Fig. 1 Partial automata specifying the three objects

### B. Features available for User Selection (User Services)

According to the definition provided by Pamela Zave [9]: “in a software system, a feature is an increment of functionality, usually with a coherent purpose. If a system description is organized by features, then it probably takes the form  $B + F1 + F2 + F3 \dots$ , where  $B$  is a base description, each  $F_i$  is a feature module, and  $+$  denotes some feature-composition operation”. Therefore, telecommunication software systems have been designed in terms of features. So different customers can subscribe to the features they need. Many features can be enabled or disabled dynamically by their subscribers. Among the telecommunications features provided by a telephone system we found: Call Waiting, Three Way Calling, Call Forwarding, and Originating Call Screening.

#### 1) Call Waiting (CW)

A Call Waiting feature (CW) is a service added to the basic service POTS described earlier. It allows a subscriber A (having the service CW) already engaged in a communication with a user B to be informed if another user C tries to reach him. A can either ignore the call of C, or press a flash\_hook button to get connected to C. In other words, if C makes a call to A, while A is in communication

with B, then C receives an Echo\_ring, as if A was available, and A receives an “on hold” signal. Then A could switch between B and C by pressing the flash\_hook button. If B or C hangs up, then A will be in communication with the user still on line. The basic service POTS to which is added the Call Waiting feature is symbolically designated by POTS + CW.

A partial formal specification of *POTS+CW* is an FSM Fcw shown in Figure 2. The states  $Q_i$ , for  $i=1$  to 5, have the following semantics:

- $Q_1$  : A and B are connected and start communicating.
- $Q_2$  : A and B are communicating, then a call from C occurs on the switch of A.
- $Q_3$  : A and B are communicating, and A receives the signal *call-waiting* indicating that someone is calling.
- $Q_4$  : B is waiting, A and C are communicating.
- $Q_5$  : C is waiting, A and B are communicating.

The events  $E_i$ , for  $i=1$  to 4, have the following semantics.

- $E_1$  : a call from C arrived on the switch of A.
- $E_2$  : A receives the signal *call-waiting* indicating that someone else is calling.
- $E_3$  : A pushes the *flash\_hook* button.

#### 2) Three Way Calling (TWC<sup>2</sup>)

The Three Way Calling is a service which extends the basic service POTS. It allows three users A, B and C to communicate in the following way: Consider a subscriber A (having the TWC feature) who is communicating with B. A can then add C in the conversation. To reach this goal, A put first B on hold by pressing a button flash hook button. Then, establish a communication with C. And finally, press the flash hook button again, to get, A, B and C connected. A can remove C from the conversation by pressing the flash hook button. If A hangs up, B and C remain in communication. The basic service POTS to which is added the Three Way Calling feature is symbolically designated by POTS + TWC.

A partial formal specification of POTS+TWC is the FSM FTWC shown in Figure 3. The states  $R_i$ , for  $i=1$  to 4, have the following semantics:

- $R_1$  : A and B are communicating.
- $R_2$  : B is waiting.
- $R_3$  : B is waiting, A and C are communicating.
- $R_4$  : A, B and C are communicating.

The events  $E_i$ , for  $i=3$  and 4, have already been defined for the specification POTS+CW.

The event  $E_5$  has as its semantics :

- $E_5$  : A is communicating with C.

Note that the states “in bold”  $Q_1$  and  $R_1$  represent nested FSM. For instance this means that the state  $Q_1$  corresponds to an FSM which is a portion of the global specification, nested in this state  $Q_1$ .

<sup>2</sup> The abbreviation TWC for Three Way Calling should not be confused with trust-worthy computing.

### 3) Call Forwarding on Busy (CFB)

Call forwarding on busy is a feature on some telephone networks that allows an incoming call to a called party, which would be otherwise unavailable, to be redirected to another telephone number where the desired called party is situated.

### 4) Originating Call Screening (OCS)

The OCS Feature allows a user to define a list of subscribers hoping to screen outgoing calls made to any number in this screening list. A user A (with the OCS

feature) who registered user B on the list will no longer make a call to B, but B could call A.

### C. Feature Interactions

Feature interactions could be considered as all interactions that interfere with the desired operation of a feature and that may occur between a feature and its environment, including other features. Therefore, a feature interaction may refer to situations where a combination of different services behaves differently than expected.

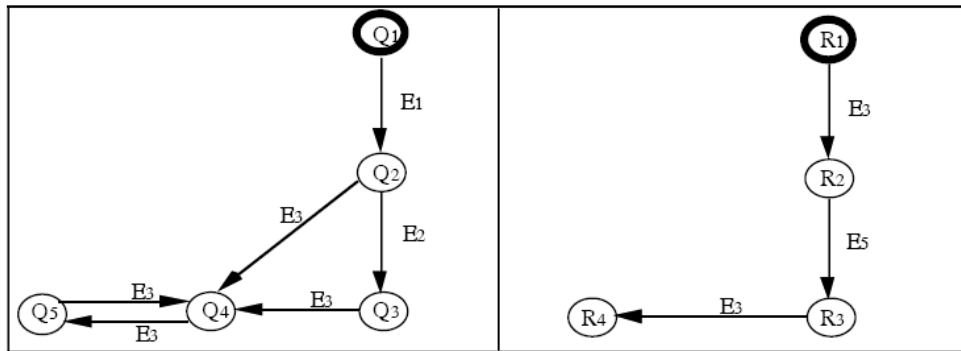


Fig. 2: Specification FCW of POTS+CW

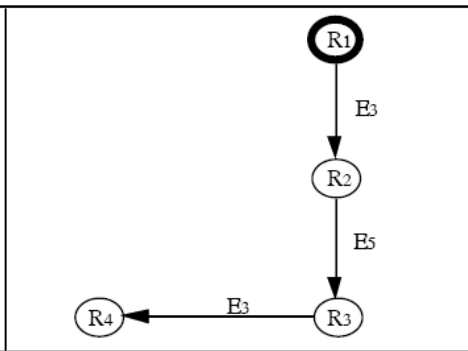


Fig. 3: Specification FTWC de POTS+TWC

For instance, pressing a “tap” button can mean different things depending on which feature is anticipated. This is the case of a flash-hook signal (generated by pressing such button) issued by a busy party could mean adding a third party to an established call (Three Way Calling) or to accept a connection attempt from a new caller while putting the current conversation on hold (Call Waiting). Should the flash hook be considered the response of Call Waiting, or an initiation signal for Three-Way Calling?

Another feature interaction may occur if we consider a situation where a user A has subscribed to the Originating Call Screening (OCS) feature and screens calls to user C. Suppose that a user B has activated the service Call Forwarding (CF) to user C. In this situation, if A calls B, the intention of OCS not to be connected to C will be violated since the call will be established to C by way of B.

Usually, the causes of interactions may be due to the violation of assumptions related to the feature functionality, to the lack of a technical support from the network, or to problems related to the distributed implementation of a feature. Despite the lack of a formal definition of a feature interaction due to the diversity of the interactions types, the reader will find a detailed taxonomy of the features interactions [10].

Our approach to process the feature interaction problem consists in two methods based on formal techniques. The first method is used to detect the interactions while the second resolves them. In the context of formal techniques, interactions are considered as “conflicting statements”. This may be a deadlock, a non-determinism, or constraints

violation which may result from states incompatibility between two interacting features. The incompatibility between states can be detected using a “Model-Checking” technique.

### III. PROBLEM STATEMENT

Feature interaction is considered a major obstacle to the introduction of new features and the provision of reliable services. In practical service development, however, the analysis of interactions has often been conducted in an ad hoc manner.

However, the feature interactions problem is not limited to the telecommunications domain. The phenomenon of undesirable interactions between components of a system can occur in any software system that is subject to changes. This is certainly the case for service-oriented architectures. First, we can observe that interaction is at the very basis of the web services concept. Web services need to interact, and useful web services will emerge from the interaction of more specialized services. Second, as the number of web services increases, their interactions will become more complex. Many of these interactions will be desirable, but others may be unexpected and undesirable, and we need to prevent their consequences from occurring.

Aspect-oriented programming (AOP) enables developers to modularize such non-functional concerns in OO languages. Important AOP concepts are pointcut, join point model, and advice. Pointcuts are predicates over program execution actions called join points. That is, a pointcut

defines a set of join points related by some property; a pointcut is said to be triggered or to match at a join point, if the join point is in that set. It is also common to speak about join points intercepted by a pointcut. Such a join-point model (JPM) characterizes the kinds of execution actions and the information about them exposed to pointcuts (e.g. a method call). An Advice is a piece of code associated with a pointcut, it is executed whenever the pointcut is triggered, thus implementing crosscutting functionality. There are three types of advice, **before**, **after**, and **around**; relating the execution of advice to that of the action that triggered the pointcut the advice is associated with. The code of an around advice may trigger the execution of the intercepted action by calling the special method **proceed**.

However, there is a lack of a general approach to weave on code fragments of DSLs. The problem is that current AOP tools support only one JPM at a time, which is for most aspect-oriented (AO) languages one JPM for the events in the execution of an OO language [4]. Only for some DSLs, there is a domain-specific aspect language with a domain-specific JPM [13] (e.g. encompassing join points like a state transition in a state machine). Still, current AOP tools do not provide support for special quantifications for weaving aspects into programs written in several languages that have different kinds of join-point models.

For example, consider implementing a logging feature as an aspect that needs to be woven into the code of several languages for debugging, such as it need to be woven into code in Java with an Aspect-like JPM, code in SDL<sup>3</sup> that defines a JPM for FSMs, and code in LOTOS<sup>4</sup> that defines a JPM on top of protocols as communicating processes.

#### IV. BEHAVIORAL MODELING OF FEATURES

This paper proposes to model software using models that defines details of the behavior of a system and each of its features. As elaborated in the following, the proposed formalism is based on finite state machines (Section IV.A). It defines the basic system in a behavioral model (Section IV.B) and it defines the behavior of features using aspects (Section IV.V).

##### A. Finite State Machines (FSMs)

An automaton with a set of states, and its “control” moves from state to state in response to external “inputs” is called a Finite State Machine (FSM). A Finite State Machine, provides the simplest model of a computing device. It has a central processor of finite capacity and it is based on the concept of state. It can also be given a formal mathematical definition. Finite State Machines are used for pattern matching in text editors, for compiler lexical analysis, for communication protocols specifications [16]. Another useful

notion is the notion of the non-deterministic automaton. We can prove that deterministic finite State Machine, DFSM, recognize the same class of languages as Non-Deterministic Finite State Machine (NDFSM), i.e. they are equivalent formalisms.

**Definition 1:** A non-deterministic Finite State Machine is defined by a quadruplet  $\langle Q, \Sigma, \delta, q_0 \rangle$  where  $Q$  is a set of states,  $\Sigma$  is an alphabet,  $\delta$  is the transition function, and  $q_0$  is the initial state. The transition function is  $\delta: Q \times \Sigma \rightarrow 2^Q$  where  $2^Q$  is the set of subsets of  $Q$ .

An event  $\sigma \in \Sigma$  is accepted out from a state  $q \in Q$  if the occurrence of  $\sigma$  is possible from the state  $q$ , i.e. if  $\delta(q, \sigma)$  is not empty, we denote this by  $\delta(q, \sigma)!$ .

When  $\delta(q, \sigma)$  is empty, we write  $\delta(q, \sigma) \neg!$ . We consider a blocking state  $q$  (deadlock) if no transition is possible from this state. Formally:  $q$  is blocking  $\Leftrightarrow \forall \sigma \in \Sigma, \delta(q, \sigma) \neg!$ .

**Definition 2:** A deterministic finite state machine is defined by a quadruplet  $\langle Q, \Sigma, \delta, q_0 \rangle$  and corresponds to a particular case of the non-deterministic finite state machine where for any  $q$  and for any event  $\sigma$ ,  $\delta(q, \sigma)$  is either the empty set or a singleton. When  $\delta(q, \sigma)$  is not empty,  $\delta(q, \sigma) = \{r\}$  will be simply noted  $\delta(q, \sigma) = r$ .

For all FSM  $A$ , the set of accepted traces will be designated by  $L_A$ .

**Definition 3:** Consider 2 FSMs  $A = \langle Q_A, \Sigma_A, \delta_A, q_{A0} \rangle$  and  $B = \langle Q_B, \Sigma_B, \delta_B, q_{B0} \rangle$  respectively accepting regular languages  $L_A$  and  $L_B$ , **the sum of A and B** is designated  $A \oplus B$  accepting the regular language  $L_A \cup L_B$ . Moreover, if  $A$  and  $B$  are deterministic then  $A \oplus B$  is also deterministic. Intuitively if  $A$  and  $B$  specifies 2 processes, then  $A \oplus B$  is the global specification of the two processes operating in an exclusive manner.

**Definition 4:** Consider 2 FSMs  $A = \langle Q_A, \Sigma_A, \delta_A, q_{A0} \rangle$  and  $B = \langle Q_B, \Sigma_B, \delta_B, q_{B0} \rangle$ . Let  $\Omega$  be a subset of  $\Sigma_A$  and  $\Sigma_B$ , in other words  $\Omega \subseteq \Sigma_A \cap \Sigma_B$ . **The Synchronized Product of A and B**, according to  $\Omega$ , is an FSM represented by  $A * B[\Omega] = \langle Q, \Sigma, \delta, q_0 \rangle$  defined formally as follows:

- $Q \subseteq Q_A \times Q_B, \Sigma = \Sigma_A \cup \Sigma_B, q_0 = (q_{A0}, q_{B0})$
- $\forall q = \langle q_A, q_B \rangle \in Q, \forall \sigma \in \Omega:$   
 $(\delta(q, \sigma)!) \Leftrightarrow (\delta_A(q_A, \sigma_A)!) \wedge \delta_B(q_B, \sigma_B)!$   
 $(\delta(q, \sigma)) \Rightarrow (\delta(q, \sigma)) = (\delta_A(q_A, \sigma) \times \delta_B(q_B, \sigma))$
- $\forall q = \langle q_A, q_B \rangle \in Q, \forall \sigma \notin \Omega:$   
 $(\delta(q, \sigma)!) \Leftrightarrow (\delta_A(q_A, \sigma_A)!) \vee \delta_B(q_B, \sigma_B)!$   
 $(\delta(q, \sigma)) \Rightarrow (\delta(q, \sigma)) = (\delta_A(q_A, \sigma) \times q_B) \cup (q_A \times \delta_B(q_B, \sigma))$

When  $\Omega$  is empty, two processes are said to be independent and their product is denoted  $A * B[\ ]$ . When  $\Omega = \Sigma_A \cap \Sigma_B$ , their product is denoted  $A * B$ . Intuitively, if  $A$  and  $B$  specifies 2

<sup>3</sup> SDL: Specification and Definition Language:  
<http://www.sdl-forum.org/SDL/index.htm>

<sup>4</sup> LOTOS: Language Of Temporal Ordering Specification:  
<http://language-of-temporal-ordering-specification.co.tv/>

processes, then  $A*B[\Omega]$  is the global specification of the 2 processes composed in parallel and have to synchronize on  $\Omega$ 's actions.

Note that  $A\otimes B[\Omega]$  is the product of the automaton A and B obtained by removing the blocking state from the *Synchronized Product*  $A*B[\Omega]$ .

**Definition 5:** (Sum of two FSMs, the Extension relationship)

Consider two FSMs  $A=(Q_A, \Sigma_A, \delta_A, q_{A0})$  and  $B=(Q_B, \Sigma_B, \delta_B, q_{B0})$  which accept respectively the regular languages  $L_A$  and  $L_B$ . The sum of A and B noted  $A\oplus B$  accepts the regular language  $L_A\cup L_B$ . In addition, if A and B are deterministic then  $A\oplus B$  is deterministic.

Intuitively, if A and B specify two processes, then  $A\oplus B$  is the global specification of the two processes behaving exclusively.

### B. Essential Behavioral Model (EBM)

The principle of our method for managing feature interactions, consists in three phases: the global behavior specification, the interactions detection and the interactions resolution. Interactions can be presented by states called *conflicting states*. This can be a *deadlock* (blocking) situation, a *non-determinism* or a *constraints violation* that is presented as an incompatibility between two states of features in interaction.

B.1. Global Behavior Specification: this phase consists in two steps:

**Step 1:** Specify formally each feature (involved in the interaction) with the basic system service (i.e. POTS in the case of a telecommunication system). This specification can possibly be partial.

**Step 2:** Make a parallel composition of the features, leading to a global behavior called an Essential Behavioral Model (EBM), to be analyzed. This implies making a synchronized automaton product (as shown in definition 4) of the behaviors of the composed features. The synchronization alphabet could be possibly empty.

### B.2. Interactions Detection:

Identify conflicting states by analyzing the EBM automaton produced in Step 2. Such states could be either a state where a given transition can lead to two distinct states (this is the case of non-determinism which is defined in definition 1), to a deadlock state (where one can execute no transition) or to a state constraints violation (i.e. a state

belonging to the product of two features specifications), and that results from two incompatible states). Formally, this violation means that two incompatible states allocate different "logical" values to the same variable.

The method of interaction resolution consists in three strategies. Recall that these strategies need to be applied during the specification phase. One among these strategies could be chosen and used depending on the type of interaction.

### B.3. Interactions Resolution:

**Strategy 1:** Make a composition using an exclusive choice of the two features specifications involved in an interaction. The designer could use existing merge algorithms [17] for LTS (Labeled Transition Systems) based specifications. Such algorithm produces a specification where its behavior extends the merged ones. The definition of the "extension" relation was given in Definition 5.

**Strategy 2:** Solve the interaction by making a precedence order upon the occurrence of certain events of the features in interaction. This allows a feature to hide some events from the other feature.

**Strategy 3:** Establish a protocol between features involved in an interaction. This protocol consists in exchanging the necessary information to avoid the interaction. This approach is more adapted in the case where the features are dedicated to be implemented on distant sites.

In the following we explain the suggested method in the case where an interaction occurs between the call waiting (CW) feature and the Three Way Calling (TWC) Feature specified in Section II.

Using simultaneously both services (Call waiting and Three Way Calling) is formally represented by  $F_{CW||TWC}$  which is the product (Definition 4) of two FSMs  $F_{CW}$  and  $F_{TWC}$ . In other words,  $F_{CW||TWC}=F_{CW}*F_{TWC}[\Omega]$  (Definition 4).  $\Omega$  is empty since here we consider the case (event  $E_3$ ) where pushing the *flash\_hook* button by A is considered by one among the provided features and not by both of them simultaneously. The states of  $F_{CW||TWC}$  will be designated by  $\langle Q_i, R_j \rangle$  where  $Q_i$  and  $R_j$  are respectively the states of  $F_{CW}$  and  $F_{TWC}$ .

The interaction (ambiguity) is detected by the presence of a non-determinism on the states  $\langle Q_i, R_j \rangle$  of  $F_{CW||TWC}$  where  $i=2,3,4,5$  and  $j=1,3,4$ . Intuitively, when he pushed the *flash\_hook* the subscriber A could not know if the signal is interpreted (executed) by the feature CW or by the TWC.

### V. Aspect-Oriented Finite State Machines (AO-FSMs)

In this paper, we propose a new formalism for aspect-oriented state machines (AO-FSM) which is based on finite-state machines and the Essential Behavioral Model. An AO-FSM defines a set of states and transitions like an FSM, but states and transitions do not need to be completely specified. Developers can selectively omit states, transitions, and labels, and therefore constitutes a partial FSM in which parts are missing so that it can be used as a pattern for matching against other FSMs and for manipulating them.

In an AO-FSM aspect, there are two parts: a *pointcut* and *advice* – like in other aspect-oriented languages for GPLs, but our pointcut and advice adapt DSL state machine artifacts. An AO-FSM pointcut defines a state and transition pattern that selects all FSMs that the advice adapts. The advice defines a state and transition pattern that it applies at the selected points, i.e. it may insert new states and transitions as well as it may delete existing ones.

Fig. 2 shows visual models of all types of AO-FSMs. The upper row enumerates all pointcut types (alphabetic indices), in which only the shown parts define the pattern and omitted parts match like wildcards. The lower row enumerates all advice types (roman indices), in which only the bold parts adapt the corresponding parts of a FSM. When constructing an AO-FSM aspect, the different types of pointcut and advice types can be composed.

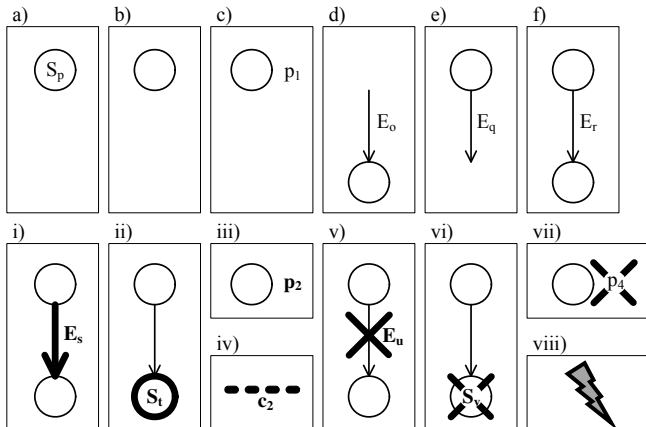


Fig. 2: Types of aspect-oriented finite state machines

There are 6 different kinds of pointcuts: a) matches a labeled state, b) matches any state, c) matches a state that meets a certain preposition, d) matches a state with an incoming transition, e) matches a state with an outgoing transition, and f) matches a sequence of two states with a transition.

The are 6 different kinds of advice: i) inserts a new transition for event  $E_s$ , ii) inserts a new state  $S_i$ , iii) adds a new preposition to a state, iv) defines a dependency constraint  $c_2$  between two states or two transitions, v) deletes the transition for event  $E_u$ , vi) deletes the state  $S_v$ , vii) deletes the property  $p_3$ , and finally, viii) defines a conflicting composition that results in an error message.

To weave an aspect, we match all pointcuts and apply all advice for all FSMs. For a single FSM, the pointcut matches at every point in the FSM and applies the advice at each of these points. The adapted FSMs are then used for execution.

### VI. RESOLVING FEATURE INTERACTIONS WITH AO-FSMs

To control feature interactions, developers uses aspects to analyze and manipulate the behavior of a system that they compose from a set of modular feature specifications. In a nutshell, when they compose specifications into an Essential Behavioral Model consisting of nested state machines, they uses AO-FSM aspects to detect interactions that manifest in singularities in the composed specification. There are three possible singularities: 1) the composed EBM is non-deterministic, 2) the composed EBM has contradicting prepositions, or 3) the composed EBM has blocking states. The main advantage of our approach is that feature interactions can be directly identified from the model. Finally, the developer can resolve feature interactions by eliminating singularities using AO-FSM aspect.

For example, there is a feature interaction when we compose the two feature specifications: Call Waiting (CW) and Three Way Calling (TWC). For instance, when A is in communication with B and A gets an incoming call from C, will the CW feature or the TWC feature be invoked?

To identify interactions, the system composes all models using the FSM synchronized cross-product operator of Definition 4, which corresponds to the parallel composition of the state machines of such specifications. It composes feature specifications with the core feature and the aspects.

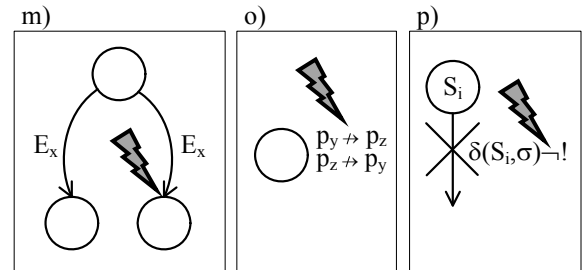


Fig. 3: Three detection aspects checking for composition singularities

When composing the aspects, a set of so-called *detection aspects* check the composition for possible conflicts. A detection aspect detects a singularity using a pointcut and its advice always declares a conflict, which makes the composition fail as long as the singularity is not corrected. Fig. 3 shows three analysis aspects that detect the three aforementioned singularities: m) matches any state if there are more than one transition with the same event  $E_x$ , o) matches any state with contradicting prepositions  $p_y$  and  $p_z$ , and p) matches every blocking state  $S_i$  for which there is no outgoing transition. When necessary, developers can define their own detection aspects. Whenever one of the detection aspects' pointcuts matches in a composed system, its advice will report a conflict.

Detection aspects are in particular useful when composing many models and aspects that manipulate those models. Detecting composition singularities prevents any further incorrect processing of the system in a potentially undefined state. The above three detection aspects help automatically detecting the most important composition singularities. Therefore, the developer does no longer have to worry about them. Similar to related work on aspects interaction [5], [16], automatic feature interaction detection is enabled. However, automatic feature conflict resolution is not possible [5].

To resolve the conflict, the developer need to specific a set of resolution aspects. Each aspect intercepts the reception of events, and removes a singularity (e.g. non-determinism) from the composed specification. Depending on corresponding context (e.g. the current state and the received events), the aspect can make a choice which of the conflicting features should be active and which not.

A resolution aspect defines a pointcut and advice for the corresponding conflict resolution, which may have been detected using a detection aspect. Its pointcut matches the conflict situation. Further, its advice declares what states and transitions to remove from the composition such that it becomes deterministic.

For example, consider the feature interaction between CW and TWC. First, the detection aspect in Fig. 3 at index  $m$  identifies this non-determinism singularity. Second, the developer specifies the resolution aspect in Fig. 4. That resolution aspect resolves the interaction of the CW and TWC features by defining a precedence between those features that depends on the sequence of previous events. Intuitively, if a call of C arrives on agent A (event  $E_1$ ) before A presses the flash\_back button (event  $E_3$ ), the CW feature will be active. In this case, the left pointcut in Fig. 4 will match and temporarily remove the transition  $TWC.E_3$ . Conversely, if  $E_3$  takes place before  $E_1$ , then the TWC feature will be active. In this case, the right pointcut in Fig. 4 will match and temporarily remove the transition  $CW.E_3$ .

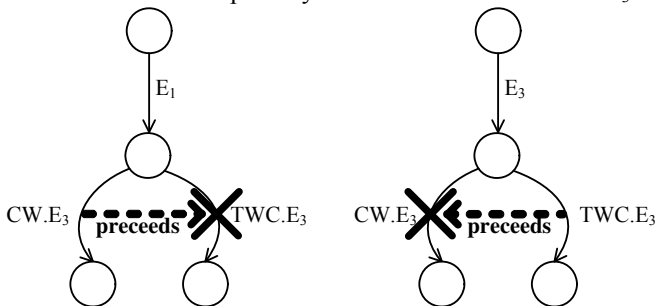


Fig. 4: A resolution aspect that resolves the  $CW \leftrightarrow TWC$  interaction from Section IV.B

## VII. DISCUSSION

To validate the approach, we have implemented a prototype of AO-FMS in the Groovy language [18] using the POPART framework [17] that allows embedding DSLs and developing aspect-oriented extensions for those DSLs in

form of plug-ins. Further, we have implemented the examples presented in [7] and which were used as a running example in this paper as a case study. As a proof of concept, the AO-FSM prototype automatically detects the interaction from Section IV.B, and we have developed a resolution aspect to revolve this interaction. We could achieve objectives stated in the introduction, namely the support for separation of concerns (in particular crosscutting features), the formalization of behavior, and dealing with interactions. With the current prototype, conflicts can successfully be detected and resolved. However, correct results depend on whether the developer completely specifies the model and correctly implements aspects with the AO-FSM tool.

Furthermore, at the current stage, we cannot draw universally valid conclusions from the case study. A larger case would be more convincing. At the end, only a formalization proof of the formalism in a proof assistant (like Isabelle or Coq) would give absolute guarantees.

Our prototype implementation only covers feature detection and resolution at design time. For save feature implementation, our approach could easily integrate with a code generator from state machines to C or Java code.

Various practicable limitations need to be addressed by future work, the expressiveness of model is confined by state machines and therefore systems whose behavior can be formalized as a regular language. The approach could be extended for models with richer semantics, which consequently would make it more complicated. Because we build the synchronized product of FSMs, the approach suffers from the well-known state explosion problem when using FSMs for modeling. Therefore, the prototype can only be used to analyze small models. In future work, we want to reduce synchronized products by finding equivalent states. Another limitation is that it currently does not nicely integrate with standard modeling notations, such as UML. In future work, we would like to support for importing UML state charts and let the developer enhance them to EBMs.

## VIII. RELATED WORK

Most similar is the work in the field of FOP, AO modeling, and model driven development.

FOP [11] provides language support for implementing modular features that encapsulate basic functionality. Similar to FOP, our EBM and AO-FSM allow modular specification of features. While FOP uses so called *lifters* for inheriting features into a composition, we build on the sum for inheriting FSMs and the synchronized product for composing them. While FOP is for implementation, we focus on the specification of features. FOP allows defining known interactions. In contrast, EBM and AO-FSM allow automatic detecting of interactions that the developer is not aware of.

Aspect-oriented modeling has come up with various modeling notations into which aspects are woven. There are AO state machines [13] and other AO models available.



However, they have been little explored in the context of detecting feature interactions in behavioral models. They can only detect conflicts involving aspects, but they cannot detect interactions between base features as we do.

Model-driven development proposes various kinds of models – not only FSMs. Life-Sequence Charts [19] are similar to AO-FSM. Such models are often used for code generation. While standard model notations do not adequately consider interactions, there are a few special models that allow expressions such constraints for a restricted set of domains, such as telecommunications for which special DSLs are available. Currently, developers are left alone to encode constraints on the modeled feature using constraint languages for which often there is no complete support for code generation. In contrast to this, possible domains for EBM and AO-FSM are not limited.

## IX. CONCLUSION

In this paper, we suggested a formal approach to detect and resolve feature interactions within a distributed software system. The approach is based on a new formalism for aspect-oriented state machines (AO-FSM) based on finite-state machines and an Essential Behavioral Model (EBM). The EBM defines states and transitions as an FSM, but states and transitions do not need to be completely specified.

A specific mechanism for interactions detection and a strategy for feature interaction resolution were presented. The implementation of this mechanism and its associated strategy were made using the AO-FSM formalism. Therefore, the pointcut defines a state and transition pattern that selects all FSMs that the advice adapts, while the advice defines a state and transition pattern that it applies at the selected points. In fact, the approach uses aspect-oriented state machines to intercept, prevent, and manipulate events that cause conflicts.

## ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments. The authors would also like to thank Yassine Essadraoui who has contributed to the implementation of the prototype of AO-FSM and the telephone case study as part of his Master's thesis.

## REFERENCES

- [1] Clements, P. and Northrop, L., "Software product lines", Addison-Wesley, 2001.
- [2] Pohl, K. and Böckle, G. and Van Der Linden, F., "Software product line engineering: foundations, principles, and techniques", Springer-Verlag New York Inc, 2005.
- [3] K. Czarnecki and A. Wasowski. "Feature diagrams and logics: There and back again" in Proc. 11th Int. Software Product Line Conference (SPLC 2007), Washington, DC, USA, 2007, pp. 23–34.
- [4] Kiczales, G. and Lamping, J. and Mendhekar, A. and Maeda, C. and Lopes, C. and Loingtier, J.M. and Irwin, J.: "Aspect-oriented programming" in Proc. Europ. Conf. on Object-Oriented Programming, Springer, 1997, pp. 220–242.
- [5] G. Kniesel, "Detection and Resolution of Weaving Interactions. TAOSD: Dependencies and Interactions with Aspects", In Transactions on Aspect-Oriented Software Development V, pp. 135–186, LNCS, vol. 5490, Springer Berlin / Heidelberg, 2009.
- [6] Tanter, E., "Aspects of composition in the Reflex AOP kernel", Software Composition, Springer, 2006, pp. 98–113.
- [7] M. Erradi and A. Khoumsi, "Une approche pour le traitement des interactions de fonctionnalités des systèmes téléphoniques", in Proc. Colloque Francophone International sur l'Ingénierie des Protocoles (CFIP'95), Rennes, France, 1995.
- [8] M. Mernik, J. Heering, and A.M. Sloane, "When and how to develop Domain-Specific Languages" ACM Computing Surveys (CSUR), vol. 37, no. 4, 2005, pp. 316–344.
- [9] Pamela Zave, "Feature Interaction", <http://www2.research.att.com/~pamela/fi.html>
- [10] E.J. Cameron, N.D. Griffeth, Y.-J. Lin, M. Nilson, W.K. Schnure, et H. Vlethuijsen. "A feature Interaction Benchmark for IN and beyond", Feature Interactions in Telecommunications Systems, Eds. L.G. Bouma and H. Velthuijsen, IOS Press, Amsterdam, 1994.
- [11] Prehofer, C.: "Feature-oriented programming: A fresh look at objects" in Proc. ECOOP, Springer, 1997, pp.419–443.
- [12] Parnas, D.L., "On the criteria to be used in decomposing systems into modules", *Communications of the ACM*, vol. 15, no. 12, 1972, pp. 1053–1058.
- [13] M. Mahoney, T. Elrad, "A Pattern Story for Aspect-Oriented State Machines", LNCS, Vol. 5770, 2009.
- [14] G. v. Bochmann, "Finite State Description of Communication Protocols", *Computer Networks*, Vol. 2 (1978), pp. 361-372.
- [15] F. Khendek and G. v. Bochmann, "Merging Behavior specifications", Proc. FORTE'1993, Boston, USA.
- [16] W. Havinga, I. Nagy, L. Bergmans, M. Aksit, "A graph-based approach to modeling and detecting composition conflicts related to introductions". In Proc. International Conference on Aspect-Oriented Software Development, ACM, 2007.
- [17] T. Dinkelaker, M. Eichberg, and M. Mezini, "An Architecture for Composing Embedded Domain-Specific Languages". In Proc. Aspect-Oriented Software Development ACM New York, 2010.
- [18] D. König, A. Glover, "Groovy in Action". Manning, 2007.
- [19] W. Damm and D. Harel. LSCs: Breathing Life into Message Sequence Charts. *Formal Methods in System Design*, vol. 19, no. 1, pp. 45–80, 2001.

## Domain-Specific Modeling in Document Engineering

Verislav Djukić  
Djukić – Software Solutions  
Nürnberg, Germany  
+49 (0)911 4313-686  
info@dvdocgen.com

Ivan Luković  
University of Novi Sad  
Faculty of Technical Sciences  
Novi Sad, Serbia  
+381 (0)21 4852-445  
ivan@uns.ac.rs

Aleksandar Popović  
University of Montenegro  
Faculty of Sciences,  
Podgorica, Montenegro  
aleksandarp@rc.pmf.ac.me

□ **Abstract**— Specification languages play a central role in supporting document engineering. We describe in this paper how domain-specific languages, along with domain-specific frameworks and generators, can support formal specification and document rendering in directory publishing. With flexible metamodel-based tools we have developed four languages for the modeling of: (i) small advertisements, (ii) appropriate documents, (iii) workflow control and (iv) layout patterns. The paper provides a more detailed description of the first and the third language, including a brief account of the language interpreter, as well as code, document and application generators. The presented approach enables, in a typical document-centric system, specification of both static and dynamic characteristics of the system on a high abstraction level with domain-specific concepts. The concepts of incremental document specification and incremental document rendering have been introduced, in order to address the problem of very frequent specification(s) refinements. The expression power of the created languages is demonstrated with a representative examples of document engineering covering document content specification, workflow control and application generation. All of the aforementioned languages are integrated into a single meta-model, under the name of DVDocLang which is, due to its simplicity, highly applicable for user-driven conceptual modeling.

### I. INTRODUCTION

Document engineering (DocEng) represents a scientific discipline which attempts to unify different types of analyses and modeling perspectives, in order to aid various specification, design and document implementation activities and all the processes which, both, create and consume them [2]. Formal document specification and rendering, as a part of document engineering, comprise a research area in which recent years two distinct directions have become prominent: (i) the first, mostly presented in academic work adhere to general approaches and solutions based on XML languages and (ii) the second, emerging due to the need for the production of large amounts of valid documents, is characterized by adherence to the complex layout document

rules in specific business domain. These different directions aim to solve document engineering work in two ways. One side attempts to describe domain-specific problems by the using General Purpose Languages (GPL) which has a rather negative direct consequence creating the need for the development of a multitude of applications, concentrated on either one single problem or on a class of similar problems. Others attempt to develop custom and complete frameworks which would, to the greatest extent, simplify and automate the document production process as well as significantly improve their overall layout quality. This dichotomy, existing between the approaches based on GPLs and those based on domain-specific ones, is not only present in the area of document engineering, but in software engineering in general ([14],[15]). This topic is receiving more and more attention in the academic community, with the methodology tools necessary for solving the problematic aspects being intensively (constantly) developed. In this paper, the authors present their substantial long-term experience in the area of Domain-Specific Modeling (DSM) [1], with the emphasis on the development of the domain-specific framework. The examples chosen for the illustration of DSM in document engineering [2], have been acquired from the directory publishing, and applied to the formal specification and visualization of the documents. The problems being solved are essentially diverse in nature, and are in direct relation to a number of software engineering domains such as: construction of formal languages, Domain-Specific Languages (DSL), conceptual modeling, form-based analysis (FBA) [3], user-driven modeling (UDM), model-driven architecture (MDA) [16], service-oriented architecture (SOA), rendering of the PDF and HTML documents and generation of web applications. The first part of the paper describes the domain and the domain-specific languages (DSL), together with the analysis of the practical benefits of their usage. The second part of the paper provides a brief theoretical overview on incremental specification, and rendering of documents and applications. Accordingly, the paper is divided into seven sections: next Section introduces the domain and provides a substantial example of a small advertisements modeling language DVDocAd. This language is constructed for the purpose of expressing topological and semantic relations between the content units (CU) of small advertisements [9]. Section three describes the workflow control language DVDocFlow as well as the underlying principle for modeling the business activities. Section four

□ A part of the research presented in this paper was supported by Ministry of Education and Science of Republic of Serbia, Grant III-44010, Title: Intelligent Systems for Software Product Development and Business Support based on Models.

describes the notion of incremental specification, which can be regarded as the greatest contribution to DSM in document engineering. Section five describes the usage of the meta-models and the repository to generate applications. Section six provides an explanation of the domain-oriented libraries. We conclude by describing our experiences of the practical value of the presented approach as well as the plans for a further course of action.

## II. MODELING OF SMALL ADVERTISEMENTS

By using meta-concepts, as GOPRR (Graph, Object, Property, Role and Relationship) employed in MetaEdit+ language workbench [4], any domain-specific language can be constructed. We describe here one particular language that is constructed towards the most important characteristics of small advertisements (“small ads” for short). They include the following:

- Small ad represents a collection of semantically based content units (CU), predominantly textual;
- Optionally, small ad contains a picture, i.e. a logo;
- Small ad always contains at least one single name, either the name of the company or the name of the person;
- Elementary-type content units are mostly limited to a name, address, telephone number, E-mail, web address and a logo;
- When displaying content units emphasis is placed on topological relations, primarily on their sequence;
- Telephone number is always placed in the top right corner, in continuation of a name or an address;
- In the course of element formatting, splitting of the text for specific content types is not allowed;
- Alignment can be left, right, centered or combined (applicable to each content unit separately);
- Specific content type can be assigned the leading role in a logical unit, i.e. implicitly defined complex CU;
- Small ad can be comprised of a number of logical units.

The aforementioned characteristics can be described by different languages. In this case, 2D graphics and the following meta-concepts presented in Fig. 1 are employed:

- Object of the type “picture” (logo);
- Object of the type “textual contents” including subtypes: “name”, “location”, “telephone number”, “street address and number”, “E-mail” and “web address” (represented by rectangles containing text within);
- Relations: “content line” (represented by an ellipsis with three dots), “telephone connection” (represented by a telephone symbol) and “content unit” (filled in circle);
- Roles: “is a part of content unit”, “leading in line”, “successor in line”, “telephones in” and “tel. rings in”.
- Properties: ad height and width, logotype height, font and text colour, alignment and leading symbol.

By employing MetaEdit+, within just one hour for middle experience user, a new domain language has been

constructed and an example, presented in the Fig. 1, is modeled. The tool provides graphical editors for the creation of ad instances and checking mechanisms to validation the specifications through given set of language rules. Such an approach, to ad modeling, is inherently different from the process of drawing in general-usage graphic tools. Hence, what is achieved this way is that each advertisement becomes a “pentaformat” document [5]. It means that the document is viewed as a 5D entity: content, structure, layout, meta-data and behavior.

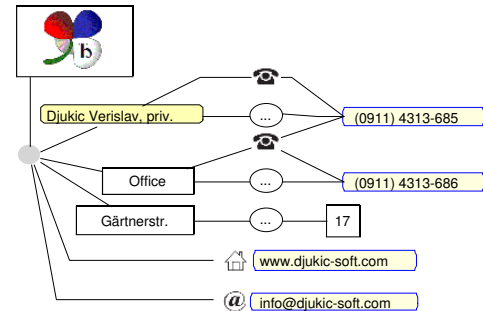


Fig. 1 Model of simple advertisement

The ad from Fig. 1 does not represent the final image a user requires. It is not usual for relations and roles of ad elements to be displayed explicitly. Instead, in the course of rendering, the layout rules are interpreted in the way in which the elements will unambiguously point to the type of content and relation by their position, font, colour and alignment. MetaEdit+ as a modeling tool was not primarily employed here for the purpose of drawing small ads, but to specify domain knowledge in a form of a DSL so as to provide a validation of the created specifications and a generation of special-purpose applications. Ad-production interface can then completely rely on the rules of DVDocAd language. Due to practical reasons, and for the purpose of accelerating the modeling process, a textual language equivalent - DVDocLang [6], has been created. Moreover, the language in question can be integrated into arbitrary framework much easier, and is quite suitable as a basic interface for the end user. Transformation to and from the graphic language is possible. We denote a statement created by DVDocLang syntax, specifying the content, structure, layout and behavior of the document as “logical script”.

**Example 1.** A logical script that specifies content of the advertisement from Fig. 1, is specified as follows:

```
<LOGO>7937,20
<NA>Djukic Verislav, priv.<PH>(0911)4313685
<NA>Office<PH>(0911)4313686
<ST>Gärtnerstr.<HN>17
<EM>info@djukic-soft.com
<IN>www.djukic-soft.com □
```

Knowledge about objects, relations, roles and layout is, due to practical reasons, singled out and regarded as a part of the document type definition. It is comprised of global attributes (elements, logo\_enabled, logical\_fonts, width, height, logo\_params) and content unit attributes (POS,

GROUP, SYMBOL, FORMAT, FONT, LOG\_FONT, ID and ROLE) [6], as demonstrated in the following example.

**Example 2:** A logical script that specifies layout and type definition of the advertisement from Fig. 1:

```
<TEMPLATE>=elements:(NA;Name),(ST;Street),
(HN;HouseNr),(PH;Phone),(IN;Inet),(EM;eMail)
<TEMPLATE>=logical_fonts:(s1;Tahoma,Bold;2;97;2),
(s2;Tahoma;2;97;2)
<TEMPLATE>=logo_enabled:true
<TEMPLATE>=width:40
<TEMPLATE>=height:50
<TEMPLATE>=logo_params:7937,center,20,false
<NA>=POS:new_line,GROUP:true,LOG_FONT:s1
<PH>=POS:con,GROUP:true,FORMAT:'2n|3n',...
<ST>=POS:new_line,GROUP:true
<HN>=POS:con,GROUP:true,LOG_FONT:s1
<IN>=SYMBOL:(SymbolFont;&H29),POS:new_line,
GROUP:false,LOG_FONT:s2
<EM>=SYMBOL:(SymbolFont;&H40),POS:new_line,
GROUP:false,LOG_FONT:s2
```

For the two, grouped (GROUP:true), content-unit types differing in position (POS:new\_line and POS:con) on a level of appropriate instances, establishment of a relation “content line” with the following roles – “line leader” and “adherent in line” is permitted.

This way, the modeling of small ads is reduced to a description of the structure and the content, based on a fairly simple syntax. It is intuitively acceptable, and in agreement with the expectations of the producer of small ads<sup>1</sup>. Fig. 2 shows previously specified ad, generated through DVDocRender. This is a domain-specific document renderer that interprets DVDocLang specifications and produces PDF or HTML specifications, as well as images.



Fig. 2 Small ad generated by DVDocRender

In agreement with the DSM approach, the first step was the construction of small ad modeling language. Subsequent steps encompass the process of making or integration of domain-specific libraries, used for interpreting and rendering of advertisement documents. Grammar rules and templates are generated with the assistance of MERL [4], the reporting language of MetaEdit+. This report serves as an entry point for automatic parser generation. Such instance and type specifications of small ads are sufficient for generating other specification varieties, e.g. in SGML, XSL-FO, HTML, DVDocLang and other formats. Furthermore, such a description of relations between objects is sufficient for

pattern construction which is employed for advertisement structure validation, i.e. validation of documents [7]. Capabilities of the document generator were discussed in detail in the comparative analysis of the concepts and characteristics of XSL-FO and DVDocLang provided in [8]. In the case of the ad from Fig. 2, our document generator is even more than forty times faster [10] than the FOP renderer [13] accepting XSL-FO specifications as the input.

### III. WORKFLOW CONTROL LANGUAGE

The fact is that a number of renowned workflow control systems, such as Bonita, JWT, Alfresco et.al. exist. Still, they do not provide a solution to the problem of parallel refinement of the document layout and business process models. Incremental specification by means of a simple DSL unifying both activities, together with the incremental rendering, is seen as a prerequisite for the building of the software system in which the progress in each of the activities is documented by a valid document instance. This is the exact reason why we have constructed a new language, by employing MetaEdit+. We describe in this section DVDocFlow – a specific language used for workflow control as well as for synchronization of parallel activities. It is employed for the description of the dynamic characteristics of the system, i.e. document behavior. This represents one of the dimensions of the previously mentioned “pentaformat”. Typically the production of small ads is not insular, but is a part of more complex activities such as sending of offers, error correction in ads and conclusion of the advertising contracts. The analysis of the documents, in which small ads represent certain content units, pointed out to two important facts: (i) the state of an ad affects the state of the document it is a part of and (ii) collection of potential states of the documents depends on the overall state of all content units, and transitional rules. This collection of states and transitions, which we refer to as the document life cycle, varies depending on the concrete production model. In addition, it is indirectly determined by the degree of automation of the production process. Accordingly, our main goals in the course of constructing a new language is that it should provide a solid foundation for the description of different production models as well as that the documents which did not yet reach their end state can “respond” to the change of a production model. It is not necessary for a concrete system to possess a workflow engine (WFE). Knowledge of the life cycle, built in each document instance, can completely replace the WFE. Incremental specification and rendering are viewed as a domain-specific solution, used for realization of the specification and tracing of the document behavior. It is presented in Section 5 in more detail.

In Fig. 3 we present an example of an offer-production model, specified in DVDocFlow. In the course of language creation process for DVDocFlow the same meta-concepts

<sup>1</sup> Successfully applied in production of ads in ten countries.

(GOPRR) have been used, as in the case of DVDocAd. The DVDocFlow meta-concepts and symbols are:

Objects:

- Document states:
- Activities:
- Content units (Complex, Logo):
- Template:
- Document generator:

Relations and included roles:

- Increment of a state: (“Sets into a specific state”, “Comes with a content unit”);
- Layout definition: (“Preferred for document”, “Document contains content unit”)
- Pattern based: (“Template uses a pattern”, “Pattern for document templates”);
- Activity states: (“State from an activity”, “Sends to state”, “Visual repres. of progress in PDF format”); and
- Synchronization action: (“Waits an activity to

end”, “Request accepted”).

The overall number of concepts and properties available in DVDocFlow is much greater, but we explained here just those ones necessary to introduce our production model in brief. Aside from the explicitly described concepts, for modeling and rendering documents the following is also important:

- Content units, their layout included, are linked to a specific role in a real system;
- In each state, except for the final, a specification increment can be joined to a document instance. It can alter the life cycle independently from the previously defined one, for the type instance is a part of;
- All the changes, in a real system, are documented by concrete document instances in PDF, that can be generated for any state of activity;
- From the specification of dynamic characteristics, acquired from modeling tools, a code, which manages document transitions, is generated;
- Relying on modeling infrastructure (repository, editors, code generation language and API) application prototypes are generated;

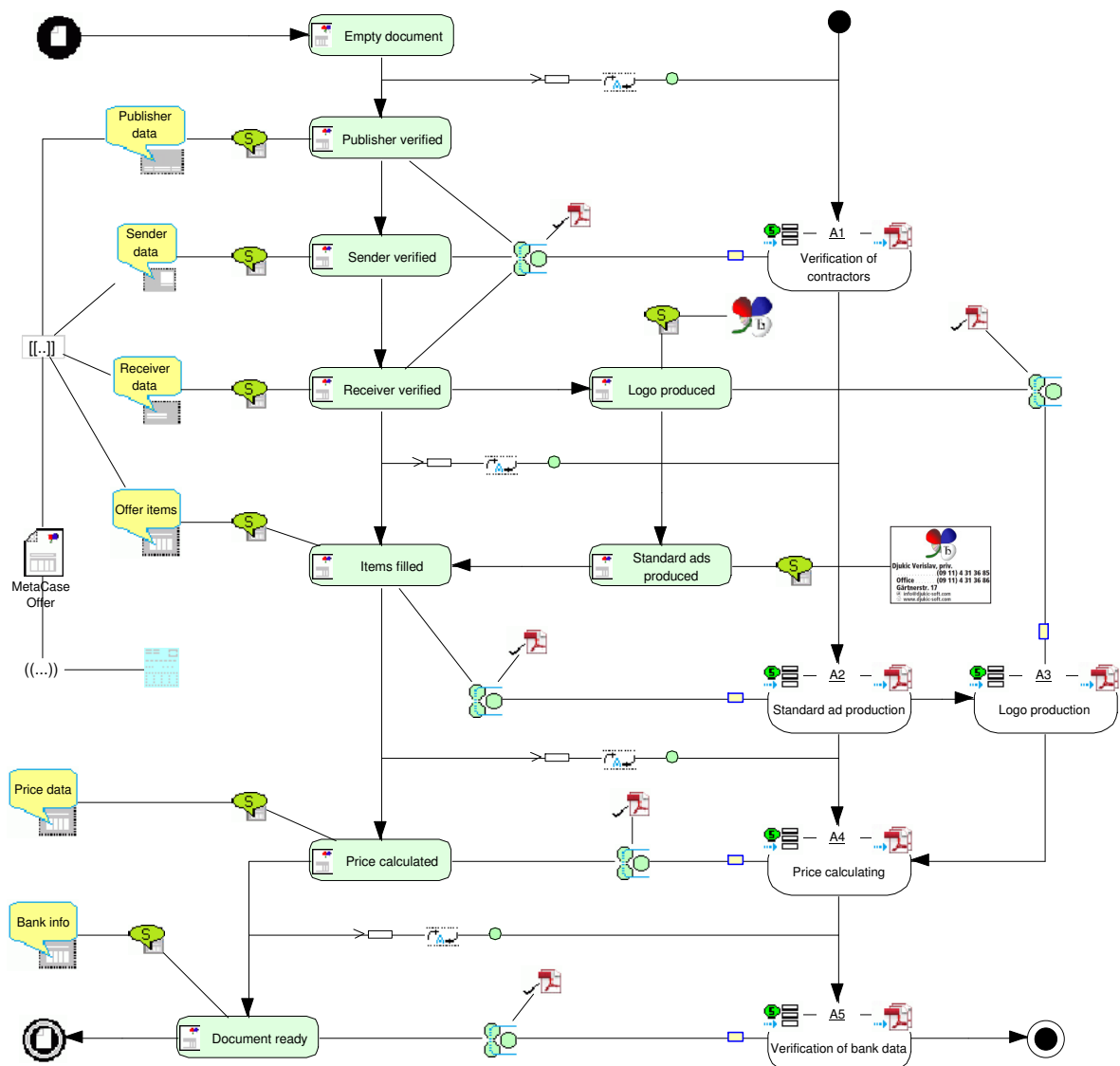


Fig. 3 Production model for "Offer"



- There exists a domain-specific editor that provides specification of document types by drawing typical instances and using available patterns for different types of content units;
- UML activity and state diagrams are suitable for specification of the document activities and states. Thus, in DVDocFlow language, similar symbols are used for the activities, states and transitions; and
- The main purpose of DVDocFlow language is to provide modeling relations between the states of the documents, content units, layout forms and activities in which the appropriate content units are created.

Formal specification of a production model in DVDocLang, expressed by the concepts which integrate activities, states, layouts and behavior of the documents, is an extension of incrementally oriented logical script, as shown in Examples 1 and 2.

**Example 3:** A basic form of logical script for expressing production model of document type “Offer”:

```

<STATE>Empty document
<CU>Script from validation of sender
<STATE>Sender verified
<CU>Script from validation of receiver
<STATE>Receiver verified
<CU>Script from validation of publisher
<STATE>Publisher verified
<CU>Script from logo production
<STATE>Logo produced
<CU>Script for simple ads,i.e. (1) and (2)
<STATE>Standard ads produced
<STATE>Items filled
<CU>Script from price calculating
<STATE>Price calculated
<CU>Script from verification of bank data
<STATE>Document ready □
    
```

Each element of a logical script, marked with tag <CU>, is either a simple value or a complex content unit. In the case of a complex content unit, it is related to a number of different document states. Recipient, publisher and sender data, displayed in Fig. 3, can be represented by one composite content unit. Commands, in the form of <STATE>Name, are a part of a logical script, related content units to the states of a document.

Further characteristics of such conceptualized workflow control language include the following:

- General approach in the specification of a document layout represents one simple case in which all of the <CU> are known prior to the beginning of rendering;
- The content of the expression with <CU> is instance-related. The definition of content unit type, which includes the layout properties, can precede an instance and has the form <CU>=Definition of CU\_type. This implies that the layout can be redefined while it is in a non-terminal state, depending on a certain activity;
- Expressions - in the form of <CU>=Definition of CU\_type, also define a collection of new and potential states

since they alter the layout, regardless of the fact that the content remains unchanged;

- Definition of the content unit type is rich enough to enable PDF and HTML generation, as well as that of a web application;
- Work progress is documented by a valid, legible content unit or document instance;
- In case meta-data is placed in the form of annotations in a PDF or meta-properties in HTML, then each instance also describes completely the type it belongs to, and can, thus, be cloned and inherited. Viewed from the standpoint of optimal refinement of layout-definition, cloning and inheriting are highly important; and
- Separation of the content from a layout definition enables for a specific component to acquire layout definition instances, for the purpose of their merging in the course of rendering process.

Fig. 4 shows a layout of a document's instance, in four different states. From the standpoint of the user who produces/creates documents, this specific manner of business process progress reporting is, by far, most acceptable. State represented by (1) refers to the confirmed contractor data, (2) readymade small ads and filled in list, (3) calculated cost and (4) verified banking data. Presented example refers to a rather simplified case – when the layout of content units displayed, remains unchanged regardless of the states.

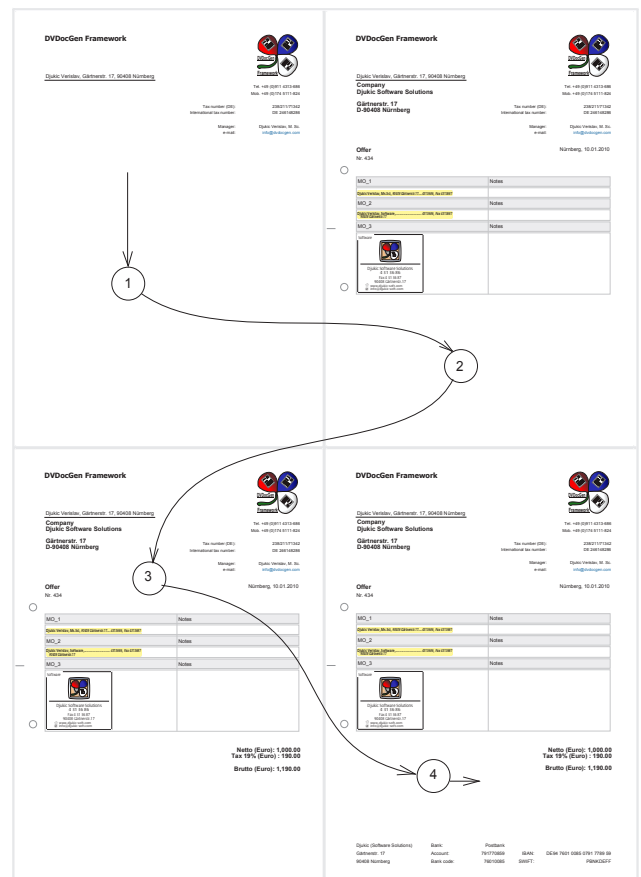


Fig. 4 Document rendering for each of the different states

#### IV. INCREMENTAL SPECIFICATION AND RENDERING

The emphasis in the workflow control language was placed on the procedure that enables the synchronization of activities in a real system as well as on reporting work-progress. In document engineering we also need to specify document production models, rendering processes as well as their complete implementation based on an incremental approach. In the broadest sense, incremental specification and rendering are regarded as document engineering approaches, allowing content, structure, layout and behavior related changes (or additions) of a document being in an arbitrary state. Aside from substantial improvements regarding the rendering speed, our approach significantly simplifies the entire document production process and application generation necessary for the automation of production. By means of an increment specification, whose primary source is the activity producing the content unit, the approach makes the controlled and automated document-knowledge refinement possible. Such a refinement simplifies the production of untypical document instances. Those are the instances somehow differing from the previously defined type or that cannot be created by the existing applications. The notion of a "modifier" denotes a knowledge increment. It can be named, unnamed or unresolvable. A named modifier consists of a group of attributes recognized in advance as a possible type variation. It is unnamed if it does not belong to any previously defined variation. It is unresolvable if there is no any language concept suitable for a full specification of the document instance.

The core elements of the proposed incremental specification, illustrated in Fig. 5, are the following:

- Each content unit or each combination of content units matches at least one specific document state. Fig. 5 displays the potential document states (Initial, S1, S2, Final\_1, Final\_2) for a document „D”. Marked with D1-D5 are visual representations of the documents, in specific states. In an initial state a document is empty.
- At least one layout specification is associated to each content unit, as well as a set of known layout variations (a visual pattern with its variations). The visual pattern is a pattern necessary for the representation of a particular type of the content unit. It is referred from the logical script and easily customized (contextualized) by referring to its variations. The list of content units is placed separately, in the upper right corner of Fig. 5. The content units are marked with CU1-CU4.
- The increment named as SpcIncr in Fig. 5, as well as a complete document, is specified by the domain-specific language DVDocLang. It is possible to define an increment in advance, so the document would ‘carry’ it from the previous state, while attempting to change to the next state. The role of any activity would then be just to accept and return the increment, but not to create it.

- An empty document corresponds to the initial state. End-state candidates are all those in which a document contains a single or a group of content units which are a result of a specific, self-contained real system activity.
- It is possible to define a type and document instance constraints, referring to a structure, contents and layout in each of the states, as well as interpretation specific constraints for any output format (PDF, HTML).
- When, at the time of generation, a document requires change of a state, the generator informs the environment and waits for the continuation signal (WaitForCU).
- Each subsequent state can be defined by an increment specification with additions to the content, structure and layout. The increment is presented in Fig. 5 as a parameter „spcIncr”, in a function call for document rendering: ContDoc(docID,lastState, spcIncr).

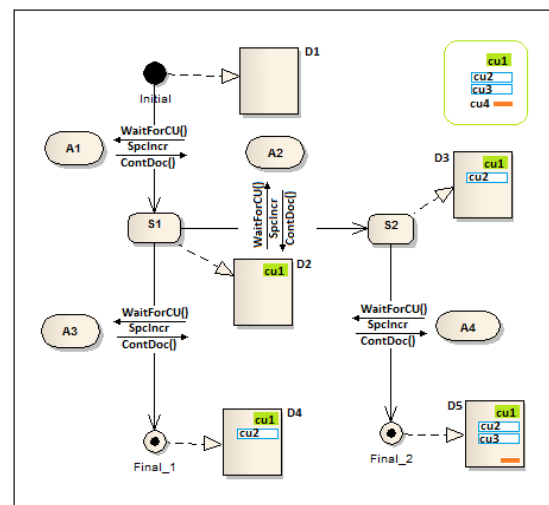


Fig. 5 Incremental specification and rendering

The core elements of the proposed incremental document rendering are the following:

- It is possible to generate a document in PDF, HTML, PS or other specific format in each state. Additionally, it is also possible to continue the process of generation by starting from a previous state. An entry point to continue the generation process is previously generated PDF or HTML document, containing the meta-data (specification of a document class it is a part of, content and the last-state identifier) and optionally, the increment of the logical script.
- Base specifications of content units are stored on a template server, a component which is a part of the document generator infrastructure.
- In each output format, a document contains meta-data, while the infrastructure services can be called by an interchange of XML packages over the known scheme, or by the interchange of documents in PDF or HTML format.



- In a simplified case, when document states are not of any importance, rendering is reduced to a change from the initial into an end state, without the synchronizations of real system activities.

## V. GENERATION OF APPLICATIONS

In the incremental approach, a document is observed as a collection of elementary units – grouped together according to particular rules. DVDocLang is a language for linear textual representation of such a collection. The rules, discussed previously, are, to the greatest extent, dependent on concrete topological relations of CU types valid for a specific document, as well as on an semantic domain of attribute values. If semantic domains and the rules for composing the structure are set, it is sufficient to connect particular content types to adequate visual patterns (Fig. 3). The MetaEdit+ tool, storing the life cycle of an “Offer”, allows us to automatically generate and test an application employed for the purpose of “Offer” production. Instead of the logical script from the Example (3),  $\langle CU \rangle$ Script from... a script, in the form of  $\langle CU.edit \rangle$ Script from..., is generated. The difference is in a modifier “edit”, which is interpreted as an application generation command aimed at editing the contents of the current CU. Formal DSL specification and MERL reports allow us to create a representative pattern collection, as well as grammatical rules necessary for their validation. The patterns in question simplify the creation of new documents and enable validation and altering of the existing document structure. For the generation of applications the following criteria have been met:

- For applications, to be driven by a business-process model, described basically at the moment of specification of the production model and document layout;
- For a HTML and PDF document’s visual interpretations, to be as close as possible to one another, or to, even, be completely indistinguishable;
- For a simple-syntax structured text (especially in a section used to describe the structure and the content of a document), to be acceptable for a user on an intuitive level;
- For the properties describing the content-unit layout of a document, to be automatically translated into the properties of adequate screen forms (i.e. controls);
- To exist a common algorithm for transformations of an application from a particular state into an appropriate document, and vice versa;
- For each document instance, in each state, to provide a fast generation of an application used to move a document into the next state.

Once a document, in a particular state of the life cycle or in the course of rendering, needs to go into the next state, the

document generator sends to the environment the action-related signal. This way the activity manager present in a real system is informed that a particular action should be performed, i.e. that a new content unit ought to be created. Fig. 6 shows one of the possible signal processing case scenarios which requests the change of a document from the S1 state into the S2 state. In case an application for the A2 activity already exists, its signal, announcing continuation of the rendering process, is waited for, with the content increment being assumed. However, if the application does not exist, it can be generated on the basis of the script increment given in the course of document type specification, to which an instance belongs to.

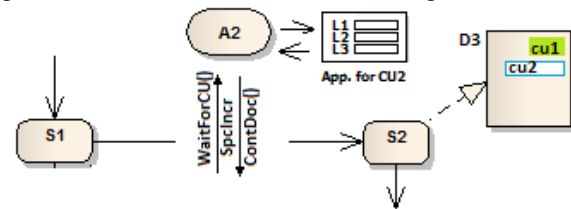


Fig. 6 Generating of application on demand

The most complex case is the one of generating production applications for such CU whose content, structure and layout were not known in advance. Owing to the DVDocLang, which is semantically rich enough, generating such web and .Net applications is possible. Web applications are implemented by way of a collection of HTML documents, which in meta-data contains: (i) a logical script, (ii) optional template definition and (iii) Java Script function which translate the current state of the interface into a logical script, and return it upon the web-service call. On a server side, combining of the initial form script with the current interface state takes place producing the current instance script.

Except for web applications, specifications of production model and layout are sufficient for incremental generation of valid application prototypes in different programming languages, such as Java and C#. Particularly important for document engineering in practice are XAML [11] and UIML [12] languages as they describe in a platform-independent way the layout and functionality of screen forms. As far as the .Net platform is concerned, priority has been given to the development of the collection of user controls, the properties of which are set dynamically – using MetaEdit+ repository and API. The next section describes such a component as well as its ‘behavior’ in a particular case.

## VI. DOMAIN ORIENTED LIBRARIES

The collection of domain-specific languages is primarily constructed with the intention to enable the mapping of domain-specific problems onto specific language concepts aspiring on a higher abstraction. Governed by the principle “what you see is what you get” (WYSIWYG), two libraries of controls have been created. The first is intended for applications in which rapid document modeling by means of

structured text is required, driven by formal DSL specification. The second is designated for template designing by means of drawing typical examples. In some cases, three types of representations are combined for the same language concept: structured text, screen forms and 2D graphic. Fig. 7 displays such combination of representations. Based on the defined data types, location, relations and anticipated size of the content unit, the space is divided into rectangles (areas). To each of the rectangles an editor is assigned, with the user being directed to the area where the data should be entered. For the 'Offer' illustrated in Fig. 4, a default layout has been displayed in Fig. 7, left. Located right is the editor, consisting of forms, i.e. of a collection of ordered triplets (CU type, modifier, value).

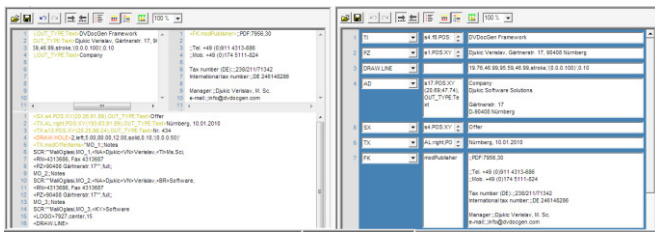


Fig. 7 Editing by structured text and screen forms

The generation of applications for document instance production is reduced to the use of the MetaEdit+ repository, for the purpose of assuming the appropriate control properties. Property values are described as MERL [4] reports, i.e. as queries on a repository. Property examples include: the list of allowed content types, modifiers allowed for a specific content type, collection of special symbols etc.

## VII. CONCLUSION

DSM can be applied to support document engineering. We have presented in more detail two domain-specific languages: DVDocAd that is employed for the modeling of small ads and DVDocFlow – that is employed for the modeling of business activities in relation to the documents and their content units. Both of the languages have been defined and implemented by means of MetaEdit+ tool. As a result, formal document specification in the domain of directory publishing is simplified with DVDocLang as it uses higher-level modeling concepts that are semantically close with the document formats. It also facilitates effective user-driven modeling. The issue of 'atypical' instances has been solved by applying the incremental specification as well as by document and application generation. With the developed approach we have got up to forty times faster rendering in the case of small ads [13] in comparison to a FOP generator. In the case of composite documents, including the concepts which almost entirely account for XSL-FO [17], the speed is ten times greater. One framework is created, which makes workflow reporting possible at any given time. By directly relating content units to activities that form them we are able to control the document related workflow. A document is

treated as a 5D entity (pentaformat). If the document is reduced to three dimensions (content, structure and layout) automatic application generation, inheritance and cloning become impossible.

We have presented in this paper an overview of the developed languages and related tools. They enable automation of document engineering in directory publishing. We believe that various areas of document engineering may also be supported by our approach. In particular two most significant areas that we are concentrating on at the moment are semantically based 2D graphic editors for template drawing (DVDoc Editor) and query language for document browsing (DVQL).

## VIII. ACKNOWLEDGEMENT

The authors would like to kindly thank Juha-Pekka Tolvanen from the University of Jyväskylä for his valuable support and proof reading.

## REFERENCES

- [1] Steven Kelly, Juha-Pekka Tolvanen, „Domain-Specific Modeling: Enabling Full Code Generation“, ISBN: 978-0-470-03666-2, March 2008, Wiley-IEEE Computer Society Press.
- [2] Robert J. Glushko, Tim Mc Grath, „Document Engineering“, MIT Press 2008.
- [3] Dirk Draheim, Gerald Weber, „Form-Oriented Analysis“, Springer-Verlag 2005, ISBN 3-540-20593-4.
- [4] MetaEdit+ Modeler, MetaCase, www.metacase.com
- [5] Di Iorio, A. Pattern-based Segmentation of Digital Documents: Model and Implementation, Ph.D. Thesis, UBLCS-2007-05, Department of Computer Science, University of Bologna. 2007.
- [6] Verislav Djukic, "DVDocLang Language Reference", www.dvdocgen.com/Framework/DVDocLang.pdf
- [7] Antonina Dattolo, Angelo Di Iorio, Silvia Duca, Antonio A. Feliziani, Fabio Vitali, „Structural patterns for descriptive documents“, Proceedings of the 7th international conference on Web engineering, Italy, Lecture Notes In Computer Science, 2007
- [8] Ivan Lukovic, Verislav Djukic, DVDocLang vs. XSL-FO, www.dvdocgen.com/Framework/DVDocLang\_XSL-FO.pdf
- [9] Angelo Di Iorio, Luca Furini, Fabio Vitali, „Higher-level Layout through Topological Abstraction“, ACM DocEng 2008
- [10] Apache Software Foundation: "FOP", <http://xmlgraphics.apache.org/fop/0.95/index.html>
- [11] Microsoft Extensible Application Markup Language (XAML) <http://xml.coverpages.org/ms-xaml.html>
- [12] User Interface Markup Language (UIML) <http://www.uiml.org/>
- [13] Verislav Djukic, "DVDoc Render Benchmark", <http://www.dvdocgen.com/Framework/DVDocRenderBench.pdf>
- [14] Kosar T., Oliveira N., Mernik M., Pereira M. J. V., Črepinšek M., Cruz D., Henriques P. R., Comparing General-Purpose and Domain-Specific Languages: An Empirical Study, Computer Science and Information Systems (ComSIS), ISSN: 1820-0214, Vol. 7, No. 2, May 2010, pp 247-264.
- [15] Mernik M., Heering J., Sloane M. A., When and How to Develop Domain-Specific Languages, ACM Computing Surveys (CSUR), Association for Computing Machinery, USA, Vol. 37, No. 4, 316-344. 2005
- [16] OMG Model Driven Architecture, <http://www.omg.org/mda/>
- [17] Extensible Stylesheet Language, Formatting Objects (XSL-FO), Reference Manual, <http://www.w3.org/TR/xsl/>.

## A MOF based Meta-Model of IIS\*Case PIM Concepts

Milan Čeliković,  
University of Novi Sad,  
Faculty of Technical Sciences,  
Trg D. Obradovića 6,  
21000 Novi Sad, Serbia,  
Email: milancel@uns.ac.rs

Ivan Luković,  
University of Novi Sad,  
Faculty of Technical Sciences,  
Trg D. Obradovića 6,  
21000 Novi Sad, Serbia,  
Email: ivan@uns.ac.rs

Slavica Aleksić,  
Vladimir Ivančević  
University of Novi Sad,  
Faculty of Technical Sciences,  
Trg D. Obradovića 6,  
21000 Novi Sad, Serbia,  
Email: {slavica,  
dragoman}@uns.ac.rs}

**Abstract**—In this paper, we present platform independent model (PIM) concepts of IIS\*Case tool for information system (IS) modeling and design. IIS\*Case is a model driven software tool that provides generation of executable application prototypes. The concepts are described by Meta Object Facility (MOF) specification, one of the commonly used approaches for describing meta-models. One of the main reasons for having IIS\*Case PIM concepts specified through the meta-model, is to provide software documentation in a formal way, as well as a domain analysis purposed to create a domain specific language to support IS design. Using the meta-model of PIM concepts, we can generate test cases that may assist in software tool verification.

### I. INTRODUCTION

IIS\*Case is a software that provides a model driven approach to information system (IS) design. It supports conceptual modeling of database schemas and business applications. IIS\*Case, as a software tool assisting in IS design and generating executable application prototypes, currently provides:

- Conceptual modelling of database schemas, transaction programs, and business applications of an IS;
- Automated design of relational database subschemas in the 3rd normal form (3NF);
- Automated integration of subschemas into a unified database schema in the 3NF;
- Automated generation of SQL/DDDL code for various database management systems (DBMSs);
- Conceptual design of common user-interface (UI) models; and
- Automated generation of executable prototypes of business applications.

In order to provide design of various platform independent models (PIM) by IIS\*Case, we created a number of modelling, meta-level concepts and formal rules that are used in the design process. Besides, we have also developed and embedded into IIS\*Case visual and repository based tools that apply such concepts and rules. They assist designers in creating formally valid models and their storing

as repository definitions in a guided way. Main features of IIS\*Case and the specification of its usage may be found in [10].

There is a strong need to have PIM concepts specified formally in a platform independent way, i.e. to be fully independent of repository based specifications that typically may include some implementation details. Our current research is based on two related approaches to formally describe IIS\*Case PIM Concepts. One of them is based on MOF and the other one on a textual Domain Specific Language (DSL). In [1], we give a specification of the IIS\*Case textual modelling language, named IIS\*CDesLang that formalizes IIS\*Case PIM concepts and provides modelling in a formal way. IIS\*CDesLang meta-model is developed under a visual programming environment for attribute grammar specifications named VisualLISA [17].

In this paper, we propose a meta-model of IIS\*Case PIM concepts, which is based on the Meta Object Facility (MOF) 2.0. MOF 2.0 is a common meta-meta-model proposed by Object Management Group (OMG) where a meta-model is created by means of UML class diagrams and Object Constraint Language (OCL) [14]. As we could not find standardized implementation of MOF, we decided to use Ecore meta-meta-model. Ecore is the Eclipse implementation of MOF 2.0 in Java programming language which is provided by Eclipse Modelling Framework (EMF) [9]. Ecore concepts are not always identical to MOF 2.0 concepts, but they are expressive enough to create our IIS\*Case meta-model. A benefit of such a meta-model is providing software documentation in formal way. Besides, created meta-model can be used for the software tool verification in EMF environment. It also represents a domain analysis specification necessary to create IIS\*CDesLang, as a textual DSL to support IS design.

In Figure 1 we illustrate the four layered architecture of our solution, which is tailored from OMG four-layered architecture standard. Level M3 comprises meta-meta-model (MOF 2.0) [7] that is used for implementation of the IIS\*Case meta-model (M2). M2 level represents the IIS\*Case PIM meta-model specified by MOF specification

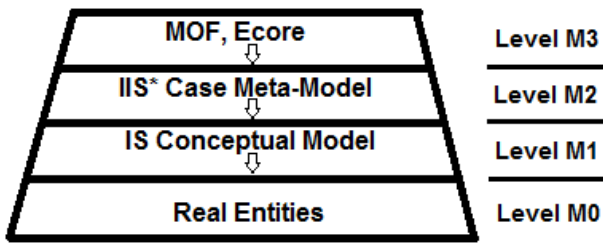


Fig 1. Four layered meta-data architecture

and implemented in EMF. Using the IIS\*Case PIM meta-model, a designer can specify and implement a conceptual model of an IS that is placed at the M1 level of the four-layered data architecture from Figure 1. By using applications of an IS generated by IIS\*Case, end users manipulate real data, i.e. they create and use models of entities from real world (M0), using the conceptual model (M1).

Apart from Introduction and Conclusion, the paper is organized in two sections. In Section 2, we present a related work, while in Section 3 we give a presentation of IIS\*Case PIM concepts specified through the meta-model that is implemented in EMF environment.

## II. RELATED WORK

Nowadays, meta-modeling is widely spread area of research and there is a huge number of references covering MOF based meta-models. However, we could not find papers presenting formal approaches to specifying meta-model implementation and design of CASE tools, based on MOF or Ecore meta-meta-models.

We found a vast number of meta-model specifications and implementations based on MOF or Ecore specifications. Meta-models based on MOF are presented in [2], [3]. The authors in both papers propose the meta-models of the Web

Modeling Language. The meta-model specification and design is implemented under EMF environment. Defining W2000 [2] as a MOF meta-model the authors specify it as an UML profile. In [3], the authors provide a solution for generation of MOF meta-models from document type definition (DTD) specifications [15]. A formal specification of OCL is given in [4]. In their meta-model, the authors precisely define the syntax of OCL, as it is given in [14]. They propose a solution for the presented meta-model integration with the UML meta-model. In [5], the authors propose the Kernel MetaMetaModel (KM3) that represents a DSL for meta-model definition. In [16], the authors propose the UML Profile, EUIS, for the specification of business applications' user interfaces. Their solution provides automatic interface code generation that is based on their own HCI standard. They developed a DSL specified as UML Profile that offers user interface modeling and generation.

There are various meta-modeling tools that are generally based on their own meta-meta-model specifications. One of them is Generic Modeling Environment (GME) [8], a configurable toolkit for domain specific modeling and program synthesis based on UML meta-models. MetaEdit+ [6] allows the creation and the design of meta-models in graphical editor using the Object-Property-Role-Relationship data model. All of these tools can also be used for the IIS\*Case PIM meta-model description in a formal way.

## III. IIS\*CASE META-MODEL

In this paper, we present the IIS\*Case PIM meta-model specified by Ecore meta-meta-model. Hereby we give an overview of the following IIS\*Case main PIM concepts: *Project*, *Application system*, *Form type*, *Component type*, *Application type*, *Program unit* as well as *Fundamental concepts* such as *Attributes* and *Domains*. A model of the

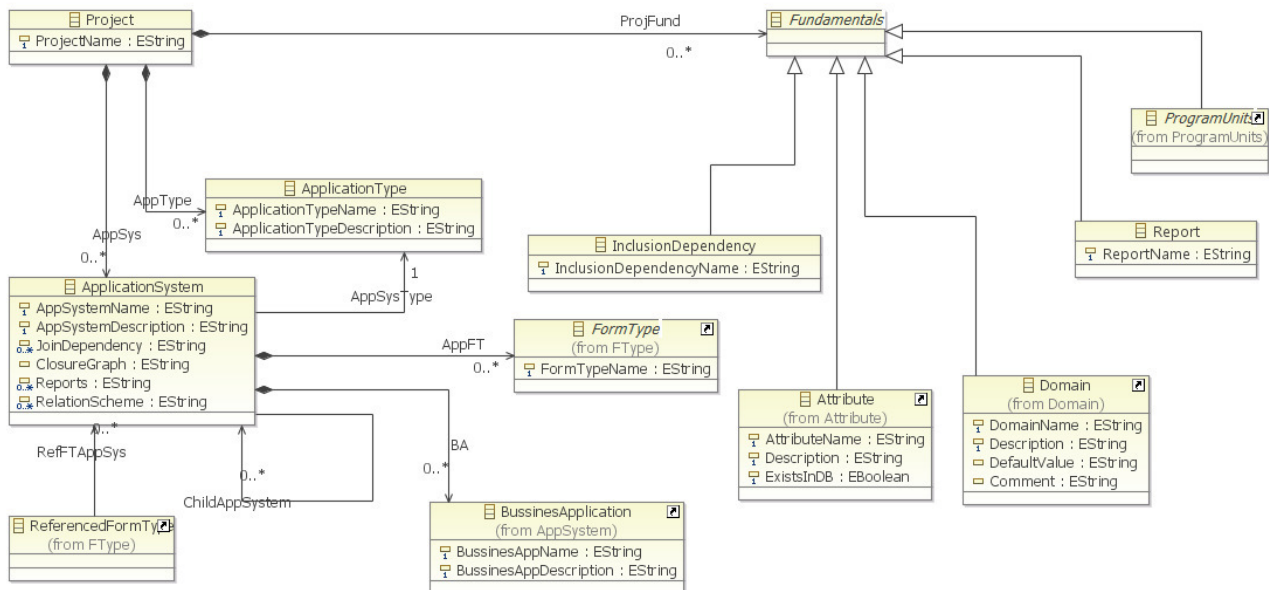


Fig. 2. A meta-model of IIS\*Case main PIM concepts



IIS\*Case main concepts with their properties and relationships is presented in Figure 2. More information about these concepts may be found in [10] and [11], as well as in many other authors' references.

#### A. Project

A modeling process in the IIS\*Case tool is organized through one or more projects. Therefore, the central concept in our meta-model from Figure 2 is *Project*. For each project, a designer defines the project name as its mandatory property. All existing elements in the repository of IIS\*Case are always created in the context of a project. *Fundamental concepts* and *Application systems* are subunits of a *Project*. For each project, we can define zero, or more instances of the *Application system*. A designer of an IS can create application systems of various types. By the *Application type* concept, a designer may introduce various application system types and then associate each instance of an application system to exactly one application type.

In the following example, we illustrate the usage of the application system and application type concepts. We have two application systems created: *Student Service* and *Faculty Organization*. *Student Service* is the child application system of the parent application system *Faculty Organization*. Two kinds of application types are created: a) *System* and b) *Subsystem*. Further, we classify application system *Faculty Organization* as the *System* and *Student Service* as the *Subsystem* application type.

Each project is organized through application systems and fundamental concepts. Fundamental concepts are formally independent of any application system. Fundamental concept instances can be used in more than one application system, because they are defined at the level of a project.

Fundamental concepts comprise zero or more:

- *Attributes*,
- *Domains*,
- *Program units*,
- *Reports* and
- *Inclusion dependencies*.

At the level of a project, IIS\*Case provides generation of various types of repository reports.

#### B. Domain

*Domains* specify allowed values of database attributes. They are classified as:

- Primitive and
- User defined.

Therefore, in our meta-model, there are two classes: *PrimitiveDomain* and *UserDefinedDomain* that are subclasses of a *Domain* class.

Primitive domains represent primitive data types that exist in formal languages, such as string, integer, char, etc. The reason for existence of user defined domain concept is to allow designers to create their own data types in order to raise the expressivity of their models. Each domain has its

domain name, description and default value. At the level of a primitive domain, a designer may specify *length required* item value. It specifies if a numeric length: must be, may be, or is not to be given. For user defined domains, a designer needs to define a domain type and a check condition. IIS\*Case supports two classes of user defined domains:

- Domains created by the inheritance rule and
- Complex domains.

A domain created by the inheritance rule references a specification of some primitive or user defined domain. By the inheritance, all the rules defined at the level of a referenced (superordinated) domain also hold for the specified domain. We call it a child domain.

Complex domains may be created by the tuple rule, set rule, or choice rule. A domain created by the tuple rule we call simply tuple domain, because it represents a tuple of values. The items of such a tuple structure are some of already created attributes. A domain created by the choice rule we call a choice domain. It is specified in almost the same way as a tuple domain. The choice domain concept is the same as the choice type of XML Schema Language. Each value of a choice domain corresponds to exactly one attribute. A set domain represents sets of allowed values over a specified domain.

Check condition is a regular expression that can additionally constrains possible values of a domain created by a designer.

*Domain* concept allows definition of display properties of screen items that correspond to attributes and their domains. Each domain corresponds to exactly one element of type *Display*. The *Display* concept specifies rules, later used by the application generator to generate screen or report items that correspond to some of the attributes, and attributes correspond to some of domains. Technical aspects of the display properties implementation may be found in [12] and [13].

#### C. Attribute

In Figure 3, we present a meta-model of the IIS\*Case *Attribute* concept. Each attribute in an IIS\*Case project is identified by its name. It also has a description and a Boolean specifier if it belongs to the database schema. In practice, the most of created attributes belong to the database schema. For attributes representing derived (calculated) values in reports or screen forms a designer may decide if they are to be included in the database schema. By this, we classify attributes as: a) included or b) non-included in a database schema.

According to the way how an attribute gains a value, we classify attributes as: a) non-derived or b) derived. A value of a non-derived attribute is created by an end user. A value of derived attribute is always calculated from the values of other attributes, by applying some function, i.e. a calculation formula. There is a rule that any non-included attribute must be specified as derived one.

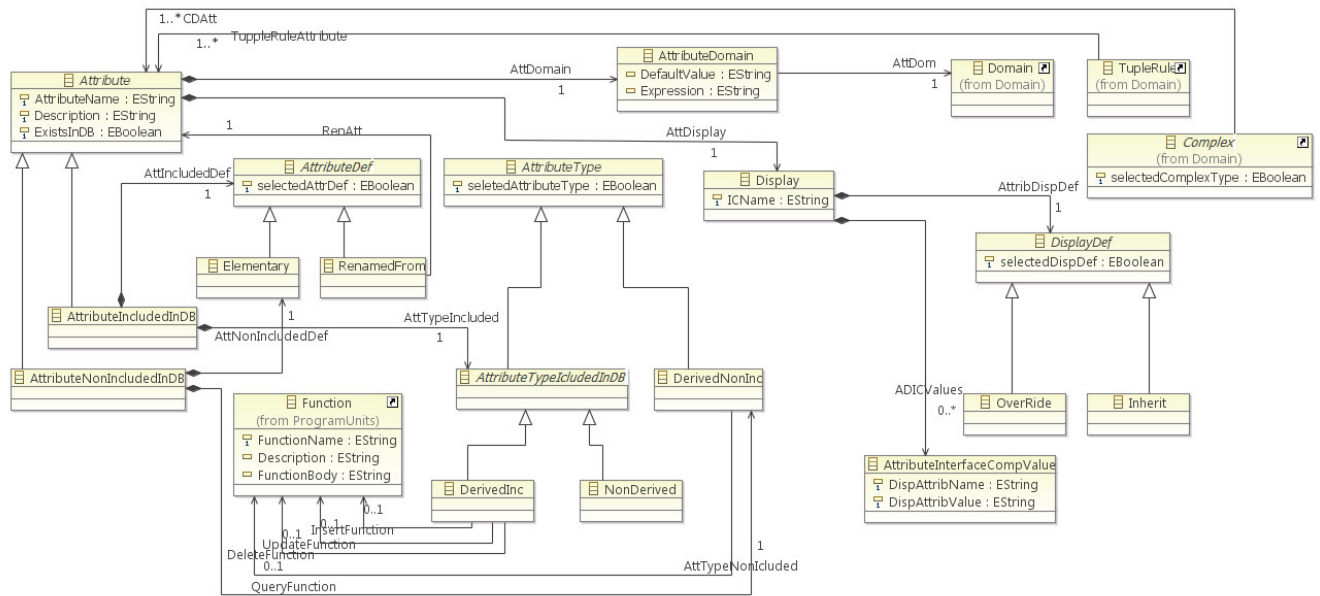


Fig. 3. A meta-model of the IIS\*Case Attribute concept

The function that is used to calculate a derived attribute value is formally specified in the IIS\*Case repository. Additionally a designer may specify parameters that are passed into the function. The *Function* concept will be presented in the following subsection, Program Units. If an attribute is non-included in a database schema, the function is referenced as a query function. Only derived attributes that are included in a database schema may additionally reference three IIS\*Case repository functions specifying how to calculate the attribute values on the following database operations: insert, update and delete.

The attribute may be specified as a) elementary or b) renamed. A renamed attribute references a previously defined attribute. The source of such an attribute is the referenced attribute, but with the different semantics. The renamed attribute needs to be included in database schema.

To each attribute a domain must be associated. This association allows defining a default value and a check condition. If the attribute value is not specified, the default value is assigned to it. Check condition is the attribute check expression that represents the regular expression that additionally constrains the value of the attribute.

At the level of an attribute, we can specify the display properties. The concept of the *Display* properties is same as the one at the level of the *Domain* concept. The values of display properties, specified at the level of the associated domain, may be inherited or overridden according to the requirements of an IS project.

#### D. Program Units

The *Program unit* concept is used to express complex application functionalities. We classify program units as: a) *Functions*, b) *Packages* and c) *Events*.

The *Function* concept is used to specify any complex functionality that later may be used in other specifications. Each function has its name and return type that are mandatory properties, as well as a formal specification of a function body and a description that are optional. The return type is a reference to a domain. A function specification may include a list of formal parameters. Each formal parameter of a function is specified by its name and a sequence number, as mandatory properties. Exactly one domain is associated to each formal parameter. Any parameter may also have a default value specified. With respect to the ways of exchanging values between the function and its calling environment, we classify formal parameters as: a) In, b) Out and c) In-Out, with a usual meaning as it is in many general purpose programming languages.

IIS\*Case provides grouping created functions into packages. Each function may be included into one or more packages, or may stay as a stand-alone object. By the location of the deployment in a multi-layer architecture, the packages are classified as: a) Database server packages, b) Application server packages and c) Client packages. A package is identified by its name, and may have an optional description.

The *Package* concept is modeled by the inheritance rule. We have the abstract class named *Package*. It is superordinated to the classes: *DBServerPackage*, *ApplicationServerPackage* and *ClientPackage*. For each instance of the *Package* class, there may be zero or more references to the instances of the *Function* class.

The *Event* concept is used to represent any software event that may trigger some action under a specified condition. Each event is identified by its name, and may have an optional description. Similar to the packages, by the location of the deployment in a multi-layer architecture, we also

classify events as: a) Database server events, b) Application server events and c) Client events. The *Event* concept is modeled in the similar way like *Package*, by applying the inheritance rule.

*E. Application System*

The *Application System* concept is used to model organizational parts of each Project. Each application system has its name and a description as mandatory properties. Besides, it may reference other, subordinated application systems and we call them child application systems. By this, a designer may create a hierarchy of application systems in a project. Application system hierarchy is modeled by a recursive reference.

Various kinds of IIS\*Case repository objects may be created at the level of an application system, but in this paper we focus on two of them only, as PIM concepts: a) *Form type* and b) *Business Application*.

*F. Form type*

Form type is the main concept in IIS\*Case. The meta-model of this concept is presented in Figure 4. It abstracts document types, screen forms, or reports that end users of an information system may use in a daily job. By means of the *Form type* concept, a designer indirectly specifies at the level of PIMs a model of a database schema with attributes and constraints included, as well as a model of transaction programs and applications of an information system.

Apart from creating form types in an application system, a designer may include into the application system form types created in other application systems. Therefore, we classify form types as: a) owned and b) referenced. A form type is owned if it is created in an application system. It may be

modified later on through the same application system without any restrictions. A referenced form type is created in another application system and then included into the application system being considered. All the referenced form types in an application system are read-only.

Each form type has a name that identifies it in the scope of a project, a title, frequency of usage, response time and usage type. Frequency is an optional property that represents the number of executions of a corresponding transaction program per time unit. Response time is also an optional property specifying expected response time of a program execution. By the usage type property, we classify form types as: a) menus and b) programs.

Menu form types are used to model menus without data items. Program form types model transaction programs providing data operations over a database. They may represent either screen forms for data retrievals and updates, or just reports for data retrievals. As a rule, a user interface of such programs is rather complex. A program form type may be designated as *considered in database schema design* or *not considered in database schema design*. Form types considered in database schema design are used later as the input into the database schema generation process. Form types not considered in database schema design are not used in the database schema generation process. They may represent reports for data retrievals only.

Each program form type is a tree of component types. A component type has a name, title, number of occurrences, allowed operations and a reference to the parent component type, if it is not a root component type. Name is the component type identifier. All the subordinated component types of the same parent must have different names.

Each instance of the superordinated component type in a

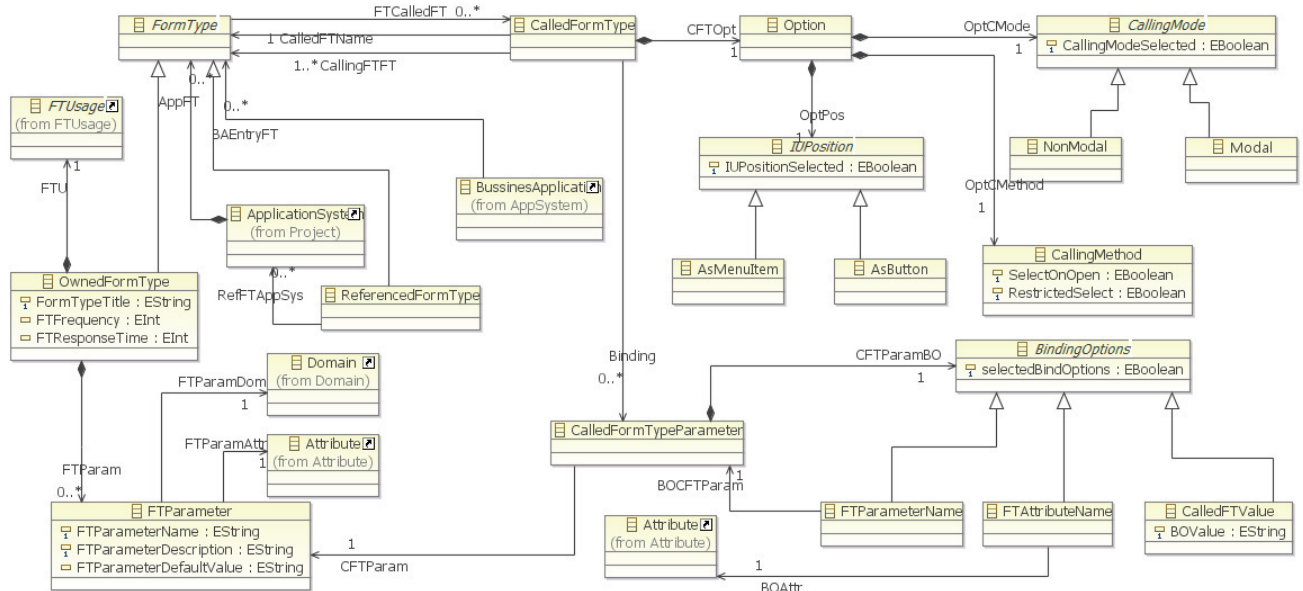


Fig. 4. A meta-model of the IIS\*Case Form Type concept



tree may have more than one related instance of the corresponding subordinated component type. The number of occurrences constrains the allowed minimal number of instances of a subordinated component type related to the same instance of a superordinated component type in the tree. It may have one of two values: 0-N or 1-N. The 0-N value means that an instance of a superordinated component type may exist while not having any related instance of the corresponding subordinated component type. The 1-N value means that each instance of a superordinated component type must have at least one related instance of the subordinated component type.

The allowed operations of a component type denote database operations that can be performed on instances of the component type. They are selected from the set {*query*, *insert*, *update*, *delete*}.

A designer can also define component type display properties that are used by the program generator. The concept of component type display is defined by properties: window layout, data layout, relative order, layout relative position, window relative position, search functionality, massive delete functionality and retain last inserted record.

Window layout has two possible values: “New window” and “Same window” and specifies if the component type is to be placed in a new window or in the same window as the parent component type. Data layout specifies the way of component type representation in a screen form. Two values are possible: “Field layout” or “Table layout”. By the “Field layout”, only one record at a time is displayed in a form. By the “Table layout”, a set of records at a time is displayed in a screen form, in a form of a table. The relative order is a sequence number representing the order of a component type relative to the other sibling component types of the same parent in a form type tree. The layout relative position represents the component type relative position to the parent component type. We may select “Bottom to parent” value if we want to place the component type below the layout of the parent component type in a generated screen form, or “Right to parent” value if it is to be placed right to the parent one. Window relative position is to be specified only when “New window” layout is selected. A designer may specify one of the three possible values: “Center”, “Left on top”, or “Custom”. The “Center” value denotes that the center of a new window is positioned to match the center of the parent window. “Left on top” specifies that the top left corner of the new window will match the top left corner of the parent window. By selecting the “Custom” value, a relative position of the new window top left corner to the top left corner of the parent window is explicitly specified by giving X and Y relative positions.

“Search functionality” represents the Boolean property that enables generation of the filter for data selection. If search functionality is enabled, end-users are allowed to refine the WHERE clause of a SQL SELECT statement. If checked, “massive delete functionality” provides generating

a delete option next to each record in a table layout. The “retain last inserted record” property specifies if the last inserted record is to be retained in the screen for future use.

Each component type includes one or more attributes. A component type attribute is a reference to a project attribute from the Fundamentals category. It has a title that will appear in the generated screen form. Also, it may be declared as mandatory or optional on the screen form. The allowed operations of a component type attribute denote database operations that can be performed on the attribute, by means of the corresponding screen item. They are selected from the set {*query*, *insert*, *update*, *nullify*}. For a component type attribute a designer may also specify display properties and by this define its presentation details in the screen form. The display properties are specified in the same way as it is for attribute specifications. Values of the display properties may be inherited from the attribute specification or overridden.

So as to unify the layout formatting rules of selected component type attributes, a designer may group them into items groups. Each item group may include one or more component type attributes or other item groups from the same component type. Any item group has its name, title, context and overflow properties. The name and title are mandatory properties. Context and overflow are Boolean properties, specifying if an item group is to be used for presenting layout contextual information or as a layout overflow area.

Each component type attribute provides defining a “List of values” (LOV) functionality. To do that, a designer needs to reference a form type that will serve as a LOV form type. He or she should also define how an end user can edit attributes: “Only via LOV” or “Directly & via LOV”. “Only via LOV” property means that attribute value may not be inserted or edited using a keyboard, but only using the LOV. “Directly & via LOV” means that inserting or editing attribute values is provided both via keyboard and LOV. “Filter value by LOV” property specifies if all values from LOV will be displayed, or only those filtered according to the pattern given by an end user. Restrict expression represents the where clause that is concatenated to the rest of where clause in the SQL statement supporting the LOV.

Each component type has one or more keys. Each component type key comprises one or more component type attributes. It represents the unique identification of a component type instance but only in the scope of its superordinated component instance. Uniqueness constraints may be defined for each component type also. Each component type uniqueness constraint comprises at least one component type attribute, but may have more than one. If uniqueness constraint attributes have non-null values, it is possible to uniquely identify a component type instance but only in the scope of the superordinated component instance.

In Figure 5, we illustrate the usage of the Form Type concept. We have the form type *Student\_Grades*, that has two component types *Students* and *Grades*. *Student\_Grades*

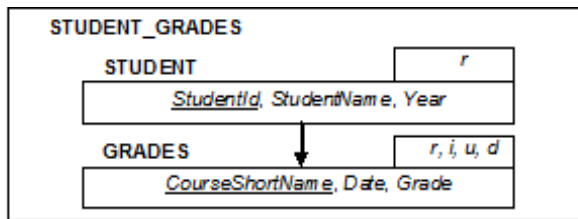


Fig. 5. A form type Student\_Grades

form type refers to the information about student grades. *Student* component type represents instances of students, while *Grades* represents instances of grades for each student. *Student* component type is the parent to the *Grades* component type.

Allowed database operation for *Student* is read, while the allowed operations for *Grades* are read, insert, update and delete. The end user of the generated transaction program specified through the form type *Student\_Grades*, is able to read data from the set of student instances. He or she can read, update and delete existing grades for each of the students, but can also insert new instances of the grades. *Student* component type should be positioned in *New Window* layout and *centered* to its parent window. Data layout of *Student* component type is in the form of *field layout* style. *Multiple deletions* and *retaining last inserted record* for student records in the screen form are not allowed, while *search functionality* for student records is enabled. Search functionality property allows generation of the filter for the selection of student instances in the generated screen form, so that the end users are able to refine the SQL SELECT statement. *Student* component type owns three attributes: *StudentId*, *StudentName* and *Year*. *StudentId* is the key of the *Student* component type. For each of *Student* component type attributes, we may specify values of the properties, previously presented.

In the similar way we can give the specification of the *Grades* component type with attributes *CourseShortName*, *Date* and *Grade*. *CourseShortName* is the key of the *Grades* component type.

### G. Business Application

*Business Application* concept represents the way to formally describe an IS functionality and is organized through a structure of form types. Each business application has a name and a description. One of the form types included into the structure must be declared as the entry form type of the application. It represents the first transaction program invoked upon the start of the application. Each business application must have the entry form type. To create the form type structure of an application, a concept of the form type call is used. By the form type calls, designers model execution of calls between generated transaction programs. They are also used to model parameters and passing the values between two transaction programs during the call

executions. The concept of a form type call comprises two form types: a calling form type and a called form type.

Any form type may have formal parameters defined. Each formal parameter has a mandatory name as the identifier. It must be related to exactly one domain. In the specification of a form type call, it is possible to associate each parameter to a called form type attribute. By this, a designer specifies to which attributes real parameter values will be passed during the call execution.

For a called form type in a call we need to specify Binding and Options properties. Binding property comprises formal parameters of a called form type. For each parameter a designer specifies how a real argument value is to be passed to the parameter. There are three possible options: "value", "attribute reference", or "parameter reference". The value is a constant that will be passed during a call execution. The "attribute reference" provides a relation to a calling form type attribute that gives a value to be passed to the parameter during a call execution. The "parameter reference" provides a relation to a calling form type parameter that gives a value to be passed to the parameter during a call execution.

The Options properties comprise: calling method, calling mode, and UI position. Calling method comprises two Boolean properties: a) "Select on open" and b) "Restricted select". "Select on open" means that the called form type is opened with an automatic data selection. "Restricted select" allows the data selection in the called form type restricted just to the values of passed parameters. Calling mode specifies a general behavior of the calling form type during the call execution. Three possibilities are allowed: "Modal", "Non-modal" or "Close calling form". "Modal" means that a user cannot activate the calling form type while the called form type is opened. "Non-modal" means that both the calling and the called form type are simultaneously active in the screen. "Close calling form" is used to cause the closing of the calling form type during the call execution. UI position specifies how a call will be provided at the level of UI: as a menu item or as a button item.

## IV. CONCLUSION

In this paper we presented the IIS\*Case PIM meta-model, based on MOF 2.0 specification. Our intention was not to present all the elements of our meta-model in detail. Instead, we tried to focus just on those meta-model details that are necessary to give a general picture of the model. We believe that the formal specification of our meta-model is not for documentation purposes only, but it is a necessary step in creating a textual DSL to support IS design and give another view of the IS description.

We may use meta-model presented in this paper in the verification of relational database schemas. We assist designers to detect conflicts at the level of relational database model, and then we can help them at the level of meta-models to find the appropriate solution of detected problems. Although the algorithms for detection and

resolving constraint collisions at the level of relational data model has already been implemented in IIS\* Case, we want to raise the process of collision resolving at the PIM level of abstraction.

Our further research will include experiments with other technologies that rely on MOF. The presented meta-model is a good base for a research in the area of Query View Transform (QVT) set of languages. Our intention is to embed into IIS\*Case transformations between different data models. Providing data model transformations may play an important role in the IS design process. In the course of data reengineering process, our plan is to provide the data integration from various sources based on different data models. Data transformation rules specified by QVT could be applied at the level of meta-models specified by various data-models, all expressed in a unified manner in MOF. Our intention is to provide transformations of the models specified in IIS\* Case to the UML models. Providing such transformations we allow designers to have models specified in UML standard with OCL constraints.

#### ACKNOWLEDGMENT

The research presented in this paper was supported by Ministry of Education and Science of Republic of Serbia, Grant III-44010: Intelligent Systems for Software Product Development and Business Support based on Models.

#### REFERENCES

- [1] I. Lukovic, M. J. Varanda Pereira, N. Oliveira, D. Cruz, P. R. Henriques, "A DSL for PIM Specifications: Design and Attribute Grammar based Implementation", *Computer Science and Information Systems (ComSIS)*, ISSN: 1820-0214, DOI: 10.2298/CSIS101229018L, Vol. 8, No. 2, 2011, pp. 379-403.
- [2] L. Baresi, F. Garzotto, M. Maritati, "W2000 as a MOF Metamodel." In Proc. of the 6th World Multiconference on Systemics, Cybernetics and Informatics - Web Engineering track. Orlando, USA, 2002.
- [3] A. Schauerhuber, M. Wimmer, E. Kapsammer, "Bridging existing web modeling languages to model-driven engineering: A metamodel for webML", *International Workshop on Model Driven Web Engineering (2nd)*, Palo Alto, CA, 2006.
- [4] M. Richters, M. Gogolla, "A meta-model for OCL" In Proc. of the 2nd international conference on The unified modeling language beyond the standard, ISBN:3-540-66712-1, 1999.
- [5] F. Jouault, J. Bézivin, "KM3: a DSL for Metamodel Specification", In Proc. of 8th IFIP International Conference on Formal Methods for Open Object-Based Distributed Systems, Bologna, Italy, 2006, Springer LNCS 4037, pp. 171-185.
- [6] MetaCase Metaedit+, [Online] Available: <http://www.metacase.com/>.
- [7] Meta-Object Facility, [Online] Available: <http://www.omg.org/mof/>.
- [8] GME: Generic Modeling Environment, [Online] Available: <http://www.isis.vanderbilt.edu/Projects/gme/>.
- [9] Eclipse Modeling Framework, [Online] Available: <http://www.eclipse.org/modeling/emf/>.
- [10] I. Luković, P. Mogin, J. Pavićević, S. Ristić, "An Approach to Developing Complex Database Schemas Using Form Types", *Software: Practice and Experience*, 2007, DOI: 10.1002/spe.820, Vol. 37, No. 15, pp. 1621-1656.
- [11] I. Luković, S. Ristić, P. Mogin, J. Pavićević, "Database Schema Integration Process – A Methodology and Aspects of Its Applying", *Novi Sad Journal of Mathematics*, Serbia, ISSN: 1450-5444, Vol. 36, No. 1, 2006, pp. 115-150.
- [12] J. Banović, "An Approach to Generating Executable Software Specifications of an Information System", Ph.D. Thesis, University of Novi Sad, Faculty of Technical Sciences, Novi Sad, 2010.
- [13] A. Popović, "A Specification of Visual Attributes and Business Application Structures in the IIS\*Case Tool", Mr (M.Sc.) Thesis, University of Novi Sad, Faculty of Technical Sciences, 2008.
- [14] Object Management Group (OMG), OCL Specification Version 2.0, [Online] Available: <http://www.omg.org/docs/ptc/05-06-06.pdf>, June 2005.
- [15] Document Type definition (DTD), [Online] Available: <http://www.w3.org/TR/html4/sgml/dtd.html>.
- [16] B. Perisic, G. Milosavljevic, I. Dejanovic, B. Milosavljevic, "UML Profile for Specifying User Interfaces of Business Applications", *Computer Science and Information Systems (ComSIS)*, ISSN: 1820-0214, DOI: 10.2298/CSIS110112010P, Vol. 8, No. 2, 2011, pp. 405-426.
- [17] N. Oliveira, M. J. Varanda Pereira, P. R. Henriques, D. Cruz, B. Cramer, "VisualLISA: A Visual Environment to Develop Attribute Grammars", *Computer Science and Information Systems (ComSIS)*, ISSN:1820-0214, Vol. 7, No. 2, 2010, pp. 265-289.

# Memory Safety and Race Freedom in Concurrent Programming Languages with Linear Capabilities

Niki Vazou  
 Email: nvazou@softlab.ntua.gr

Michalis Papakyriakou  
 Email: mpapakyr@softlab.ntua.gr

Nikolaos Papaspyrou  
 Email: nickie@softlab.ntua.gr

School of Electrical and Computer Engineering  
 National Technical University of Athens  
 Polytechnioupoli, 15780 Zografou, Athens, Greece

**Abstract**—In this paper we show how to statically detect memory violations and data races in a concurrent language, using a substructural type system based on linear capabilities. However, in contrast to many similar type-based approaches, our capabilities are not only linear, providing full access to a memory location but unshareable; they can also be read-only, thread-exclusive, and unrestricted, all providing restricted access to memory but extended shareability in the program source. Our language features two new operators, *let!* and *lock*, which convert between the various types of capabilities.

## I. INTRODUCTION

Multi-core computers have emerged as a new wave of technology and dictate the usage of concurrent programming languages. However, shared-memory concurrency further complicates existing problems, such as how to guarantee *memory safety*, and introduces new problems, such as how to avoid *data races*. Languages in the family of ML provide a good compromise between functional and imperative characteristics; their functional nature helps in restricting problems such as the above and their strong static type system helps detecting such problems at compile time, thus saving testing time and resources. However, when ML-style references are combined with concurrency, both problems remain and cannot be tackled by ML's type system.

As far as memory safety is concerned, most problems stem from aliasing. Many approaches have been proposed to control aliasing, the most direct of which are based on Girard's linear logic [7]. Linear types systems [22] have been used for region-based languages [21], [14], for a Lisp dialect [1], and for Cyclone [9], [20], a safe dialect of C, which uses a light-weight version of linearity in the form of tracked (unique) pointers. Our work is based on a linear language with locations [13], in which aliasing is controlled via linear capabilities.

A *capability* is a communicable, unforgeable token of authority which refers to a memory location and provides an associated set of access rights. In a linear language with locations and capabilities, operator *new* allocates a new memory location  $\ell$  and returns, apart from a reference to the location  $\ell$ , a linear capability for it. This capability must be provided for any further access to that memory location. However, capabilities are *linear* objects, which means that they can be used exactly once. Every operator that reads or writes

to a memory location consumes the capability provided and, to enable further access to the location, must return a new one. Finally, operator *free* consumes the capability, rendering further access to that memory location impossible.

Data races are one of the most frequent sources of bugs in programs written in concurrent languages with shared memory. A data race occurs when two threads concurrently access the same data without synchronization, and at least one of the accesses is a write. Being time-dependent, data races can be one of the most difficult programming errors to detect, reproduce, and eliminate and this is why many researchers have implemented tools for their detection [16]. Such tools are either dynamic and lockset-based [17], [3] or static, type-based [2], [6], [5].

A direct way to detect data races in a concurrent language is to use a communication channel through which a thread can send a linear capability to another thread. If a thread owns a linear capability for a memory location  $\ell$  and sends it to another thread via a channel, this is equivalent to the first thread unlocking  $\ell$  and the second thread locking it. The  $\pi$ -calculus [12] supports communication via channels and can be extended to support linearity, thus forming a language that guarantees the absence of data races [11]. As an alternative, virtual communication channels can be constructed from linear operators [18].

Using channels directly to exchange capabilities is a burden for programmers, so other more indirect approaches have been proposed. Shi and Xi [19] have proposed a type system based on a notion of types with effects, where a special modality supports the sharing of linear resources. Ennals *et al.* [4] have proposed an imperative concurrent language for packet processing applications. They use a simple linear type system to ensure that no packet (representing a memory region) can be referenced simultaneously by multiple threads. This constraint is arguably not too restrictive, as statistically "processing of a packet by multiple threads simultaneously is rare." When allocating a packet, a thread acquires a linear handler for it. Each child of this thread can access the packet with operations that either need a linear handler (i.e., free or strong updates) or an unrestricted one (i.e., dereference or weak updates). If a child needs a linear handler, it gains it and that packet is no longer visible by the parent thread. Otherwise, the child gains

an unrestricted copy of the handler, which is alive inside its body. At runtime, this is equivalent to implicitly locking the packet. Unlocking will be performed by thread's termination.

A less restrictive approach has been proposed by Wittie and Lockhart [24]. They start with the core language  $\lambda^{low}$  [10], a sequential language whose linear type system guarantees memory safety, based on Walker's and Watkins language with linear regions [23]. On top of this, they build  $\lambda^{concurrent}$  which provides concurrency and a locking mechanism, based on capabilities. As linear capabilities can be used by exactly one thread, they introduce the notion of *lock*, which can be thought of as a non-linear capability that can be shared between several threads. A lock can be *created* by consuming a linear capability for a memory location. Later, a lock can be *acquired*, producing again the linear capability from which it was created; however, the runtime semantics ensures that at most one thread has acquired the lock at any time. When a thread that acquired the lock does not need it anymore, it can *release* it, thus consuming again the linear capability and enabling other threads to acquire the lock. A disadvantage of this mechanism is that once a lock is created for a memory location, it is not possible any more to safely deallocate this memory location.

A different approach, using a type and effect system to guarantee data race freedom, has been proposed by Gerakios *et al.* [6]. Their work extends Cyclone [8], a safe dialect of C, with concurrency. They use capabilities that are annotated with two counts, the region and the lock count, which denote whether a region is live and locked respectively. Incrementing a lock count from 0 to 1 amounts to acquiring a region lock, while decrementing counts amounts to releasing it; numbers bigger than one can be used to support aliasing and re-entrant locks. Moreover, their system includes *impure* capabilities  $(\bar{n}_1, \bar{n}_2)$ , which are obtained by splitting pure or other impure capabilities in pieces, in order to pass them to multiple threads (e.g., the pure capability  $(3, 2)$  can be split into two impure capabilities  $(2, 1)$  and  $(1, 1)$ ). An impure capability denotes that a thread's knowledge of a region's counts is inexact. The collaboration of the two kinds of capabilities with the runtime system ensures that regions are safely deallocated. A disadvantage of this system, however, is that it does not support read/write locks.

In this work we propose a type system with annotated capabilities for detecting memory violations and data races. Our language is based on  $L^3$  [13] extended with the *let!* operator [15] and a lock operator. These operators are used to annotate capabilities not only as linear or unrestricted, but also, as *read-only* or *thread-exclusive*. The typing rules ensure that each thread-exclusive value can be visible by exactly one thread. One of the main advantages of our system is that it permits concurrent read-only access to the same memory location, without reporting false positive data races. In Section II we describe our language through examples, whereas in Section III we formally define the language's syntax, typing, and operational semantics. We finish with some concluding remarks.

## II. AN INFORMAL DESCRIPTION

We use linear capabilities to guarantee memory safety and race freedom in a shared memory concurrent language. To this end, we use an appropriate set of qualifiers that provide specific privileges for memory access. Qualifiers are used in languages where linear and unrestricted values coexist, in order to distinguish between these two. In our language, only capabilities are meant to be qualified. However, as capabilities can be stored in pairs or existential packages, and used by function closures, all these kinds of values must also be qualified, in order to prevent the abuse of capabilities.

In languages with capabilities that can only be linear, there are two big disadvantages. Once a linear capability is used, it is automatically consumed; therefore, all operators that do not mean to consume a capability (e.g., read and write) must return a new instance of the capability, and programming becomes awkward. Moreover, linear capabilities cannot be duplicated; therefore they cannot be shared between threads (or other independent parts of the program) that are meant to have the same access privileges. To overcome both disadvantages, several languages with linear capabilities provide mechanisms for controllably transforming linear objects to unrestricted ones, such as Wadler's *let!* [21], Odersky's observer types [14], or the freeze and thaw operators in  $L^3$  [13].

For the same purpose, in our language we use the scope-based operator *let!*  $(x = e)$  as  $s$  at  $\rho$  then  $y = e_1$  in  $e_2$ , which has been presented in our previous work [15] in a setting with just two states: L and U. This operator evaluates  $e$  and binds its linear value to the variable  $x$ . This variable is used during the evaluation of  $e_1$  with a non-linear qualifier; it is then reinstated as linear during the evaluation of  $e_2$ . Also, the result of  $e_1$  is bound to the variable  $y$ , which may be used in  $e_2$ . In order to control the uses of the non-linear  $x$  and avoid, for instance, that it escapes in the result of  $e_1$  or through a function closure, the *let!* operator introduces a new (type-level) scope  $\rho$ , which is valid only in the context of  $e_1$ , and annotates the type of the non-linear version of  $x$  with this scope.

To be more specific, in our language a qualifier  $q$  is of the form  $s$  at  $\pi$ , where  $s$  is a state and  $\pi$  is the qualifier's scope (which can be a type-level variable  $\rho$  or the special scope  $\perp$ ). There are four possible states:

- *Unrestricted* (U): An unrestricted capability for a memory location provides no privileges for that location. It can be freely shared.
- *Read-Only* (R): A read-only capability for a memory location provides only read access for that location. It can be freely shared.
- *Thread-Exclusive* (T): A thread-exclusive capability for a memory location provides write and read access for that location. It can be shared in the context of a single thread.
- *Linear* (L): A linear capability for a memory location provides access to deallocate that location. It can never be shared.

A partial order  $\sqsubseteq$  is defined on states by  $L \sqsubseteq T \sqsubseteq R \sqsubseteq U$ . In the operator *let!*  $(x = e)$  as  $s$  at  $\rho$  then  $y = e_1$  in  $e_2$ , the

non-linear version of  $x$  is qualified by  $s$  at  $\rho$ , for some  $s \neq L$ .

When a new location is created, its capability starts with a qualifier  $L$  at  $\perp$ . (From now on, we will use  $s$  as an abbreviation for  $s$  at  $\perp$ , to simplify presentation.) However, before this capability can be used for any other purpose than deallocating the location, it must be converted to a non-linear qualifier using the `let!` operator. In this way, its qualifier is “downgraded” but the capability can now be shared. For instance, if the new qualifier has a state of  $R$  or  $U$ , the capability can be shared among various concurrent threads. Notice however that if one of these threads needs to have write access to the location, a qualifier of state  $T$  will be required. The symmetric operator `lock` ( $x = e$ ) as  $s$  at  $\rho$  then  $y = e_1$  in  $e_2$  performs this qualifier “upgrading.” It evaluates  $e$  and binds its non-linear value to the variable  $x$ . This variable is used during the evaluation of  $e_1$  with a qualifier of  $s$  at  $\rho$  (where  $s \neq L$ ); it is then reinstated to its previous qualifier during the evaluation of  $e_2$ . Also, the result of  $e_1$  is bound to  $y$  for use in  $e_2$ . In contrast to the `let!` operator, the `lock!` operator must make sure (at runtime) that no other thread possesses a conflicting capability, e.g., if a thread-exclusive lock for some location  $\ell$  is requested, that no other thread possesses a  $T$  or  $R$  capability for  $\ell$ .

We present our language informally with a series of examples. In the rest of this section, we liberally extend the formal language that will be presented in Section III with features that are orthogonal to what we present there and could easily be introduced. Most notably, we use integer and boolean values (types `Int` and `Bool` respectively) and assorted operators; we also use recursive functions, defined with a `letrec` construct.

In the simplest example, a thread allocates a memory location, accesses its contents and deallocates it.

#### Example 1

```
let  $\ulcorner \ell, p \urcorner = \text{new } 0$  in
let  $(c, r) = p$  in           //  $c : \text{L}^{\text{Cap}} \ell \text{ Int}, r : \text{Loc } \ell$ 
let!  $(x = c)$  as  $T$  at  $\rho$  then //  $x : \text{T}^{\text{at } \rho} \text{Cap } \ell \text{ Int}$ 
   $y =$ 
     $(x, r) := !(x, r) + 1$ 
in
free  $\ulcorner \ell, (x, r) \urcorner$       //  $x : \text{L}^{\text{Cap}} \ell \text{ Int}$ 
```

The expression `new 0` allocates a new memory location and returns an existential package of type  $\text{L}^{\text{Xref}} \text{Int}$ , where

$${}^q\text{Xref } \tau \equiv {}^q\exists \ell. {}^q\text{Lref } \ell \tau$$

We immediately open the package, whose contents are a type-level variable  $\ell$  (a type-level abstraction of the new memory location) and a pair  $p$  of type  $\text{L}^{\text{Lref}} \ell \text{ Int}$ , where

$${}^q\text{Lref } r \tau \equiv {}^q\langle {}^q\text{Cap } r \tau * \text{Loc } r \rangle$$

The contents of this pair are a pointer  $r$  to the new memory location and a linear capability  $c$  for this. Notice that, to unpack the pair  $p$  we use the construct `let`  $(x, y) = e_1$  in  $e_2$  which extracts both components by consuming the pair once, as required in languages that support linearity.<sup>1</sup>

<sup>1</sup>To make presentation simpler, in the examples of this section we often omit the external qualifier when constructing pairs, as it can easily be deduced from the pair’s contents. In the strict syntax of Section III, the assignment in the first example would be  $\text{T}(x, r) := \text{T}!(x, r) + 1$ .

The `let!` operator is then used to downgrade the capability  $c$  to thread-exclusive ( $T$ ) and store the downgraded capability in  $x$ . This downgraded capability provides write and read access and can be used multiple times, in the first clause of `let!`. Both the capability  $x$  and the location  $r$  are provided to enable access to that location, in both uses of operators `:=` (assignment) and `!` (dereference); variable  $y$ , the result of the assignment, is not used. Finally, in the second clause of `let!` the capability  $x$  is reinstated to linear and the free operator is used to consume it. Operator `free` is the complement of `new`, taking an argument of type  $\text{L}^{\text{Xref}} \tau$ , deallocating the memory location and returning the contents that were stored in it.

In a second example, we downgrade the capability from linear to read-only ( $R$ ), utilizing the `let!` construct, in order to share it among a couple of new threads we create.

#### Example 2

```
let!  $(x = c)$  as  $R$  at  $\rho$  then //  $c : \text{L}^{\text{Cap}} \ell \text{ Int}, r : \text{Loc } \ell$ 
   $y =$  //  $x : \text{R}^{\text{at } \rho} \text{Cap } \ell \text{ Int}$ 
     $\dots !(x, r) \dots \parallel \dots !(x, r) \dots$ 
in
   $\dots$ 
```

We should note that if the downgrade was to thread-exclusive ( $T$ ), this program would not typecheck because capabilities of state  $T$  cannot be shared among multiple threads.

Nonetheless, a thread is able to upgrade a capability’s state and gain read or write access to a memory location, as the following example shows.

#### Example 3

```
lock  $(x = c_1)$  as  $T$  at  $\rho$  then //  $c_1 : \text{R}^{\text{at } \rho_1} \text{Cap } \ell_1 \tau, r_1 : \text{loc } \ell_1$ 
   $y =$  //  $c_2 : \text{R}^{\text{at } \rho_2} \text{Cap } \ell_2 \tau, r_2 : \text{loc } \ell_2$ 
     $(x, r_1) := !(c_2, r_2)$ 
in
   $\dots$ 
```

In this example, a thread is given two read-only capabilities for locations  $\ell_1$  and  $\ell_2$ . Before it can update the contents of  $\ell_1$ , it has to acquire a thread-exclusive capability for it. As we already mentioned, at runtime the lock operator ensures that no other thread possess a  $R$  capability for  $\ell_1$  before proceeding.

As a last and more involved example, we describe how a synchronous producer-consumer program can be formalized in our language. As an abbreviation, again, to simplify presentation we omit the qualifier  $U$  from the types of functions, monomorphic or polymorphic. We also assume the existence of the following functions:

```
produce   : Unit  $\rightarrow$  Int
consume  : Int  $\rightarrow$  Unit
empty    : Unit  $\rightarrow$  Bool
write    :  $\forall \rho. \text{Int} \rightarrow \text{U}^{\text{at } \rho} \text{Cap } \ell \text{ Int} \rightarrow \text{Loc } \ell \xrightarrow{[\rho]} \text{Unit}$ 
read     :  $\forall \rho. \text{U}^{\text{at } \rho} \text{Cap } \ell \text{ Int} \rightarrow \text{Loc } \ell \xrightarrow{[\rho]} \text{Int}$ 
```

Functions *produce* and *consume* are abstractions for the actual producing and consuming of integer values; *empty* returns true if a produced value is waiting to be consumed; *write* and *read* are primitives for storing and retrieving produced values, keeping track of the empty state.

The two recursive procedures *producer* and *consumer* form the main core of the program.

#### Example 4

$$\text{// } c : \text{U}^{\text{at } \rho} \text{Cap } \ell \text{ Int}, r : \text{Loc } \ell$$

```

letrec producer =  $\text{U} \lambda d : \text{Unit}.$ 
  if empty unit then
    let  $x = \text{produce}$  unit in
      write  $[\rho] x c r$ ;
      producer unit
  else
    producer unit
in
letrec consumer =  $\text{U} \lambda d : \text{Unit}.$ 
  if not (empty unit) then
    let  $x = \text{read} [\rho] c r$  in
      consume  $x$ ;
      consumer unit
  else
    consumer unit
in
producer unit || consumer unit

```

The producer function tests whether the memory location  $\ell$  is empty; if it is, it produces a value  $x$ , writes it to the memory location  $\ell$  and recursively calls itself, otherwise it busy-waits. Similarly, the consumer function tests if a produced value is waiting in location  $\ell$ ; if it is, it reads  $x$  from the memory location  $\ell$  and consumes it, otherwise it busy-waits.

The producer function attempts to gain a thread-exclusive lock before actually writing to  $\ell$  and, similarly, the consumer function attempts to gain a read-only lock before actually reading from the memory location. These two are implemented by functions *write* and *read*, which can be defined as follows. We assume the existence of two functions *setNonEmpty* and *setEmpty* which cooperate with function *empty*.

$$\text{// } \text{setNonEmpty} : \text{Unit} \rightarrow \text{Unit}$$

$$\text{// } \text{setEmpty} : \text{Unit} \rightarrow \text{Unit}$$

```

write =  $\text{U} \Lambda \rho. \text{U} \lambda v : \text{Int}. \text{U} \lambda c : \text{U}^{\text{at } \rho} \text{Cap } \ell \text{ Int}. \text{U} \lambda r : \text{Loc } \ell.$ 
  lock  $(x = c)$  as T at  $\rho'$  then
     $y =$ 
       $(x, r) := v$ 
  in
    setNonEmpty unit
read =  $\text{U} \Lambda \rho. \text{U} \lambda c : \text{U}^{\text{at } \rho} \text{Cap } \ell \text{ Int}. \text{U} \lambda r : \text{Loc } \ell.$ 
  lock  $(x = c)$  as R at  $\rho'$  then
     $y =$ 
       $!(x, r)$ 
  in
    setEmpty unit;
     $y$ 

```

$$s ::= \text{L} \mid \text{T} \mid \text{R} \mid \text{U}$$

$$\pi ::= \rho \mid \perp$$

$$q ::= s \text{ at } \pi$$

$$r ::= \ell \mid i$$

$$\phi ::= \text{Cap } r \tau \mid \langle \tau_1 * \tau_2 \rangle \mid \tau_1 \xrightarrow{\bar{\pi}} \tau_2 \mid \forall \rho. \tau \mid \exists \ell. \tau$$

$$\tau ::= \text{Unit} \mid \text{Loc } r \mid q \phi$$

$$e ::= \text{unit} \mid x \mid \lambda x : \tau. e \mid e_1 e_2 \mid q \Lambda \rho. e \mid e [\pi]$$

$$\mid q(e_1, e_2) \mid \text{let } (x, y) = e_1 \text{ in } e_2$$

$$\mid q^\top r, e^\top \mid \text{let } \lceil l, x^\top = e_1 \text{ in } e_2$$

$$\mid \text{new } e \mid \text{free } e \mid !e \mid e_1 := e_2$$

$$\mid e_1; e_2 \mid e_1 \parallel e_2$$

$$\mid \text{let! } (x = e) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2$$

$$\mid \text{lock } (x = e) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2$$

$$\mid \text{loc } i \mid q \text{cap } i$$

$$\mid \text{let\$ } (x = e) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2$$

$$\mid \text{lock\$ } (x = e) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2$$

$$v ::= \text{unit} \mid \text{loc } i \mid q u$$

$$u ::= \text{cap } i \mid \lambda x : \tau. e \mid \Lambda \rho. v \mid (v_1, v_2) \mid \lceil i, v^\top$$

Fig. 1. Syntax.

### III. FORMALISM

#### A. The syntax

The language we use is a typed lambda calculus with ML-style references. It is polymorphic with respect to scope variables. We have kept the language as simple as possible, including only the features that are necessary to demonstrate our approach, in the light of the issues that we discussed in the previous sections.

The syntax of our language is shown in Fig. 1. Our expressions include unit, term variables, term and scope abstractions, application for terms and scopes, pairs, sequential and concurrent execution of two expressions. There are primitives to create a reference, to read or update its contents and to deallocate it. We also include existential packages, and a primitive to open them.

There are two constructs that manipulate capability qualifiers: the let! primitive, which downgrades a linear capability, and the lock primitive, which blocks thread execution until it is safe to grant the requested capability. The constructs let\$ and lock\$ are not available to the programmer, but appear only during the evaluation of a program. The same is true about the expressions for locations (loc  $i$ ) and capabilities (cap  $i$ ).

Types ( $\tau$ ) may be the unit type, location types or pretypes ( $\phi$ ) annotated by a qualifier. A qualifier consists of a state  $s$  and a scope  $\pi$ . The state can be L, T, R and U, for linear, thread-exclusive, read-only or unrestricted values, respectively. Scopes are either  $\rho$  for a scope variable or  $\perp$ , which is the external scope and is always valid. As mentioned earlier, not only capabilities must be annotated with qualifiers, but also every expression type that may contain a capability. Thus, besides capabilities, pretypes include pairs, functions, scope



$$\boxed{\Gamma; \Delta; Z; M \vdash e : \tau}$$

$$\frac{\text{state } \Gamma \neq L}{\Gamma; \Delta; Z; M \vdash \text{unit} : \text{Unit}} \quad \frac{\text{state } \Gamma \neq L \quad \text{frv}(\tau) \subseteq \Delta \quad Z \models \text{scope } \tau}{\Gamma, x : \tau; \Delta; Z; M \vdash x : \tau}$$

$$\frac{s \sqsubseteq \text{state } \Gamma \quad Z \models \rho \quad \Gamma, x : \tau; \Delta; Z_e; M \vdash e : \tau_1}{\Gamma; \Delta; Z; M \vdash {}^{s \text{ at } \rho} \lambda x : \tau. e : {}^{s \text{ at } \rho} (\tau \xrightarrow{Z_e} \tau_1)} \quad \frac{\Gamma_1; \Delta; Z_1; M \vdash e_1 : {}^q (\tau_1 \xrightarrow{Z_e} \tau_2) \quad \Gamma_2; \Delta; Z_2; M \vdash e_2 : \tau_1 \quad Z \models \text{scope } \tau_2}{\Gamma_1 \oplus \Gamma_2; \Delta; Z_1 \cup Z_2 \cup Z \cup Z_e; M \vdash e_1 e_2 : \tau_2}$$

$$\frac{\text{fresh } \rho' \quad Z' = Z \vee Z, \rho' \quad Z_1 \models \rho_1 \quad \Gamma; \Delta; Z'; M \vdash e[\rho \mapsto \rho'] : \tau \quad s \sqsubseteq \text{state } \tau}{\Gamma; \Delta; Z \cup Z_1; M \vdash {}^{s \text{ at } \rho_1} \Lambda \rho. e : {}^{s \text{ at } \rho_1} \forall \rho'. \tau} \quad \frac{Z_\tau \models \text{scope } \tau \quad Z_\pi \models \pi \quad \Gamma; \Delta; Z; M \vdash e : {}^q \forall \rho. \tau}{\Gamma; \Delta; Z \cup Z_\pi \cup Z_\tau; M \vdash e[\pi] : \tau}$$

$$\frac{\Gamma; \Delta; Z; M \vdash e : \tau}{\Gamma; \Delta; Z; M \vdash \text{new } e : {}^L \text{Xref } \tau} \quad \frac{Z_\tau \models \text{scope } \tau \quad \Gamma; \Delta; Z; M \vdash e : {}^L \text{Xref } \tau}{\Gamma; \Delta; Z \cup Z_\tau; M \vdash \text{free } e : \tau}$$

$$\frac{\Gamma_1; \Delta; Z_1; M \vdash e_1 : {}^{\text{T at } \pi} \text{Lref } r \tau \quad \Gamma_2; \Delta; Z_2; M \vdash e_2 : \tau \quad \text{state } \tau \neq L}{\Gamma_1 \oplus \Gamma_2; \Delta; Z_1 \cup Z_2; M \vdash e_1 := e_2 : \text{Unit}} \quad \frac{\Gamma; \Delta; Z; M \vdash e : {}^{s \text{ at } \pi} \text{Lref } r \tau \quad s = R \vee T \quad Z_\tau \models \text{scope } \tau}{\Gamma; \Delta; Z \cup Z_\tau; M \vdash !e : \tau}$$

$$\frac{\Gamma_1; \Delta; Z_1; M \vdash e_1 : \tau_1 \quad \Gamma_2; \Delta; Z_2; M \vdash e_2 : \tau_2 \quad s = \text{LUB}(\text{state } \tau_1, \text{state } \tau_2)}{\Gamma_1 \odot \Gamma_2; \Delta; Z_1 \cup Z_2; M \vdash e_1 \parallel e_2 : {}^s \langle \tau_1 * \tau_2 \rangle}$$

$$\frac{\text{fresh } \rho' \quad L \sqsubseteq s \quad \Gamma; \Delta; Z; M \vdash e : {}^L \text{Cap } r \tau \quad Z'_1 = Z_1 \vee Z_1, \rho' \quad \Gamma_1, x : {}^{s \text{ at } \rho'} \text{Cap } r \tau; \Delta; Z'_1; M \vdash e_1[\rho \mapsto \rho'] : \tau_1 \quad \Gamma_2, x : {}^L \text{Cap } r \tau, y : \tau_1; \Delta; Z_2; M \vdash e_2 : \tau_2}{\Gamma \oplus \Gamma_1 \oplus \Gamma_2; \Delta; Z \cup Z_2 \cup Z_3; M \vdash \text{let! } (x = e) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2 : \tau_2}$$

$$\frac{\Gamma; \Delta; Z; M \vdash e : {}^{s' \text{ at } \pi} \text{Cap } r \tau \quad L \sqsubseteq s \sqsubseteq s' \quad \text{fresh } \rho' \quad Z'_1 = Z_1 \vee Z_1, \rho' \quad \Gamma_1, x : {}^{s \text{ at } \rho'} \text{Cap } r \tau; \Delta; Z'_1; M \vdash e_1[\rho \mapsto \rho'] : \tau_1 \quad \Gamma_2, y : \tau_1; \Delta; Z_2; M \vdash e_2 : \tau_2}{\Gamma \oplus \Gamma_1 \oplus \Gamma_2; \Delta; Z \cup Z_1 \cup Z_2; M \vdash \text{lock } (x = e) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2 : \tau}$$

Fig. 2. Typing rules.

abstractions and existential packages. Function types are also annotated with a set of scopes that are used by the function body, much like in our previous work [15].

### B. Typechecking

The typing relation for our language is  $\Gamma; \Delta; Z; M \vdash e : \tau$ . Some selected typing rules are presented in Fig. 2.  $\Gamma$  is the environment that binds variables to types,  $\Delta$  is the set of live location variables,  $M$  binds locations to types (and is only needed for the metatheory), and  $Z$  is the set of live scopes.

To simplify the rules we have used the following abbreviations, which we already mentioned in Section II.

$$\begin{aligned}
{}^q \text{Xref } \tau &\equiv {}^q \exists \ell. {}^q \text{Lref } \ell \tau \\
{}^q \text{Lref } r \tau &\equiv {}^q \langle {}^q \text{Cap } r \tau * \text{Loc } r \rangle
\end{aligned}$$

Typing judgments of linear languages differ from those of regular, unrestricted languages mainly in the way they handle typing environments. In our language, only values may be linear, hence special treatment must be made only for environment  $\Gamma$ . In the typing of a composite expression, this environment is split into an appropriate number of pieces and it is ensured that each linear variable appears in exactly one piece.

To this means, we define a union operator  $\Gamma_1 \oplus \Gamma_2$ , for term environments, which is valid only if the intersection of  $\Gamma_1$  and  $\Gamma_2$  does not contain any linear values. This operator is defined in Fig. 3. To prevent linear values from being discarded, the typing of all base cases restricts  $\Gamma$  to contain only the linear bindings that are actually used. In a similar way we ensure that each thread-exclusive value can appear exactly in one thread. To this means, we define one more environment operator,  $\Gamma_1 \odot \Gamma_2$  (Fig. 3) which enforces that the intersection of  $\Gamma_1$  and  $\Gamma_2$  contains neither linear nor thread-exclusive values. According to the definition of this operator, read-only values can either be duplicated to both  $\Gamma_1$  and  $\Gamma_2$  or passed exclusively to one of them. The rationale behind this behaviour is to give us the flexibility to pass a read-only lock exclusively to one of two parallel expressions, in case the other one does not require it.

The state  $\tau$  operator is defined in Fig. 4 and returns the state part of the qualifier of a type  $\tau$ . We use function LUB to denote the least upper bound according to the ordering defined by  $\sqsubseteq$ . This operator is used in the typing rules to ensure that a pair can never exploit a linear value. It is extended to state  $\Gamma$ , which returns the LUB of all the types stored in the environment  $\Gamma$ . The scope  $\tau$  returns the scope part of the

$$\begin{array}{c}
\boxed{\Gamma = \Gamma_1 \oplus \Gamma_2} \\
\frac{\emptyset = \emptyset \oplus \emptyset}{\Gamma = \Gamma_1 \oplus \Gamma_2 \quad s = T \vee R \vee U} \\
\frac{\Gamma, x : {}^{s \text{ at } \pi} \phi = \Gamma_1, x : {}^{s \text{ at } \pi} \phi \oplus \Gamma_2, x : {}^{s \text{ at } \pi} \phi}{\Gamma = \Gamma_1 \oplus \Gamma_2} \\
\frac{\Gamma, x : {}^{L \text{ at } \pi} \phi = \Gamma_1, x : {}^{L \text{ at } \pi} \phi \oplus \Gamma_2}{\Gamma = \Gamma_1 \oplus \Gamma_2} \\
\frac{\Gamma, x : {}^{L \text{ at } \pi} \phi = \Gamma_1 \oplus \Gamma_2, x : {}^{L \text{ at } \pi} \phi}{\Gamma = \Gamma_1 \odot \Gamma_2} \\
\boxed{\Gamma = \Gamma_1 \odot \Gamma_2} \\
\frac{\emptyset = \emptyset \odot \emptyset}{\Gamma = \Gamma_1 \odot \Gamma_2 \quad s = R \vee U} \\
\frac{\Gamma, x : {}^{s \text{ at } \pi} \phi = \Gamma_1, x : {}^{s \text{ at } \pi} \phi \odot \Gamma_2, x : {}^{s \text{ at } \pi} \phi}{\Gamma = \Gamma_1 \odot \Gamma_2 \quad s = L \vee T \vee R} \\
\frac{\Gamma, x : {}^{s \text{ at } \pi} \phi = \Gamma_1, x : {}^{s \text{ at } \pi} \phi \odot \Gamma_2}{\Gamma = \Gamma_1 \odot \Gamma_2 \quad s = L \vee T \vee R} \\
\frac{\Gamma, x : {}^{s \text{ at } \pi} \phi = \Gamma_1 \odot \Gamma_2, x : {}^{s \text{ at } \pi} \phi}{\Gamma = \Gamma_1 \odot \Gamma_2}
\end{array}$$

Fig. 3. Sequential and concurrent union operator.

$$\begin{array}{c}
\boxed{\text{state } \tau} \\
\text{state Unit} = U \quad \text{state (Loc } i) = U \quad \text{state } ({}^{s \text{ at } \rho} \phi) = s \\
\boxed{\text{state } \Gamma} \\
\frac{\text{state } \Gamma = s_1 \quad \text{state } \tau = s_2}{\text{state } \emptyset = U \quad \text{state } (\Gamma, x : \tau) = \text{LUB}(s_1, s_2)} \\
\boxed{\text{scope } \tau} \\
\text{scope Unit} = \perp \quad \text{scope (Loc } i) = \perp \quad \text{scope } ({}^{s \text{ at } \rho} \phi) = \rho \\
\boxed{Z \models \pi} \\
\emptyset \models \perp \quad \{\rho\} \models \rho
\end{array}$$

Fig. 4. Auxiliary definitions in typechecking.

qualifier of a type  $\tau$  and it is used to check that this scope is valid with respect to  $Z$ . We use the  $\text{frv}(\tau)$  to gain all the free location variables that appear in  $\tau$ . We use this operator to check that all free location variables of a term variable are valid according to  $\Delta$ .

In the typing rules exposed in Fig. 2 one may notice that all reference related expressions require along with the reference the corresponding capability. Typing rules for term abstraction and application ensure that scopes used in the function closure will be alive during any application of this function. For that matter, we handle environment  $Z$  in a relevant way, thus avoiding scopes from being added if they are not actually used. Relevant treatment of  $Z$  is enforced by the *minimal scope* relation  $Z \models \pi$  between scope  $\pi$  and scope environment

$$\begin{array}{c}
\boxed{S; x \Downarrow S'; v} \\
\frac{\text{state } v = L}{S, x \mapsto v; x \Downarrow S; v} \quad \frac{\text{state } v \neq L}{S, x \mapsto v; x \Downarrow S, x \mapsto v; v} \\
\boxed{\text{state } v} \\
\text{state unit} = U \quad \text{state (loc } i) = U \quad \text{state } ({}^{s \text{ at } \rho} u) = s
\end{array}$$

Fig. 5. Auxiliary definitions in operational semantics.

$Z$ , also defined in 4. The  $\oplus$  split operator is used in every typing rule which contains subexpressions except the rule for parallel execution in which case the  $\odot$  split operator is used. Typing rules for `let!` and lock expressions are pretty-much the same, varying only on the restriction between the current and requested qualifier scope.

### C. Operational Semantics

We define a small-step, call-by-value operational semantics for our language.

Our evaluation rules require two kinds of annotations. Every expression must be annotated with its thread identifier and in every parallel expression, each of the subexpressions must be annotated with the set of locks it inherits from its parent thread. Thread identifiers are introduced during the evaluation, whereas lock set annotations can be inferred statically with the aid of our type system. To this end, we define a function  $\text{cl } e$  on expressions which, given the typing derivation, calculates a lock set for every parallel subexpression in  $e$ . It is defined recursively, treating the expression  $e_1 \parallel e_2$  in the following way:  $\text{cl}(e_1 \parallel e_2) = \text{cl}_1(\text{cl } e_1) \parallel \text{cl}_2(\text{cl } e_2)$ , where  ${}^{s \text{ at } \rho} \text{cap } l \in \text{cl}_i$  iff  $(s = R \vee T) \wedge (\exists \tau)[{}^{s \text{ at } \rho} \text{Cap } l \tau \in \Gamma_i]$ , where  $\Gamma_i$  is the environment in which  $e_i$  was typechecked. By using our type system, we ensure that locks will be distributed among child threads in a safe manner. Here, by safe, we mean that *all* locks will be distributed and that thread-exclusive locks will be given to only one child thread.

Our semantics is a relation between configurations consisting of a store  $S$ , which is a mapping from variables to values, a memory  $\mu$  which is a mapping from locations to variables, a lock environment  $t$  which is a mapping from pairs of locations and thread identifiers to states and a language term annotated by its thread identifier  $n : e$ . The basic rules of this relation are depicted in Fig. 6. For lack of space, we have omitted the propagation rules, which are straightforward.

In the relation we use an idiom that is standard in languages with linear values. Evaluation does not terminate with a value, but with a variable. This variable, called *auto-variable*, is automatically produced, but can be merged with program variables in a transparent way. Once a value  $v$  has been reached, it gets bound to a fresh variable  $z$  and placed in the store  $S$ . Access to this value may be given only through the fresh variable. We define in Fig. 5 a store lookup function  $S; x \Downarrow S'; v$  which ensures that linear objects may be used by the program only once and the state  $x$  operator that returns

$$\boxed{S; \mu; t; n : e \hookrightarrow S'; \mu'; t'; n : e'}$$

$$\frac{\text{fresh } z}{S; \mu; t; n : v \hookrightarrow S, z \mapsto v; \mu; t; n : z} \quad \frac{S; w \Downarrow S'; {}^q\lambda x : \tau. e}{S; \mu; t; n : w y \hookrightarrow S'; \mu; t; n : e[x \mapsto y]} \quad \frac{}{S; \mu; t; n : e[\rho] \hookrightarrow S; \mu; t; n : e}$$

$$\frac{\text{fresh } i}{S; \mu; t; n : \text{new } x \hookrightarrow S; \mu, i \mapsto x; t; n : {}^{L\Gamma}i, {}^L(\text{Lcap } i, \text{loc } i)^\top} \quad \frac{S; w \Downarrow S_1; {}^{L\Gamma}i, w_0^\top \quad S_1; w_0 \Downarrow S_2; {}^L(w_1, w_2) \quad S_2; w_1 \Downarrow S_3; {}^L\text{cap } i \quad S_3; w_2 \Downarrow S_4; \text{loc } i}{S; \mu, i \mapsto z; t; n : \text{free } w \hookrightarrow S_4; \mu; t; n : z}$$

$$\frac{(i \times n, s) \in t \quad T \sqsubseteq s \sqsubseteq R \quad S; w \Downarrow S_1; {}^q(w_1, w_2) \quad S_1; w_1 \Downarrow S_2; {}^{q'}\text{cap } i \quad S_2; w_2 \Downarrow S_3; \text{loc } i}{S; \mu, i \mapsto z; t; n : !w \hookrightarrow S_3; \mu, i \mapsto z; t; n : z}$$

$$\frac{(i \times n, T) \in t \quad S; x \Downarrow S_1; {}^q(x_1, x_2) \quad S_1; x_1 \Downarrow S_2; {}^{q'}\text{cap } i \quad S_2; x_2 \Downarrow S_3; \text{loc } i}{S; \mu, i \mapsto z; t; n : x := y \hookrightarrow S_3; \mu, i \mapsto y; t; n : \text{unit}}$$

$$\frac{\text{fresh } n_1 \quad \text{fresh } n_2 \quad t' = t \setminus \{i' \times n : s' \mid (\exists i' s'). [i' \times n : s' \in t]\} \cup \{i' \times n_1 : s \mid ({}^{s\text{at}\pi}\text{cap } i \in cl_1)\} \cup \{i' \times n_2 : s \mid ({}^{s\text{at}\pi}\text{cap } i \in cl_2)\}}{S; \mu; t; n : {}^{cl_1}e_1 \parallel {}^{cl_2}e_2 \hookrightarrow S; \mu; t'; n : n_1 : e_1 \# n_2 : e_2}$$

$$\frac{S; \mu; t; n_1 : e_1 \hookrightarrow S'; \mu'; t'; n_1 : e'_1 \quad S; \mu; t; n_2 : e_2 \hookrightarrow S'; \mu'; t'; n_2 : e'_2}{S; \mu; t; n : n_1 : e_1 \# n_2 : e_2 \hookrightarrow S'; \mu'; t'; n : n_1 : e'_1 \# n_2 : e'_2}$$

$$\frac{t' = t \setminus \{i' \times n' : s' \mid (i' \times n' : s') \in t \wedge (n' = n_1 \vee n' = n_2)\} \cup \{i' \times n : s' \mid (n' = n_1 \vee n' = n_2) \wedge (\exists i' s'). [i' \times n' : s'] \in t\}}{s = \text{LUB}(\text{state } v_1, \text{state } v_2)}$$

$$\frac{S, z_1 \mapsto v_1, z_2 \mapsto v_2; \mu; t; n : n_1 : z_1 \# n_2 : z_2 \hookrightarrow S; \mu; t'; n : {}^s(z_1, z_2)}{\text{fresh } \rho' \quad S; z \Downarrow S'; {}^q\text{cap } i \quad s = T \Rightarrow (\nexists n' s')[n \neq n' \wedge (i \times n' : s') \in t] \quad s = R \Rightarrow (\nexists n')[i \times n' : T] \in t}$$

$$\frac{S; \mu; t; n : \text{lock } (x = z) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2 \hookrightarrow S', z \mapsto {}^{s\text{at}\rho'}\text{cap } i; \mu; t, (i \times n, s); n : \text{lock\$ } (x = z) \text{ as } s \text{ at } \rho' \text{ then } y = e_1[\rho \mapsto \rho'] [x \mapsto z] \text{ in } e_2}{S; z \Downarrow S'; {}^q\text{cap } i \quad s = R \Rightarrow (\exists n')[i \times n' : T] \in t \quad s = T \Rightarrow (\exists n' s')[n \neq n' \wedge (i \times n' : s') \in t]}$$

$$\frac{S; z \mapsto {}^{s\text{at}\rho}\text{cap } i; \mu; t, (i \times n, s); n : \text{lock\$ } (x = z) \text{ as } s \text{ at } \rho \text{ then } y = w \text{ in } e_2 \hookrightarrow S; \mu; t; n : e_2[y \mapsto w]}{\text{fresh } \rho' \quad s \sqsubseteq R \Rightarrow t' = t, i \times n, s \quad s = U \Rightarrow t' = t}$$

$$\frac{S, z \mapsto {}^L\text{cap } i; \mu; t; n : \text{let! } (x = z) \text{ as } s \text{ at } \rho \text{ then } y = e_1 \text{ in } e_2 \hookrightarrow S, z \mapsto {}^{s\text{at}\rho'}\text{cap } i; \mu; t'; n : \text{let\$ } (x = z) \text{ as } s \text{ at } \rho' \text{ then } y = e_1[\rho \mapsto \rho'] [x \mapsto z] \text{ in } e_2}{s \sqsubseteq R \Rightarrow t' = t \setminus \{i \times n, s\} \quad s = U \Rightarrow t' = t}$$

$$\frac{}{S, z \mapsto {}^{s\text{at}\rho}\text{cap } i; \mu; t; n : \text{let\$ } (x = z) \text{ as } s \text{ at } \rho \text{ then } y = w \text{ in } e_2 \hookrightarrow S, z \mapsto {}^L\text{cap } i; \mu; t'; n : e_2[x \mapsto z][y \mapsto w]}$$

Fig. 6. Operational semantics.

the state part of the qualifier of a value  $v$ .

Memory  $\mu$  in our semantic rules is used to handle the contents of references, in a standard way. Following our previous work [15], locations are bound to variables, instead of values, which comes naturally, given that evaluation in our language ends with variables. The actual value may be regained from the binding of the variable inside the store. In this way, we take linearity handling for the contents of references for free.

A parallel expression  $n : {}^{cl_1}e_1 \parallel {}^{cl_2}e_2$  is evaluated to

an intermediate expression  $n : n_1 : e_1 \# n_2 : e_2$  attributing fresh thread identifiers to the new threads. At the same time, all locks with thread identifier  $n$  are removed from the lock set  $t$  and are passed to the child threads as dictated by the annotations  $cl_1$  and  $cl_2$ . After non-deterministically evaluating both  $e_1$  and  $e_2$  to values, we restore the original lock environment and return a pair consisting of these two values with an appropriate qualifier.

Both expressions that manipulate states, `let!` and `lock` utilize two auxiliary expressions, not available to the user: `let$`

and lock\$. These kind of expressions are needed for our metatheory, as shown in our previous work [15]. In the original expressions,  $e$  is evaluated to a capability and evaluation continues with the auxiliary expression. In the case of lock, evaluation continues only when the requested lock is available, in which case the lock and the corresponding capability are added to the set  $t$  and  $S$  respectively, for the evaluation of  $e_1$ , and are removed when this is finished. In the case of let!, the existing linear capability is removed from  $S$  in order to avoid having a linear and a non-linear capability for the same location at the same time. The new capability is added to  $S$ , and in case this is not an U capability, the appropriate lock is also added to  $t$ . When evaluation of  $e_1$  is finished, the capability and the lock are purged and the linear capability is reinstated.

In the evaluation of lock ( $x = e$ ) as  $s$  at  $\rho$  then  $y = e_1$  in  $e_2$  we follow the spin-lock approach. That is, once  $e$  is evaluated to a capability for a location  $i$ , we check whether any other thread possesses a conflicting lock for  $i$ . In case the lock is not available, the expression evaluates to itself. Otherwise, the current thread is given the required lock, and proceeds as described above.

#### IV. CONCLUSION

In this paper we have presented a substructural type system, that can be used to detect memory violations and data races in a concurrent programming language. Our system is based on capabilities, which can be of four types: linear (exclusive access, non shareable), thread-exclusive (read and write access, shareable within a thread), read-only (read access, freely shareable), and unrestricted (no access, freely shareable). Two special language constructs, let! and lock can be used to change the types of capabilities in a safe and controlled way.

#### ACKNOWLEDGMENT

This research is partially funded by the programme for supporting basic research (IIEBE 2010) of the National Technical University of Athens under a project titled "Safety properties for concurrent programming languages."

#### REFERENCES

- [1] H. G. Baker, "Lively linear Lisp: Look ma, no garbage!" *ACM SIGPLAN Notices*, vol. 27, no. 8, pp. 89–98, Aug. 1992.
- [2] C. Boyapati, R. Lee, and M. Rinard, "Ownership types for safe programming: Preventing data races and deadlocks," in *Proceedings of the ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications*, Nov. 2002, pp. 211–230.
- [3] J.-D. Choi, K. Lee, A. Loginov, R. O'Callahan, V. Sarkar, and M. Sridharan, "Efficient and precise datarace detection for multithreaded object-oriented programs," in *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation*, 2002, pp. 258–269.
- [4] R. Ennals, R. Sharp, and A. Mycroft, "Linear types for packet processing," in *Proceedings of the 13th European Symposium on Programming*. Springer, 2004, pp. 204–218.
- [5] C. Flanagan and S. N. Freund, "Type inference against races," in *Proceedings of the International Symposium on Static Analysis*. Springer-Verlag, 2004, pp. 116–132.
- [6] P. Gerakios, N. Pappaspyrou, and K. Sagonas, "Race-free and memory-safe multithreading: design and implementation in Cyclone," in *Proceedings of the ACM SIGPLAN International Workshop on Types in Languages Design and Implementation*, 2010, pp. 15–26.
- [7] J.-Y. Girard, "Linear logic," *Theoretical Computer Science*, vol. 50, pp. 1–102, 1987.
- [8] D. Grossman, G. Morrisett, T. Jim, M. Hicks, Y. Wang, and J. Cheney, "Region-based memory management in Cyclone," in *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation*, 2002, pp. 282–293.
- [9] M. Hicks, G. Morrisett, D. Grossman, and T. Jim, "Safe and flexible memory management in Cyclone," University of Maryland, Department of Computer Science, Tech. Rep. CS-TR-4514, Jul. 2003.
- [10] H. Huang, L. Wittie, and C. Hawblitzel, "Formal properties of linear memory types," Dartmouth College, Computer Science, Hanover, NH, Tech. Rep. TR2003-468, August 2003.
- [11] N. Kobayashi, "Type systems for concurrent programs," in *Proceedings of 10th Anniversary Colloquium of UNU/IIST*, ser. LNCS, vol. 2757. Springer, 2003, pp. 439–453.
- [12] R. Milner, J. Parrow, and D. Walker, "A calculus of mobile processes, I," *Information and Computation*, vol. 100, no. 1, pp. 1–40, 1992.
- [13] G. Morrisett, A. Ahmed, and M. Fluet, "A linear language with locations," in *Proceedings of the 7th International Conference on Typed Lambda Calculi and Applications*, 2005, pp. 293–307.
- [14] M. Odersky, "Observers for linear types," in *Proceedings of the 4th European Symposium on Programming*. Springer-Verlag, 1992, pp. 390–407.
- [15] M. A. Papakyriakou and N. S. Pappaspyrou, "From linear to unrestricted and back: Type safety and the let-bang construct," 2010, unpublished manuscript, School of Electrical and Computer Engineering, National Technical University of Athens, Greece.
- [16] M. C. Rinard, "Analysis of multithreaded programs," in *Proceedings of the International Symposium on Static Analysis*, 2001, pp. 1–19.
- [17] S. Savage, M. Burrows, G. Nelson, P. Sobalvarro, and T. Anderson, "Eraser: A dynamic data race detector for multi-threaded programs," *ACM Transactions on Computer Systems*, vol. 15, 1997.
- [18] R. Shi and H. Xi, "A linear type system for multicore programming," in *Proceedings of the 13th Brazilian Symposium on Programming Languages*, August 2009.
- [19] R. Shi, D. Zhu, and H. Xi, "A modality for safe resource sharing and code reentrancy," in *Proceedings of International Colloquium on Theoretical Aspects of Computing*, ser. LNCS, vol. 6255. Natal, Brazil: Springer-Verlag, September 2010, pp. 382–396.
- [20] N. Swamy, M. Hicks, G. Morrisett, D. Grossman, and T. Jim, "Safe manual memory management in Cyclone," *Science of Computer Programming*, vol. 62, no. 2, pp. 122–144, 2006.
- [21] P. Wadler, "Linear types can change the world!" in *Programming Concepts and Methods*, M. Broy and C. Jones, Eds. Amsterdam: North Holland, 1990, pp. 347–359.
- [22] D. Walker, "Substructural type systems," in *Advanced Topics in Types and Programming Languages*, B. C. Pierce, Ed. The MIT Press, 2005, ch. 1.
- [23] D. Walker and K. Watkins, "On regions and linear types," in *Proceedings of the 6th ACM SIGPLAN International Conference on Functional Programming*, 2001, pp. 181–192.
- [24] L. Wittie and J. Lockhart, "Type-safe concurrent resource sharing," *Concurrency and Computation: Practice and Experience*, vol. 23, no. 8, pp. 767–795, 2011.

# Decomposition of SBQL Queries for Optimal Result Caching

Piotr Cybula  
Institute of Mathematics  
and Computer Science  
University of Lodz, Poland  
Email: cybula@math.uni.lodz.pl

Kazimierz Subieta  
Institute of Computer Science  
Polish Academy of Sciences, Poland  
Polish-Japanese Institute of Information  
Technology, Warsaw, Poland  
Email: subieta@ipipan.waw.pl

**Abstract**—We present a new approach to optimization of query languages using cached results of previously evaluated queries. It is based on the stack-based approach (SBA) which assumes description of semantics in the form of abstract implementation of query/programming language constructs. Pragmatic universality of object-oriented query language SBQL and its precise, formal operational semantics make it possible to investigate various crucial issues related to this kind of optimization. There are two main issues concerning this topic - the first is strategy for fast retrieval and high reuse of cached queries, the second issue is development of fast methods to recognize and maintain consistency of query results after database updates. This paper is focused on the first issue. We introduce data structures and algorithms for optimal, fast and transparent utilization of the result cache, involving methods of query normalization with preservation of original query semantics and decomposition of complex queries into smaller ones. We present experimental results of the optimization that demonstrate the effectiveness of our technique.

## I. INTRODUCTION

CACHING results of previously evaluated queries seems to be an obvious method of query optimization. It assumes that there is a relatively high probability that the same query will be issued again by the same or another application, thus instead of evaluating the query the cached result can be reused. There are many cases when such an optimization strategy makes a sense. This concerns the environments where data are not updated or are updated not frequently (say, one update for 100 retrieval operations). Examples are data warehouses (OLAP applications), various kinds of archives, operational databases, knowledge bases, decision support systems, etc.

Conceptually, the cache can be understood as a two-column table, where one column contains cached queries in some internal format (e.g. normalized syntactic query trees), and the second column contains query results. A query result can be stored as a collection of OIDs, but for special purposes can also be stored e.g. as an XML file enabling further quick reuse in Web applications. A cached query is created as a side effect of normal evaluation of user query. A transparency is the most essential property of a cached query. It implies that programmers need not to involve explicit operations on cached results into an application program. In contrast to other query optimization methods, which strongly depend on the semantics

of a particular query, the query caching method is independent of a query type, its complexity and a current database state.

Our research is done within the stack-based approach (SBA) to object-oriented query/programming languages. SBA is a formal theory and a universal conceptual frame addressing this kind of languages, thus it allows precise reasoning concerning various aspects of cached queries, in particular, query semantics, query decomposition, query indexing in the cache, and so on. We have implemented the caching methods as a part of the optimizer developed for the query language SBQL in our last project ODRA (Object Database for Rapid Application development) devoted to Web and grid applications [1]. In [2] we have described how query caching can be used to enhance performance of applications operating on grids.

There are two key aspects concerning the development of database query optimization using cached queries. The first concerns the organization of the cache enabling fast retrieval of cached queries (for optimal queries selection and rewriting new queries with use of cached results) and optimal, fast and transparent utilization of the cache, involving methods of query normalization with preservation of original query semantics (enabling higher reuse of cached queries for semantically equivalent but syntactically different queries), decomposition of complex queries into smaller ones and maintenance of assigned resources by removing rarely used results. The second problem is development of fast methods to recognize consistency of queries and automatic incremental altering of cached query results after database updates (sometimes removing or re-calculating).

In this paper we deal mainly with the first issue of the optimization method. The second aspect is widely researched in [3], [4]. The paper is organized as follows. Section II discusses known solutions that are related to the contributions of the paper. In section III we briefly present the Stack-Based Approach. Section IV shortly describes the architecture of the caching query optimizer. Sections V and VI contain the description of optimization strategies - query normalization, decomposition and rewriting rules. Section VII presents experimental results and Section VIII concludes.

## II. RELATED WORK

Cached queries remind materialized views, which are also snapshots on database states and are used for enhancing information retrieval [5], [6], [7]. The papers assume some restrictions on a query language expressions and cached structures. Such materialized views are currently implemented in popular relational database systems as DB2 and Oracle [8], [9], [10], [11]. Materialization of query results in object-oriented algebras in the form of materialized views is considered in [12] and [13]. Some solutions for view result caching at client-side in object and relational databases and for optimal combination of materialized results in cache to answer a given query are presented in [14] and [15]. In [16] and [17] a solution for XML query processing using materialized XQuery views is proposed.

There are, however, two essential differences between cached queries and materialized views. The first one concerns the scale. One can expect that there will be at most dozens of materialized views, but the number of cached queries could be thousands or millions. Such scale difference implies the conceptual difference. The second difference concerns transparency: while materialized views are explicit for software developers, cached queries are an internal feature that is fully transparent for them. Our research is just about how this transparent mechanism can be used to query optimization, assuming no changes to syntax, semantics and pragmatics of the query language itself.

New Oracle 11g database system [11] offers also caching of SQL and PL/SQL results. The cached results of SQL queries and PL/SQL functions are automatically reused while subsequent invocation and updated after database modifications. On the other hand, in opposition to our proposal, materialization of the results is not fully transparent. Query results are cached only when query code contains a comment with a special parameter `result_cache`, so the evaluation of old codes without the parameter is not optimized.

Query cache is also implemented in MySQL database [18], where only full `SELECT` query texts together with the corresponding results are stored in the cache. In the solution caching does not work for subselects and stored procedure calls (even if it simply performs a `SELECT` query). Queries must be absolutely the same - they have to match byte by byte for cache utilization, because of matching of not normalized query texts (e.g. the use of different letter case causes insertion of different queries into the query cache).

There is in Microsoft .NET query language LINQ [19] some kind of query result caching as an optimization technique for often requested queries, but it is also not transparent for programmers. They have to explicitly place the results of queries into a list or an array (calling one of the methods `ToList` or `ToArray`) and in a consequence each subsequent request of such query will cause getting its results from the cache instead of the query reevaluation.

But there is not any result caching solutions implemented in current leading commercial and non-commercial object-

oriented database systems. Most of them bases their query languages on OQL (Object Query Language) proposed as a model query language by ODMG (Open Database Management Group) [20]. Only a cache of objects is introduced in some implementations for fast access of data in a distributed database environment.

## III. OVERVIEW OF THE STACK-BASED APPROACH (SBA)

The *Stack-Based Approach* (SBA) along with its query language SBQL are thoroughly described in [21], [22], [23]. SBA assumes that query languages are a special case of programming languages. The approach is abstract and universal, which makes it relevant to a general object model. The SBQL language has several implementations - for the XML DOM model, for OODBMS Objectivity/DB, and recently for the object-oriented ODRA system [1]. SBQL is based on an abstract syntax and the principle of *compositionality*: it avoids syntactic sugar and syntactically separates as far as possible query operators. In contrast to SQL and OQL, SBQL queries have the useful property: they can be easily decomposed into subqueries, down to atomic ones, connected by unary or binary operators. The property simplifies implementation and greatly supports query optimization. The SBQL operational semantics introduces two stacks, ENVs responsible for scope control and for binding names and QRES known as query result stack for storing temporary and final query results. The two stacks architecture is the core of SBA. The syntax of SBQL is as follows:

- A single name or a single literal is an (atomic) query. For instance, `Student`, `name`, `year`, `x`, `y`, `"Smith"`, `2`, `2500`, etc., are queries.
- If  $q$  is a query, and  $\sigma$  is a unary operator (e.g. `sum`, `count`, `distinct`, `sin`, `sqrt`), then  $\sigma(q)$  is a query.
- If  $q_1$  and  $q_2$  are queries, and  $\theta$  is a binary operator (e.g. `where`, `.(dot)`, `join`, `+`, `=`, `and`), then  $q_1 \theta q_2$  is a query.
- There are not other queries in SBQL.

SBQL, unlike SQL and other query languages, avoids big syntactic and semantic patterns. Atomic queries are single names and literals. Nested queries can be arbitrarily composed from atomic and nested queries by unary and binary operators, providing they have a sense for the programmer and do not violate typing constraints. Classical query operators, such as selection, projection/navigation, join, quantifiers, etc. are also binary operators, but their semantics involves ENVs. For this reason they are called "non-algebraic" - their semantics cannot be expressed by any algebra designed in the style of the relational algebra. Below we present the exemplary operational semantics for one of the often used "non-algebraic" operator of projection (dot operator):

- 1) Initialize an empty bag (*eres*).
- 2) Execute the left subquery.
- 3) Take a result collection from QRES (*colres*).
- 4) For each element *el* of the *colres* result do:
  - a) Open new section on ENVs.

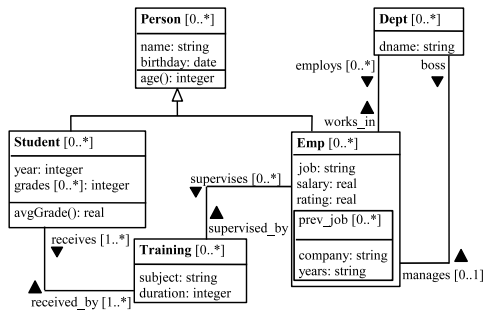


Fig. 1. Class diagram of the example database

- b) Execute function *nested*(*el*).
- c) Execute the right subquery.
- d) Take its result from QRES (*elres*).
- e) Insert *elres* result into *eres*.

5) Push *eres* on QRES.

Step 4b) employs a special function *nested* which formalizes all cases that require pushing new sections on the ENVs, particularly the concept of pushing the interior of an object. This function takes any query result as a parameter and returns a set of binders.

For the operator of selection (*where*) all steps are the same except for 4e) and a new 4f):

- e) Verify whether *elres* is a single result (if not exception is raised).
- f) If *elres* is equal to `true` add *el* to *eres*.

For the navigational join operator (*join*) the steps are:

- e) Perform Cartesian Product operation on *el* and *elres*.
- f) Insert obtained structure into *eres*.

For SBQL optimization examples presented in next sections we assume the class diagram in Fig. 1. The schema defines five classes (i.e. five collections of objects): Training, Student, Emp, Person and Dept. The classes Training, Student, Emp and Dept model students receiving trainings, which are supervised by employees of departments organizing these trainings. Person is the superclass of the classes Student and Emp. Emp objects can contain multiple complex `prev_job` subobjects (previous jobs). Names of classes (as well as names of attributes and links) are followed by cardinality numbers, unless the cardinality is 1.

#### IV. QUERY OPTIMIZER ARCHITECTURE

In most commercial client/server database systems (c.f. SQL processors) all the query processing is performed on the server. In SBA majority of query processing is shifted to the client side, to avoid server overloading and primarily to meet the orthogonal persistence principle, which implies, in particular, that a state involves persistent (server-side) and volatile (client-side) data on equal rights. Fig. 2 presents query processing architecture in SBA. Firstly, similarly to indices, the *query cache registry* is stored at the server. Hence the client-side query optimizer looks up in this registry before starts optimization and processing a given query. Secondly,

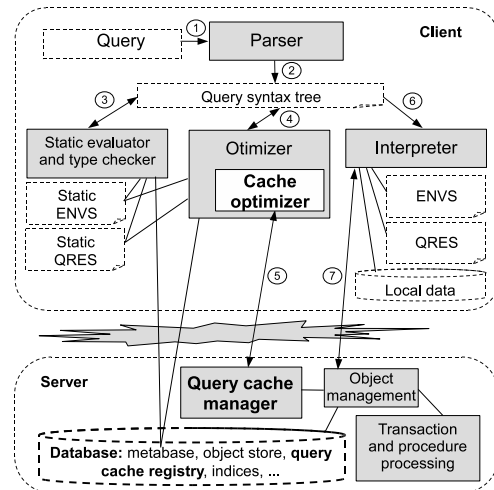


Fig. 2. Query optimization steps

in opposite to the traditional approaches, because only the client knows the form of the query and its result, the client is responsible to send the pair  $\langle \text{query}, \text{result} \rangle$  to the server in order to include it within the query cache registry. The registry indexes cached queries with search keys being query texts, normalized using some sophisticated techniques mentioned in the next section. Non-key values of the index are references to nodes storing meta-information (MB\_ID) and data (DB\_ID), mainly compiled query and results, of cached queries.

The scenario of the optimization using cached queries in query evaluation environment for SBA is as follows (step numbers as in Fig. 2):

- 1) A user sends a query to a client-side database interface.
- 2) The *parser* receives it and transforms into a *syntactic tree*.
- 3) The tree is statically evaluated for *type checking* with the use of the static stacks (ENVs and QRES) and a database schema stored in the metabase at the server-side. After successful static evaluation the nodes of the query tree are augmented with type signatures for easier optimization reasoning.
- 4) The tree is sent to the *cache optimizer* being one in a sequence of optimizers employed at the client-side database system.
- 5) The cache optimizer rewrites it using strategies presented in next two sections including the precise algorithms for the most important methods of query normalization and decomposition. The optimizer employs the server-side *cache manager* which proposes optimal matching of results cached in the query cache registry, performs proper steps for a new query caching if suggested by the optimizer and maintains cache usage statistics for optimal cache utilization and cleaning. For each new cached query the manager generates additional structures, which describe a subset of involved objects



for maintenance purposes. The system updates cached results after changes in the database [3], [4].

- 6) The optimized *query evaluation plan* is produced and sent to *query interpreter*.
- 7) The plan is evaluated by the query interpreter. Some parts of the plan rewritten by the cache optimizer suggest taking the cached results from the server-side object store instead of reevaluation of them. For new queries being candidates for caching the interpreter generates their results and sends it to the cache manager for storing at the database server.

## V. QUERY NORMALIZATION

To prevent from placing in the cache queries with different textual forms but the same semantic meaning we introduce several query text *normalization methods*. These methods are applied in a way of reconstructing a query text from early generated query syntactic tree or directly by change some nodes or their order within the tree.

**Alphabetical ordering of operands:** The method is suitable for operators, which for a succession of operands is not substantial, such as comparing operators ( $=$ ,  $\neq$ ,  $\leq$ ,  $<$ ,  $>$ ,  $\geq$ ), arithmetic operators ( $+$ ,  $-$ ,  $*$ ,  $/$ ), logical operators (`or`, `and`), operators of sum and intersection of sets, structure constructor (`struct`), i.e. a query:

Emp where salary  $\geq$  1100 or salary = 1000  
is normalized to:

Emp where 1000 = salary or 1100  $\leq$  salary

The general algorithm of the method is presented on Algorithm 1.

**Ordering of operators:** Sum and multiply operations are put before subtractions or divisions [3], i.e. an arithmetic expression is transformed as follows:

$a / b / c * d / e$

is normalized to:

$a * d / b / c / e$

**Unification of auxiliary names:** Auxiliary names used by the programmer for `as` or `group as` operator are unified, but only if such an operator doesn't finalize the evaluation of the query (it is not the root of the syntactic tree, which case is easy to recognize based on query result signature evaluated earlier by the static evaluator), i.e. a query:

```
((Emp where salary > 900) as e) join
(e.works_in.Dept as d).(e.name, d.dname)
```

is normalized to:

```
((Emp where salary > 900) as $cache_aux1)
join ($cache_aux1.works_in.Dept
as $cache_aux2).( $cache_aux1.name,
$cache_aux2.dname)
```

Algorithm for normalization of auxiliary names is presented on Algorithm 2.

---

### Algorithm 1 alphaNormalize(Q)

---

**for all** non-leaf depth-first node  $N$  in query syntax tree with the root node  $Q$  **do**

**if**  $N.op \in \{ "<", "\leq", ">", "\geq" \}$  **then**

alphaNormalize( $N.left$ );

alphaNormalize( $N.right$ );

**if**  $\text{text}(N.right) < \text{text}(N.left)$  **then**

$N.op \leftarrow ">", "\geq", "<", "\leq";$  {respectively}

swap( $N.left$ ,  $N.right$ );

**end if**

**else if**  $N.op \in \{ "=", "\neq" \}$  **then**

alphaNormalize( $N.left$ );

alphaNormalize( $N.right$ );

**if**  $\text{text}(N.right) < \text{text}(N.left)$  **then**

swap( $N.left$ ,  $N.right$ );

**end if**

**else if**  $N.op = \text{"struct"}$  **then** {possibility of more than two child nodes}

$queue \leftarrow$  empty alphabetically-sorted queue

**for all** node  $child \in N.childs$  **do**

alphaNormalize( $child$ );

$queue.push(\text{text}(child));$

**end for**

**while**  $queue \neq \emptyset$  **do**

$child \leftarrow queue.pop();$

$N.childs.push(child);$

**end while**

**else if**  $N.op \in \{ "+", "-", "*", "/", \text{"and"}, \text{"or"}, \text{"union"}, \text{"intersect"} \}$  **then**

$queue \leftarrow$  empty alphabetically-sorted queue

alphaNormalize( $N.right$ );

$queue.push(\text{text}(N.right));$

$L \leftarrow N.left;$

**while**  $L$  is non-leaf **and**  $L.op = N.op$  **do**

alphaNormalize( $L.right$ );

$queue.push(\text{text}(L.right));$

$L \leftarrow L.left;$

**end while**

alphaNormalize( $L$ );

**if**  $L.parent.op \notin \{ "-", "/" \}$  **then**

$queue.push(\text{text}(L));$

$L \leftarrow L.parent;$

$L.left \leftarrow queue.pop();$

**end if**

**while**  $queue \neq \emptyset$  **do**

$L.right \leftarrow queue.pop();$

$L \leftarrow L.parent;$

**end while**

**end if**

**end for**

---

**Algorithm 2** auxNormalize( $Q$ )

---

```

resultList ← empty sorted list;
nameMapList ← empty sorted mapping list;
for all static binder  $n(x) \in$  result signature of the query
with the root node  $Q$  do
  resultList.push( $n$ );
end for {the list remains empty for the above example query
(without any binder in its result)}
counter ← 1;
for all non-leaf depth-first node  $N$  in query syntax tree with
the root node  $Q$  do
  if  $N.op \in \{"as", "group as"\}$  then
    if  $N.name \notin$  resultList then
      nameMapList.push( $N.name$ ,
"$cache_aux" + text(counter));
       $N.name \leftarrow$  "$cache_aux" + text(counter);
      counter ← counter + 1;
    end if
  else if  $N.op$  is name expression then
    if  $N.name \in$  nameMapList then
       $N.name \leftarrow$ 
nameMapList.getMappedValue( $N.name$ );
    end if
  end if
end for

```

---

## VI. QUERY DECOMPOSITION AND REWRITING

After normalization phase query is virtually decomposed, if possible, into one or many simpler candidate subqueries. *Query decomposition* is a useful mechanism to speed up evaluating a greater number of new queries. If we materialize a small independent subquery instead of a whole complex query, then the probability of reusing of its results is risen. In addition, a simple semantic of the decomposed query reduces the costs of its updating. Each isolated subquery and finally a whole query is independently analyzed in context of the set of cached queries defined in the query cache registry and if it hasn't yet cached, it becomes a new candidate for caching.

Too simple queries (without object names or non-algebraic operators) are omitted. While analyzing, query is converted to the text form and the optimizer performs search process using query index stored in the query cache registry. If found, the tree of the query is replaced with a call of a special *cache function* parameterized with unique references to nodes of matched cached query in the metabase and the object store (these MB\_ID and DB\_ID parameters are non-key elements of cached query index mentioned earlier). Each not yet cached candidate query is also replaced with a call of the cache function - new cached query is placed into the query index. In this case a query node in the object store doesn't contain query results - it is marked as "not fully cached" and will be populated with its results while the first need of use (when the interpreter will evaluate it). The analyzing algorithm is presented on Algorithm 3.

**Algorithm 3** analyze( $Q, resultType$ )

---

```

if resultType = FULL then
  ( $MB\_ID, DB\_ID$ ) ← searchCache(text( $Q$ ));
  if  $MB\_ID \neq 0$  then {cached query found}
     $Q \leftarrow$  new tree with function call
"$cache_fun( $MB\_ID, DB\_ID$ )";
  end if
else {PARTIAL}
  if  $Q.op = "."$  and  $Q.right.op \in \{"sum", "avg", "min", "max", "count"\}$  then
    if  $Q.left.op$  is name expression and  $Q.left.name$  is
class object then
       $Q.op \leftarrow$  "join";
      ( $MB\_ID, DB\_ID$ ) ← searchCache(text( $Q$ ));
      if  $MB\_ID \neq 0$  then
        name ←  $Q.left.name$ ;
         $Q \leftarrow$  new tree with function call
"$cache_fun( $MB\_ID, DB\_ID, name$ )";
      end if
    end if
  end if
end if
if  $Q$  is not a call of function "$cache_fun" then {cached
query not found}
  ( $MB\_ID, DB\_ID$ ) ← insertCache( $Q$ ); {new cached
query}
   $Q \leftarrow$  new tree with function call
"$cache_fun( $MB\_ID, DB\_ID$ )";
end if
return  $Q$ 

```

---

**Factoring out independent subqueries:** The concept of query independence is thoroughly investigated in [3], [21], [22]. Instead of caching such a complex query as:

```
Emp where salary <
((Emp where name = "Smith").salary)
```

we isolate an internal independent query:

```
(Emp where name = "Smith").salary
```

and transform the whole query to the following form:

```
((Emp where name = "Smith").salary)
group as v).(Emp where salary < v)
```

The independent query is matched and proposed as cached query uniquely identified by its node references MB\_ID and DB\_ID, and finally the original query is rewritten to:

```
Emp where salary <
$cache_fun(MB_ID, DB_ID)
```

Algorithm of the method is presented on Algorithm 4 (assuming the existence of a special function *isIndependent* checking the independence and if need changing the query in an appropriate manner).

**Algorithm 4** `independDecompose(Q)`


---

```

if isIndependent(Q) then {query has been transformed and
starts with dot operator}
  left ← analyze(Q.left.left, FULL); {analyzing idepen-
dent query without auxiliary name}
  if left.op is function call and
  left.name = "$cache_fun" then {query cached}
    for all non-leaf depth-first node N in query syntax tree
with the root node Q.right do
      if N.op is name expression and
      N.name = Q.left.name then {auxiliary name}
        N ← left;
      end if
    end for
  end if
  Q ← Q.right;
end if
Q ← analyze(Q, FULL);

```

---

**Factoring out aggregations:** Aggregating functions (avg, min, max, sum, count) are in many cases time consuming queries. Such functions can be interpreted as virtual materialized attributes of database objects of some classes, i.e. query:

```
Dept join avg(employs.Emp.salary)
```

is cached as a group of cached queries for each Dept object instance which becomes an additional third parameter of cache function \$cache\_fun. Thus another query:

```
Emp where salary >
works_in.Dept.avg(employs.Emp.salary)
```

is decomposed by isolating cached query:

```
Dept.avg(employs.Emp.salary)
```

and rewriting the whole query as follows (Fig. 3):

```
Emp where salary >
works_in.$cache_fun(MB_ID, DB_ID, Dept)
```

Algorithm of the method is presented on Algorithm 5.

**Removing path expressions:** Reference paths finalizing query evaluation are isolated, but only if they are quickly evaluable (thanks to referential nature of object-oriented database). If a query is finalized with a sequence of navigational operators (dot) or the constructor of a structure (struct) containing such sequences, and all the objects within such expressions are unique subobjects or reference objects (with cardinality 1 or 0..1), the longest expressions fulfilling this condition are cut forming simpler independent query for caching, i.e. query:

```
(Training where count(received_by) > 12).
(subject, duration,
supervised_by.Emp.salary)
```

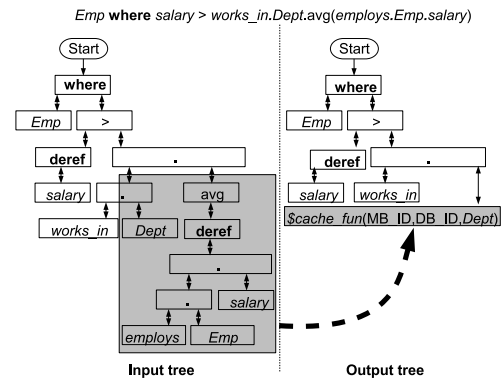


Fig. 3. Sample query optimization

**Algorithm 5** `aggDecompose(Q)`


---

```

for all non-leaf depth-first node N in query syntax tree with
the root node Q do
  if N.op = "." and N.right.op ∈ {"sum", "avg", "min",
"max", "count"} then
    if N.left.op is name expression and N.left.name is
class object then
      if N = Q then
        N ← analyze(N, FULL);
      else
        N ← analyze(N, PARTIAL);
      end if
    else if N.left.op = "." then
      if N.left.right.op is name expression and
N.left.right.name is class object then
        left ← N.left.left;
        N.left ← N.left.right;
        N.right ← analyze(N, PARTIAL);
        N.left ← left;
      end if
    end if
  end if
end for
Q ← analyze(Q, FULL);

```

---

is ended with an implicit structure constructor with a path expression. Each expression has the cardinality 1, so all expressions (and in consequence structure constructor, too) are ignored while isolating the query:

```
(Training where count(received_by) > 12)
```

and finally rewriting the input query to:

```
$cache_fun(MB_ID, DB_ID).
(subject, duration,
supervised_by.Emp.salary)
```

In case of another query:

```
(Dept where dname = "Database").
employs.Emp.prev_job
```

**Algorithm 6** pathDecompose( $Q$ )

---

```

done ← false;
if  $Q.op = "."$  then
  if  $Q.right.op$  is name expression then
    if cardinality( $Q.right.name$ ) ≤ 1 then
      pathDecompose( $Q.left$ );
      done ← true;
    else if  $Q.left.op$  is name expression
    and cardinality( $Q.left.name$ ) ≤ 1 then
      pathDecompose( $Q.left$ );
      done ← true;
    end if
  else if  $Q.right.op = "struct"$  then
    proper ← true;
    for all node  $child \in Q.right.childs$  do
      if  $child.op$  is name expression then
        if cardinality( $child.name$ ) > 1 then
          proper ← false; break
        end if
      else if  $child.op = "."$  then
        pathDecompose( $child$ );
        if  $child.op$  is name expression then
          if  $child.childs \neq \emptyset$ 
          or cardinality( $child.name$ ) > 1 then
            proper ← false; break
          end if
        else
          proper ← false; break
        end if
      else
        proper ← false; break
      end if
    end for
    if proper then
      pathDecompose( $Q.left$ );
      done ← true;
    end if
  end if
end if
if not done then
   $Q \leftarrow analyze(Q, FULL)$ ;
end if

```

---

both `prev_job` (subobject) and `employs` (reference object) attributes have cardinality 0..\*, so the optimal solution is to cache the whole query. Algorithm of the method is presented on Algorithm 6.

**Transforming queries involving logical and set-based expressions:** Thanks to the distributivity property of the selection operator in SBQL (`where`), it is possible to decompose queries with complex predicates containing some logical operators (`or`, `and`, `not`) into two or more simpler queries joined by set operators (`union`, `intersect`, `minus`) on bags of results. For instance, the complex query:

**Algorithm 7** setDecompose( $Q$ )

---

```

for all non-leaf depth-first node  $N$  in query syntax tree with
the root node  $Q$  do
  if  $N.op = "where"$  and  $N.right.op \in \{"or", "and", "not"\}$  then
    left ← empty tree;
    right ← empty tree;
     $right.op \leftarrow "where"$ ;
     $right.left \leftarrow N.left$ ;
    if  $N.right.op \in \{"or", "and"\}$  then
      left.op ← "where";
      left.left ←  $N.left$ ;
      left.right ←  $N.right.left$ ;
      right.right ←  $N.right.right$ ;
    else {"not"}
      left ←  $N.left$ ;
      right.right ←  $N.right.left$ ; {left is the only node}
    end if
    left ← analyze(left, FULL);
    right ← analyze(right, FULL);
     $N.op \leftarrow "union", "intersect", "minus"$ ; {respectively}
     $N.left \leftarrow left$ ;
     $N.right \leftarrow right$ ;
  end if
end for
 $Q \leftarrow analyze(Q, FULL)$ ;

```

---

`Emp where (job = "clerk") or (job = "consultant")`

is transformed into query:

`(Emp where job = "clerk") union (Emp where job = "consultant")`

and finally into:

`$cache_fun(MB_ID1, DB_ID1) union $cache_fun(MB_ID2, DB_ID2)`

Algorithm of the method is presented on Algorithm 7.

## VII. EXPERIMENTAL RESULTS

We have tested the performance of the optimizer by calculating response times for 100 subsequent requests using a set of queries retrieving data from database containing over 100000 objects being instances of `Dept` or `Emp` class according to the schema presented in Fig. 1. Input queries with the same semantics were syntactically different but after the normalization or decomposition they became unified. We have compared four optimization strategies: without optimization (NoCache), caching in volatile memory (TMP), caching in persistent memory (DB) and mixed caching (TMP+DB). The results presented in Fig. 4 show that in case of the TMP strategy average response time is more than 10 times shorter than response without using of the cache. In many cases, especially for more complex queries (using multi-parameterized predicates or aggregations), responses were 100 times faster.

### VIII. CONCLUSIONS AND FUTURE WORK

We have presented an approach to optimization of query execution using caching of the results of previously answered queries. Our solution addresses the stack-based approach to object-oriented query languages. The cached queries method as a tool for optimization ensures short and scalable response time to any user request types. Proper structures and strategies for fast retrieval and high utilization of cached queries results have been proposed. We have presented the architecture of the query cache optimizer for optimal query selection and rewriting new queries with the use of cached results. Methods of query normalization were developed, with preservation of the original query semantics (enabling higher reuse of cached queries for semantically equivalent but syntactically different queries). Query decomposition of complex queries into smaller ones was presented. Some experimental results of the optimization were introduced that demonstrate the effectiveness of our method.

The work on cached queries is continued. There are many open research areas concerning this optimization method. The main areas concern some additional features of SBA and SBQL not mentioned in this paper, such as inheritance and dynamic object roles. Another open issue is recognizing some parts of the cached results helpful for answering other queries and combining many cached queries while producing a result of one wider query. In general, the problem is practical rather than theoretical, hence much effort should be devoted to experiments with different strategies of caching queries and keeping in sync their stored results.

### REFERENCES

- [1] "ODRA (Object Database for Rapid Application development), Description and programmer manual." [http://sbql.pl/various/ODRA/ODRA\\_manual.html](http://sbql.pl/various/ODRA/ODRA_manual.html).
- [2] P. Cybula, H. Kozankiewicz, K. Stencel, and K. Subieta, "Optimization of distributed queries in grid via caching," in *Proceedings of the On the Move to Meaningful Internet Systems 2005, OTM GADA Workshop*, vol. 3762 of *LNCS*, pp. 387–396, Springer, 2005.
- [3] P. Cybula, *Cached Queries as an Optimization Method in the Object-Oriented Query Language SBQL*. PhD thesis, Institute of Computer Science, Polish Academy of Sciences, Warsaw, 2010. In Polish.
- [4] P. Cybula and K. Subieta, "Query optimization through cached queries for object-oriented query language SBQL," in *Proceedings of SOFSEM 2010*, vol. 5901 of *LNCS*, pp. 308–320, Springer, 2010.

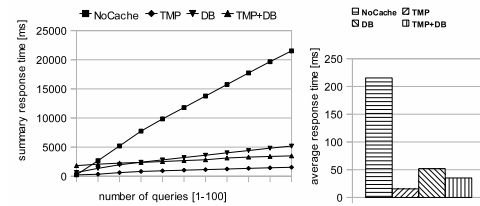


Fig. 4. Efficiency of optimization using cached queries

- [5] J. A. Blakeley, P. Larson, and W. Tompa, "Efficiently updating materialized views," in *Proc. of ACM SIGMOD*, pp. 61–71, 1986.
- [6] S. Chaudhuri, R. Krishnamurthy, S. Potamianos, and K. Shim, "Optimizing queries with materialized views," in *Proc. of Intl. Conf. on Data Engineering*, pp. 190–200, 1995.
- [7] C. M. Chen and N. Roussopoulos, "The implementation and performance evaluation of the ADMS query optimizer: Integrating query result caching and matching," in *Proc. of Intl. Conf. On Extending Database Technology*, 1994.
- [8] "IBM DB2 Universal Database SQL Reference." <http://www.ibm.com/software/data/db2/udb>, Vol. 2, Version 8, 2002.
- [9] "Faster federated queries with MQTs." [http://www.db2mag.com/db\\_area/archives/2003/q3](http://www.db2mag.com/db_area/archives/2003/q3), DB2 Magazine, Vol. 8, No. 3, 2003.
- [10] "Oracle 9i materialized views, An Oracle White Paper." <http://www.oracle.com/database>, May 2001.
- [11] "On Oracle Database 11g," Oracle Magazine, Vol. XXI, No. 5, 2007.
- [12] M. A. Ali, A. A. A. Fernandes, and N. Paton, "MOVIE: An incremental maintenance system for materialized object views," in *Proc. of Data and Knowledge Engineering*, vol. 47, pp. 131–166, 2003.
- [13] A. Kemper and G. Moerkotte, "Access support in object bases," in *Proc. of ACM SIGMOD*, pp. 364–376, 1990.
- [14] S. Dar, M. J. Franklin, B. T. Jonsson, D. Srivastava, and M. Tan, "Semantic data caching and replacement," in *Proc. of VLDB*, 1996.
- [15] H. Mistry, P. Roy, S. Sudarshan, and K. Ramamritham, "Materialized view selection and maintenance using multi-query optimization," in *Proc. of ACM SIGMOD*, pp. 307–318, 2001.
- [16] L. Chen and E. A. Rundensteiner, "ACE-XQ: A CachE-ware XQuery Answering System," in *Proc. of WebDB*, pp. 31–36, 2002.
- [17] M. El-Sayed, L. Wang, L. Ding, and E. A. Rundensteiner, "An algebraic approach for incremental maintenance of materialized XQuery views," in *Proc. of WIDM*, 2002.
- [18] "MySQL 5.4 reference manual, chapter 7.5.5: The MySQL query cache." <http://www.mysql.com>, 2009.
- [19] "LINQ: .NET Language-Integrated Query," [http://msdn.microsoft.com/pl-pl/library/bb308959\(en-us\).aspx](http://msdn.microsoft.com/pl-pl/library/bb308959(en-us).aspx), Microsoft Corporation, 2007.
- [20] R. G. G. Cattell, and D. K. Barry (eds.), "The Object Data Standard: ODMG 3.0," Morgan Kaufmann, 2000.
- [21] K. Subieta, "Theory and practice of object query languages," Polish-Japanese Institute of Information Technology, 2004. In Polish.
- [22] K. Subieta, "Stack-Based Approach (SBA) and Stack-Based Query Language (SBQL)," <http://www.sbql.pl/overview/>, 2008.
- [23] K. Subieta, C. Beeri, F. Matthes, and J. W. Schmidt, "A Stack Based Approach to query languages," in *Proc. of 2nd Springer Workshops in Computing*, 1995.

# Automated Conversion of ST Control Programs to Why for Verification Purposes

Jan Sadolewski

Rzeszów University of Technology  
 ul. W. Pola 2, 35-959 Rzeszów, Poland  
 Email: js@prz-rzeszow.pl

**Abstract**—The paper presents a prototype tool ST2Why, which converts a Behavioral Interface Specification Language for ST language from IEC 61131-3 standard to Why code. The specification annotations are stored as special comments, which are close to implementation and readable by the programmer. Further transformation with Why tool into verification lemmas, confirms compliance between specification and implementation. Proving lemmas is performed in Coq, but other provers can be used as well.

## I. INTRODUCTION

IN SOME cases control programs should be formally proved before deployment. Large control systems are usually programmed in IEC 61131-3 standard languages, such like graphical: LD (*Ladder Diagram*), FBD (*Function Block Diagram*), SFC (*Sequential Function Chart*), and textual: ST (*Structured Text*), IL (*Instruction list*). ST language is the most flexible, similar to Pascal, and it is often used by experienced programmers. It allows to declare function, function blocks and programs, called POU (*Program Organization Units*), which are components of the control application.

Contemporary program developing often uses *design by contract* method [8] and Behavioral Interface Specification Languages. JML (*Java Modelling Language*) [6] is a comprehensive example for such languages which uses the method. Such approach can be found in similar tools like Caduceus [5] and Frama C [1] for ANSI C language and in Krakatoa [7] which is also for Java.

The paper presents a proposition of Behavioral Interface Specification Language based on JML for ST language, and shows formal verification of compliance between specification and implementation. It employs multi-target open-source software Why [4] for generating Dijkstra Weakest Preconditions [3], and open-source Coq [2] as backed prover. The work presents an improved ST code verification proposed in [13], [14], which omits translation to ANSI C code and involving of Caduceus. Direct conversion from ST language to Why uses preliminary version of ST2Why, which supports functions, function blocks and programs declarations, but limits ST code to a subset composing of assignments, if statements, while loops and other function block calls.

Verification presented in paper [17] uses embedding of ST constructs in HOL (*Higher Order Logic*) terms. Function

block is visible as functional program written in HOL terms, where time is treated like additional input variable (parameter), which stays constant in each round. Its main weakness is keeping requirements in LTL (*Linear Temporal Logic*), so the verification is not simple to use by engineers. Developing user friendly language for storing specification annotations conformed with known standards (like JML) can improve applying of formal methods by the programmers.

The paper is organised as follows. Current state of verification of ST programs, and the proposition of improved version are presented in Sec. II. Next section briefly presents useful constructions of JML language adapted to ST and useful in control programs. Section IV describes direct conversion of ST code with specification annotations to Why. Code conversion is performed in three aspects: translating interface POU's into Why language functions, translating POU code into equivalent form, and translating annotations into Why form. Section V presents verification process by example of D flip-flop. The verification is processed half-automatically with prover standard tactics. Lemmas proofs are presented as tactic trees, which describe the proving method.

## II. VERIFICATION CONCEPT

Freely available software such as Why and Coq allow to be used as programs provers. These tools can prove compliance between specification and implementation and help localising mistakes and side effects of developed programs. Specifications of such programs are stored in annotations located in Why code. For ST language the specification can be saved in special comments (see sec. III), which are invisible for other ST compilers.

One of current verification method is the conversion of

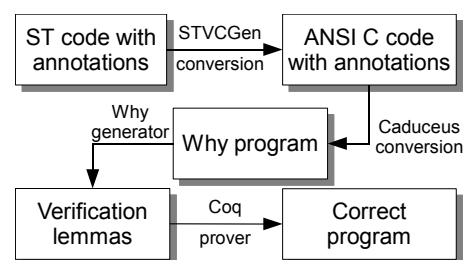


Fig. 1. Current verification of ST language control programs

The research has been supported by MNiSzW under the grant N N516 415638 in years 2010–2011

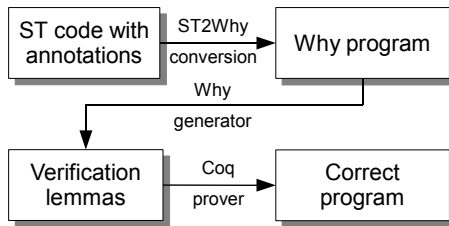


Fig. 2. Improved verification of control programs

annotated ST code to ANSI C and further conversion by Caduceus program into Why language (Fig. 1). In next step Why generator produces verification lemmas in Coq format. Lemmas can be proved half automatically with tactics. If all lemmas are proved then correctness of the code is confirmed.

The verification method uses intermediate form in ANSI C code, which is suitable for small systems. The main weakness of the approach is that one of the most popular ST types – `BOOL` has not corresponding equivalent, and it must be replaced with larger type like `char`. Operations on `char` type are treated by Caduceus like operation on numerical values. It limits access to well known laws on Boolean values like de Morgan Laws, and double negation law. The only one method to prove such numerical lemmas in Coq is using Presburger Arithmetic, but as referenced in [16] the method is slow and may explore many redundant cases. To simpler prove the verification lemma, specification clauses required special processing by the designer, for example clause `inp1=TRUE`, need to be stored as `inp1<>FALSE`. Introducing direct ST translation to Why code (Fig. 2) with ST2Why tool, such disadvantages can be avoided.

The Why language has build-in three types: `bool`, `real`, and `int` which are sufficient for typical programming. Additional user types can be declared, if necessary, with basic arithmetic operations on them. Why contains a collection of libraries which are contributed into provers at install time. They are used as support of verification lemmas and, in some standard cases, can made proving simpler and faster.

### III. BEHAVIORAL INTERFACE SPECIFICATION LANGUAGE FOR ST

The main purpose for introducing the BISO language is to define behaviour of parts of developed code. Software developing with design by contract use such languages, which can be seen in Eiffel [9], Why, and JML code. The first two languages use build-in constructions for storing specification clauses, the last one uses special kind of comments beginning with '@' character. Such method is also used for storing specification in ST language.

Specification clauses are stored as assertions. An assertion is a part of code composed of conditional Boolean expression, which evaluated at time and order of its execution must be satisfied. In design by contract two assertions are commonly used: `requires` to denote preconditions, and `ensures` for postconditions. These assertions must be kept near developed

TABLE I  
ADAPTATION OF JML IN ST LANGUAGE

Clause type	Standard JML	ST adaptation	Scope
Assertions	<code>assert</code>	<code>assert</code>	instruction
	<code>ensures</code>	<code>ensures:</code>	local
	<code>requires</code>	<code>requires:</code>	local
Localise modifiers	<code>\at</code>	<code>\at or at</code>	instruction
	<code>\old</code>	<code>\old</code>	instruction
Quantifiers	<code>\exists</code>	<code>\exists</code>	mixed
	<code>\forall</code>	<code>\forall</code>	mixed
Invariant	<code>invariant</code>	<code>invariant:</code>	instruction
	<code>label</code>	<code>label:</code>	instruction
Declarations	<code>logic</code>	<code>logic:</code>	global
	<code>ghost</code>	<code>ghost:</code>	local
	<code>predicate</code>	<code>predicate:</code>	global
	<code>axiom</code>	<code>axiom:</code>	global
Function return value	<code>\result</code>	<code>\result or function_name</code>	local
Operations	<code>set</code>	<code>set:</code>	instruction
	<code>assigns</code>	<code>assigns:</code>	local
W-F iteration	<code>variant</code>	<code>variant:</code>	instruction

code, as special comments like mentioned above. They express conditions, which must be satisfied when given subroutine is called and guaranteed at its termination.

Function blocks and programs from IEC standard are similar to lightweight Java objects, so using JML as a base of BISO for ST seems motivated. It is natural that only some subset of JML standard can be applied in control programs, so other features of that will not be described. The adaptation of JML for ST language, called *assertional extension*, is presented at Tab. I and grouped according to clause types. Each clause has its own affection scope. Scope *instruction* means that the clauses can be placed when instructions (or sometimes expressions) are expected. Scope *local* means clauses defined for whole POU, and *global* for whole project (configuration in IEC standard). *Mixed* denotes clause whose use depends on context.

The adaptation of JML for ST has been described in more details in [13]. Verification clauses, with different scope than *global*, are located inside corresponding unit. For example annotation clause of function block is written after identifier with the name of the block. The clause must contain at least `ensures` section, but often involves `requires` and `assigns` – especially when annotated POU is a program which modifies global variables. The `\result` or function name can be used in `ensures` section to access function return value. Modifier `\old` represents variable value at beginning of execution, and `\at` at specified location in the code which can be declared with `label` clause.

Additional functions not appearing in the code can be obtained with `logic` clause, similar local variables for specification can be defined by `ghost` clause and operated by `set` clause. The `predicate` declares additional logic function which returns Boolean value. The `axiom` generates new axiom which can be used by the prover. Quantifiers appear in declarations of loop invariants, declared with `invariant` clause. To examine if loops are well founded `variant` clause is employed.



IV. CONVERSION ST TO WHY

As indicated in Sec. II conversion POU from ST language into Why form is needed to use open source tools for program verification. The ST2Why tool provides the conversion, which is based on ST compiler included in CPDev package [12]. The parser is built according to top-down scheme with syntax-directed translation. It recognises of ST code shape and produces correspond Why code. In addition to ST code translating, the parser also collects annotations and emits them into valid positions of Why language.

Code translation is performed in three aspects:

- converting POU interfaces,
- converting body code,
- converting specification.

The first aspect concentrates on POU's translating into Why language functions. If POU to be converted is a function, then translation seems obviously, except return value, which must be declared locally to conform to the code. Situation is more complicated with function block or program. Function block is translated into Why function in the following way:

- block inputs are converted into function parameters,
- block outputs become function parameters, but declared as reference,
- local variables are also declared as reference parameters.

An ST program is translated into Why function as follows:

- global variables are converted into global reference parameters,
- local variables become function parameters, but declared as reference,
- local function block instances are ignored, but their reference parameters are also declared like local variables.

Such interfaces conversion is illustrated in Fig. 3. The IEC 61131-3 defines 20 standard data types, but currently the conversion process is limited for the basic ones `BOOL`, `INT`, `REAL` and `TIME`. Boolean type is translated into `bool` type in Why and floating point type `REAL` is translated into single `real` type. Due to early developing stage of ST2Why converter, all remaining fixed point types are translated into single `int` type in Why.

The second aspect is to convert instruction code into valid Why form. Most of Why common arithmetic operations are available as functions with type name and operation name. As seen at Fig. 4a the sum calculation of two integer values involves `int_add` function. Boolean expressions (Fig. 4b) can be converted without complications, like the `if` statement from Fig. 4c. More effort is necessary with `while` loop conversion. The Why language is functional, so loops can work only on pointers, which are forbidden in ST language. It requires redeclaration as pointers of those variables, which are used in condition expression and are assigned in loop body. Currently none of the remaining loops in ST language (`FOR`, `REPEAT`, etc.) are supported.

Calls of function blocks require more effort, because values of local and output variables from previous execution cycle must be preserved. As shown before in Fig. 3, the program

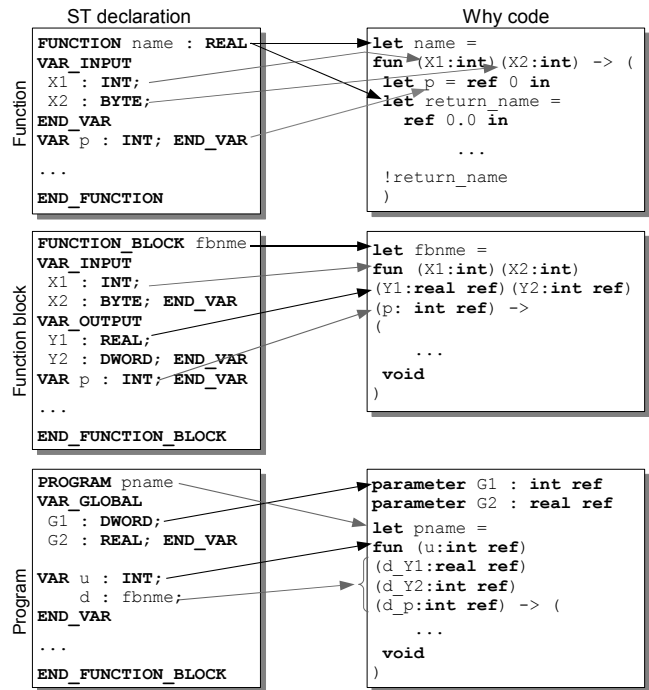


Fig. 3. Interface conversion of POU from ST to Why language

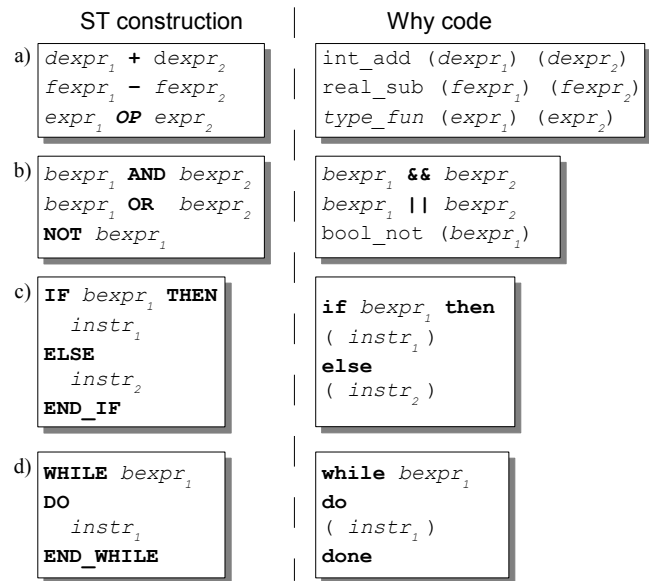


Fig. 4. ST source code to Why conversion

`pname` uses a hypothetical function block `fbnme` with the instance called `d`, so additional inputs (beginning with `d_`) have been also declared. Call of the instance `d` in ST code and the translation to Why is presented in Fig. 5. The single variable `d` does not exist here, but it is replaced by corresponding arguments of the converted program. Such approach is necessary to combine complex data type, like function block interface, from elementary data types.

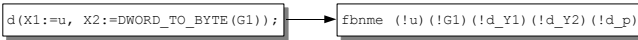


Fig. 5. Function block call conversion

The third aspect of conversion it to change annotations describing a POU in ST language into equivalent form in Why with necessary modifications. Conversion of annotations affect shape and position of its source. The `REQUIRES` clause is enclosed in `{}` brackets, with removed introducing word and following colon, and moved to position after the arrow (`->`) sign. Finally, separating semicolon is removed (Fig. 6). The `ENSURES` clause is moved after function implementation, and similar shape modification are performed.

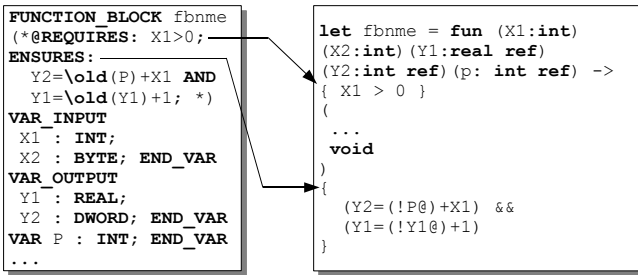


Fig. 6. Converting ST assertional extension

The composition of those aspects produce coherent Why code, which can be handled by Why tool to produce verification lemmas.

## V. VERIFICATION EXAMPLE

The verification example will be presented on developing of D flip-flop. It is a elementary block in control applications, which preserves state of one electric wire. Its symbol and time plot are given at Fig. 7a and Fig. 7b.

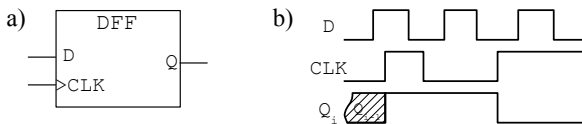


Fig. 7. D flip-flop; a) symbol, b) time plot

The *design by contract* process begins from describing function block requirements by the designer. As it can be seen at Fig. 7b, the output signal `Q` is only changing when raising edge on `CLK` input is detected, otherwise output signal remains unchanged. Detecting raising edge requires additional variable, here called `PCKL`, which holds value `CLK` from previous program cycle. It leads to following ST code interface:

```
FUNCTION_BLOCK DFF
(*@ENSURES:
((CLK=FALSE) ==> (Q=old(Q))) AND
((old(PCLK)=FALSE AND CLK=TRUE)
==> (Q=D)) AND
((old(PCLK)=TRUE AND CLK=TRUE)
==> (Q=old(Q))); *)
```

```
VAR_INPUT D : BOOL; CLK : BOOL; END_VAR
VAR_OUTPUT Q : BOOL; END_VAR
VAR PCLK : BOOL; END_VAR
```

```
END_FUNCTION_BLOCK
```

Analysing `CLK` and `PCLK` input states, one can notice that only `CLK` equal to false determines unchanging the `Q`. It produces following part of ensures expression `CLK=FALSE ==> Q=old(Q)`. Second part `old(PCLK)=FALSE AND CLK=TRUE ==> Q=D` can be taken from specification in textual form. The third part is taken from remaining input states which have not been described, so it is `old(PCLK)=TRUE AND CLK=TRUE ==> Q=old(Q)`. Because all of the inputs are fully qualified in the specification, and work of the D flip-flop is not restricted, then `REQUIRES` clause remains empty.

In the second stage of the design by contract the developer produces an implementation from the time plot according to given interface and specification:

```
IF (NOT PCLK) AND CLK THEN Q := D; END_IF;
PCLK := CLK;
```

and transforms them with rules mentioned in Sec. IV, or automatically with `ST2Why` tool into following format:

```
let dff = fun (D:bool) (CLK:bool)
(Q:bool ref) (PCLK:bool ref) -> {}
((if ((not !PCLK) && (CLK))
then (Q := D) else void;
PCLK := CLK)
{ (CLK=false -> Q=Q@) and
(PCLK=false and CLK=true -> Q=D) and
(PCLK=true and CLK=true -> Q=Q@)
}
```

From that form after Why usage two verification lemmas are obtained:

```
Lemma dff_po_1:forall(D CLK PCLK Q:bool),
forall (HW_1: PCLK=false /\ CLK=true),
forall (Q0: bool), forall (HW_2: Q0=D),
forall (PCLK0: bool),
forall (HW_3: PCLK0 = CLK),
(((CLK=false -> Q0=Q) /\ ((PCLK=
false /\ CLK=true -> Q0=D) /\
((PCLK=true /\ CLK=true -> Q0=Q))).
```

```
Lemma dff_po_2:forall(D CLK PCLK Q:bool),
forall (HW_4: PCLK=true \/ PCLK=false
/\ CLK=false), forall (PCLK0: bool),
forall (HW_5: PCLK0=CLK),
(((CLK=false -> Q=Q) /\ ((PCLK=false
/\ CLK=true -> Q=D) /\
((PCLK=true /\ CLK=true -> Q=Q))).
```

The proofs of the lemmas can be performed in Coq via a set of tactics. The tactic is a prover command which:

- transforms lemma into hypotheses and goals, or
- splits goal into subgoals, or
- indicates agreement between hypothesis and goal, or
- contradicts two hypotheses.

Verification begins from the whole lemma which is used as goal in empty context. As the first lemma `dff_po_1` is taken for proving. Tactic `intros` introduces new hypotheses from

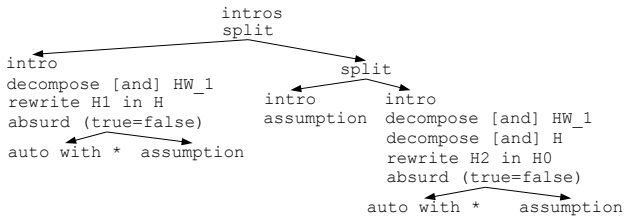


Fig. 8. Coq proof of Lemma 1

goal to context (Fig. 8). The goal will change into three conjunctions, so `split` tactic is necessary to divide goal into two subgoals. The first subgoal contains implication, so `intro` tactic is used to introduce `H0` hypothesis. Further it could be observed that part of hypothesis `HW_1` (`CLK=true`) is in contradiction with `H` (`CLK=false`). It leads to the following method: changing the one of the occurrences of variable with opposite value and proving as hypotheses contradiction. To extract a part of the hypothesis with conjunction, `decompose` tactic is used. It produces additional hypotheses in context with separated parts. Value from `H1` hypothesis is applied into `H` with tactic `rewrite H1 in H`. The hypothesis `H` become contradiction (`true=false`), so tactic `absurd (true=false)` is applied. Tactic `absurd` produces also two subgoals, first with negated contradiction, and second with the contradiction itself, so `auto with *` proves the first one (`true<>false`), which is handled automatically by internal libraries, and `assumption` proves the second one, due to existing such hypothesis in context.

After that prover returns into subgoal which was left after the first `split` command. Because it is also conjunction, so another `split` is necessary. Current first subgoal can be proved with `intro` and `assumption` which matches goal with `HW_2` hypothesis. In the last one goal after `intro` another contradiction in hypotheses can be found. Variable `PCLK` from part of hypothesis `H` cannot be equal to `true` and also equal to `false` in part of `HW_1` hypothesis. It leads to mentioned verification method with tactic `absurd`. Detailed proof of lemma `dff_po_` is presented as tree in Fig. 8. To obtain a list of proving commands for Coq simple *in-order* tree walk (begin from root node, then its left child tree, and next right child tree) should be performed.

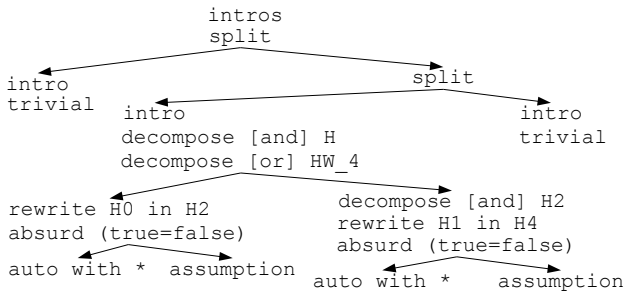


Fig. 9. Coq proof of Lemma 2

The second lemma (`dff_po_2`) can be proved in similar way. The proof begins from `intros` and `split` tactics (Fig. 9). The first subgoal is a implication, so `intro` tactic is applied. After that goal reduces to `Q=Q` form, which is very simple and can be proved with `trivial` tactic. The second subgoal is still conjunction so another `split` is required. After introducing new hypothesis `H` with `intro` tactic, current first subgoal have contradicted hypotheses. First decomposition of first hypothesis is performed with tactic `decompose [and] H`, and due to disjunction in `HW_4` the tactic `decompose [or] HW_4` is used. The last one tactic splits goal for two subgoals, each one with separated part of disjunction as hypothesis. In current first subgoal assignment with `rewrite` tactic is needed and `absurd` command can be applied. In the second subgoal additional `decompose` is necessary before rewriting, and `absurd` applying. The remaining subgoal can be proved exactly like the first one with `intro` and `trivial` tactics.

Proving all verification lemmas confirms compliance between specification and implementation. Developed code formally satisfies designer guidelines, and the contract has been fulfilled.

Presented verification tactics are not comprehensive for all programs, especially when program contains integer or floating point variables. For that lemmas more complex tactics such like `omega`, `ring` and `Fourier` are needed. Complete reference for all build-in tactics in Coq can be found in [16]. It may be helpful for an inexperienced users, but in some cases `intuition` tactic may prove the goal automatically or to present reason for which current goal cannot be proved.

VI. SUMMARY

The method of proving programs written in ST language with BISL extension has been presented. The language extension is stored as special comment inside the function, function block or program being verified. It accords to JML language which is commonly used in design by contract developing approach. Conversion with ST2Why and Why tools produce verification lemmas. Verification of compliance between specification and implementation can be performed in a few provers, but here only simple build-in Coq tactics have been used. Other provers like PVS [11] or Mizar [10] can be also used, it requires only one additional parameter in Why call, which will change the output shape of lemmas.

Future work will concentrate on transforming remaining clauses of ST language into Why code (like REPEAT and FOR loops, CASE statements), and on introducing remaining data types into Why, conformed with ST language types. It may require to develop additional Why libraries, where their logical definitions will be stored.

REFERENCES

[1] P. Baudin, P. Cuoq, J. Ch. Filliâtre, C. Marché, B. Monate, Y. Moy, V. Prevosto, “ACSL: ANSI/ISO C Specification Language”, <http://frama-c.cea.fr>, 2011.  
 [2] Y. Bertot, P. Castéran, *Interactive Theorem Proving and Program Development*, Springer-Verlag, Berlin Heidelberg, 2004.

- [3] E. W. Dijkstra, *A Discipline of Programming*, Prentice-Hall Inc., 1976.
- [4] J. Ch. Filliâtre, "The Why verification tool. Tutorial and reference manual", <http://www.lri.fr>, 2011.
- [5] J. Ch. Filliâtre, T. Hubert, C. Marché, "The Caduceus verification tool for C programs", <http://caduceus.lri.fr>, 2008.
- [6] G. T. Leavens, A. L. Baker and C. Ruby, "JML: a Notation for Detailed Design", *Behavioral Specifications of Businesses and Systems*. 1999.
- [7] C. Marché. "The Krakatoa verification tool for Java programs. Tutorial and reference manual", <http://proval.lri.fr>.
- [8] B. Meyer, "Applying design by contract", *Computer*, vol. 25, no. 10, pp. 40-51, 1992.
- [9] B. Meyer, *Eiffel: the language*. Object-Oriented Series, Prentice Hall New York, 1992.
- [10] M. Muzalewski, *An outline of PC Mizar*, Foundation Philippe le Hodey, Brussels, 1993.
- [11] S. Owre, N. Shankar, J. M. Rushby, D. W. J. Stringer-Calvert, "PVS system guide", SRI International, 2001.
- [12] D. Rzońca, J. Sadolewski, A. Stec, Z. Świder, B. Trybus, L., "A Control Program Developer". XXI MicroCAD International Scientific Conference, Miskolc, March 2009, pp. 49-54.
- [13] J. Sadolewski, Assertion extension in ST language of IEC 61131-3 standard for control systems dynamic verification, *Pomiary Automatyka Robotyka*, no 2, pp. 305-314, 2011 (in Polish).
- [14] J. Sadolewski, An introduction to verification of simple programs in ST language with Coq, Why and Caduceus tools, *Metody Informatyki Stosowanej*, vol. 19, no 2, pp. 121-138, 2009 (in Polish).
- [15] J. Sadolewski, Conversion of ST Control Programs to ANSI C for Verification Purposes, *e-Informatica Software Engineering Journal*, (in review).
- [16] The Coq Development Team, *The Coq Proof Assistant Reference Manual*, Ecole Polytechnique, INRIA, Universit de Paris-Sud, <http://coq.inria.fr>, 2010.
- [17] N. Völker, B. J. Krämer, "Modular Verification of Function Block Based Industrial Control Systems", in *Proceedings of Joint 24th IFAC/IFIP Workshop on Real-Time Programming and the 3rd International Workshop on Active and Real-Time Database Systems*, Schloß Dagstuhl, Germany, May 30th – June 2nd, 1999.

# Implementing Attribute Grammars Using Conventional Compiler Construction Tools

Daniel Rodríguez-Cerezo Antonio Sarasa-Cabezuelo, José-Luis Sierra  
Facultad de Informática. Universidad Complutense de Madrid. 28040 Madrid, Spain  
Email: drodriguez@fdi.ucm.es, {asarasa, jlsierra}@fdi.ucm.es

**Abstract**—This article describes a straightforward and structure-preserving coding pattern to encode arbitrary non-circular attribute grammars as syntax-directed translation schemes for bottom-up parser generation tools. According to this pattern, a bottom-up oriented translation scheme is systematically derived from the original attribute grammar. Semantic actions attached to each syntax rule are written in terms of a small repertory of primitive *attribution* operations. By providing alternative implementations for these attribution operations, it is possible to plug in different semantic evaluation strategies in a seamlessly way (e.g., a *demand-driven* strategy, or a *data-driven* one). The pattern makes it possible the direct implementation of attribute grammar-based specifications using widely-used translation scheme-driven tools for the development of bottom-up language translators (e.g. YACC, BISON, CUP, etc.). As a consequence, this initial coding can be subsequently refined to yield final efficient implementations. Since these implementations still preserve the ability of being extended with new features described at the attribute grammar level, the advantages from the point of view of development and maintenance become apparent.

## I. INTRODUCTION

ATTRIBUTE grammars, which were introduced by Donald E. Knuth [8] as an extension of context-free grammars for describing the syntax and semantics of context-free languages, are widely-used as a high-level specification method for the first stages of the design and implementation of a computer language [1][11].

In order to make an attribute grammar – based specification executable, it is possible to use one of the many specialized tools supporting the formalism (see, for instance, [3] [10][11]). However, regardless the realized advantages of these tools, in practice, traditional implementations of language processors are rarely based on artifacts directly generated from attribute grammars. On the contrary, attribute grammars are taken as initial specifications of the tasks to carry out, while final implementations are usually achieved by using scanner and parse generators (e.g., ANTLR, CUP, Flex, Bison...), general-purpose programming languages, or a suitable combination of both techniques [1]. The process of transforming the initial specification in a final implementa-

tion is usually ill-defined, and usually depends solely on the programmer's art, who many times discards formal specifications while directly hacks the final implementation. It seriously hinders systematic development and maintenance of language processors.

In order to bridge the gap between attribute grammar-based specifications and final implementations, we propose to articulate the language processor development process as the explicit transformation of the initial attribute grammar-based specification to the final implementation. According to our proposal, the first step to convey during the implementation stage is to explicitly encoding the attribute grammar in the input language of the development tool (usually, a parse generator like Bison or CUP). It will make it possible to yield an initial running implementation, which subsequently can be refined to achieve greater efficiency. In addition, since the refined implementation still supports the explicit incorporation and subsequent refinement of attribute grammar – based features, the incremental development and subsequent maintenance of the language processor can be largely facilitated.

This paper is focused on the first step of our proposal, i.e. how to code an attribute grammar in terms of the input language supported by a conventional parse generation tool. More precisely, we will focus on bottom-up parse generators of the YACC and CUP type. Unlike to works in LR-attributed grammars [2] and similar approaches (e.g., [6]), our approach will support the implementation of arbitrary non-circular attribute grammars. In addition, the encoding pattern will be independent of the final evaluation style chosen. Indeed, attribute grammars will be coded using a small repertory of *attribution* operations. Finally, by providing alternative implementations for these operations, it will be possible to set up the semantic evaluation style finally used.

The structure of the rest of the paper is as follows: section II describes the encoding pattern itself. Sections III and IV show how to plug in different evaluation styles by providing suitable implementations of the attribution operations. Finally, section V concludes the paper and outlines some lines of future work.

## II. ENCODING THE ATTRIBUTE GRAMMARS

In this section we introduce our coding pattern. In order to make it as general as possible, we will not compromise with any particular generation tool, and we will use pseudo-code comprising very simple and standard procedural interfaces and imperative constructs. In addition, we will use a YACC-like notation [1] to refer to semantic values of symbols in the parse stack. In subsection II.A we describe the basic attribution operations allowed in the syntax rules' semantic actions. In sections II.B and II.C we describe the coding pattern itself, and in section II.D we exemplify it.

TABLE I.  
ATTRIBUTION OPERATIONS

Operation	Intended Meaning
mkCtx( $n$ )	It creates and initializes a list of $n$ attribute instances for a symbol in the parse stack.
mkDep( $a_0, a_1$ )	It sets a dependency between two attribute instances. Indeed, it declares the attribute instance $a_0$ depends on the attribute instance $a_1$ .
inst( $a, f$ )	It <i>instruments</i> the attribute instance $a$ by establishing $f$ as the semantic function to be applied during evaluation ( $f$ is actually an integer identifier of such a semantic function)
release( $as$ )	It invokes garbage collection on the list of attribute instances $as$ .
release( $a$ )	It invokes garbage collection on the attribute instance $a$
set( $a, val$ )	It fixes the value of the attribute instance $a$ to $val$ .
val( $a$ )	It retrieves the value of the attribute instance $a$ .

### A. Attribution operations

Our coding pattern is largely based on the explicit description of the attribution structure of each grammar rule. For this purpose, we introduce the repertory of basic *attribution* operations outlined in Table 1. This table shows both the procedural interfaces of the operations and their intended meanings.

As such a description makes apparent, the purpose of these operations is to provide the developer with the necessary tools to describe how the *attribute dependency graph* associated with a sentence can be built conforming this sentence is analyzed by the parser. In addition, it also lets the developer indicate the semantic functions for computing each attribute instance. It does not necessarily mean the graph must be fully stored in memory: depending on the actual implementation of the attribution operations, it will be possible to optimize, to a greater or lesser extent, the heap overhead (see sections III and IV).

### B. Writing the translation scheme

The actual encoding of the attribute grammar requires writing a translation scheme describing how to build the aforementioned attribute dependency graph for each processed sentence. It can be done in a straightforward way by applying the following guidelines to each rule of the attribute grammar:

- First at all, we need to create the semantic value for the rule left-hand side (LHS). It is done by using an `mkCtx` operation. We only need to indicate the number of semantic attributes for the LHS.
- Next, we need to describe the dependencies between the attribute instances. Such dependencies are directly determined by examining the semantic equations, and they must be stated using the `mkDep` operation.
- Once it has been done, it is necessary to *instrument* the synthesized attribute instances in the rule's LHS, as well as the inherited attribute instances of the symbols in the rule's right-hand side (RHS). Once more, the code is straightforward: an `inst` operation for each equation. Notice we need to encode the semantic functions with integer identifiers, which can be interpreted by a *semantic function manager* (see subsection II.C).
- Finally, we need to release the attribute instance lists for the symbols in the rule's RHS.

Concerning the allocation of lexical attribute instances, it must be performed by the scanner, which will return the corresponding attribute instance list using a suitable field in the token. Also, notice the underlying context-free grammar must belong to the kind of grammars supported by the parser generation tool. Since we are using bottom-up parse generation translators, which usually support LALR(1) grammars [1], in practice it does not suppose a serious limitation.

### C. Writing the Semantic Function Manager

In addition to the translation scheme, we need to code another auxiliary component, the *semantic function manager*, supporting the execution of the semantic functions. This component can be conceived as a procedure that, taking the semantic function's identifier and the sequence of attribute instances as input, returns the result of applying the function to the attribute instances.

### D. Example

To illustrate the pattern we will consider the attribute grammar in Fig. 1. It models a very simple processor that makes it possible to evaluate simple arithmetic expressions involving addition and multiplication. To store the value we use a `val` synthesized attribute. Additionally, the processor can use a memory of predefined constants, which is propagated using an `env` inherited attribute.

Fig. 2 shows the translation scheme for this attribute grammar. In order to make the encoding more readable, we intro-

duce some constants for attribute instance indexes and for semantic function integer identifiers.

```

E ::= E + T
E1.env↓ = E0.env↓
T.env↓ = E0.env↓
E0.val↑ = E1.val↑ + T.val↑
E ::= T
T.env↓ = E.env↓
E.val↑ = T.val↑
T ::= T * F
T1.env↓ = T0.env↓
F.env↓ = T0.env↓
T0.val↑ = T1.val↑ * F.val↑
T ::= F
F.env↓ = T.env↓
T.val↑ = F.val↑
F ::= n
F.val↑ = toNum(n.lex↑)
F ::= id
F.val↑ = valOf(F.env↓, id.lex↑)
F ::= ( E )
E.env↓ = F.env↓
F.val↑ = E.val↑

```

Fig 1. Example attribute grammar. To improve readability, synthesized attribute occurrences are suffixed with ↑, and inherited occurrences are suffixed with ↓.

In this translation scheme, the first rule (which is not present in the original grammar) plays the role of initiating the processing. Indeed:

- It sets `env` in the root of the parse tree (we suppose the environment is returned by the `getEnv` external procedure).
- Then, it prints the value of `val` in such a root.
- Finally, it releases the root's attribute instance list.

The other rules are obtained by a step-by-step application of the guidelines described in the previous subsection. For instance, the encoding (shadowed in Fig. 2) of the first rule of the attribute grammar (shadowed in Fig. 1) is obtained as follows:

- Since `E`, the rule's LHS, has two semantic attributes (`env` and `val`), we need to invoke `mkCtx` with 2 as the number of attributes to be allocated. Notice that the resulting attribute instance list is assigned to `$$`, which in YACC-like notation is the pseudo-variable for the semantic value of the rule's LHS.
- From the first equation, we get  $E_1.env$  depends on  $E_0.env$ . This dependency is declared by `mkDep($1[env], $$[env])`, since (i) `$1` refers, in YACC-like notation, to the semantic value of  $E_1$ , and (ii) `$$` refers, as said before, to the semantic value of  $E_0$ .
- In a similar way, the other three `mkDep` actions are derived from the other two equations. Notice that the third equation yields two `mkDep` actions, since, according to it,  $E_0.val$  depends on two different attributes:  $E_1.val$  and  $T.val$ .
- In their turn, each equation yields an `inst` action. For doing so, firstly we need to identify the semantic

function used in the equation. It can require some intermediate analysis. For instance, to make the semantic function apparent,  $E_1.env↓ = E_0.env↓$  must be actually read as  $E_1.env↓ = \lambda_v(v)E_0.env↓$ . Thus, we can assign an integer code to this  $\lambda_v(v)$  semantic function (in Fig. 2, this code is given by the `IDEN` constant). A similar technique can be used for equations sides involving more complex expressions. For instance,  $E_0.val↑ = E_1.val↑ + T.val↑$  can be actually read as  $E_0.val↑ = \lambda_{v_0}(\lambda_{v_1}(v_0+v_1))E_1.val↑ + T.val↑$ , which leads to identify  $\lambda_{v_0}(\lambda_{v_1}(v_0+v_1))$  as the semantic function (it is identified by the `ADD` constant in Fig. 2).

```

def env=0; def val=1;
def IDEN=0; def ADD=1; def MUL=2;
def TONUM=3; def VALOF=4;
S ::= E {
  set($1[env], getEnv());
  print(val($1[val]));
  release($1);
}
E ::= E + T {
  $$ := mkCtx(2);
  mkDep($1[env], $$[env]); mkDep($3[env], $$[env]);
  mkDep($$[val], $1[val]); mkDep($$[val], $3[val]);
  inst($1[env], IDEN);
  inst($3[env], IDEN);
  inst($$[val], ADD);
  release($1); release($3);
}
E ::= T {
  $$ := mkCtx(2);
  mkDep($1[env], $$[env]); mkDep($$[val], $1[val]);
  inst($1[env], IDEN);
  inst($$[val], IDEN);
  release($1);
}
T ::= T * F {
  $$ := mkCtx(2);
  mkDep($1[env], $$[env]); mkDep($3[env], $$[env]);
  mkDep($$[val], $1[val]); mkDep($$[val], $3[val]);
  inst($1[env], IDEN);
  inst($3[env], IDEN);
  inst($$[val], MUL);
  release($1); release($3);
}
T ::= F {
  $$ := mkCtx(2);
  mkDep($1[env], $$[env]); mkDep($$[val], $1[val]);
  inst($1[env], IDEN);
  inst($$[val], IDEN);
  release($1);
}
F ::= n {
  $$ := mkCtx(2);
  mkDep($$[val], $1[lex]);
  inst($$[val], TONUM);
  release($1);
}
F ::= id {
  $$ := mkCtx(2);
  mkDep($$[val], $$[env]); mkDep($$[val], $1[lex]);
  inst($$[val], VALOF);
  release($1);
}
F ::= ( E ) {
  $$ := mkCtx(2);
  mkDep($$[val], $2[val]);
  mkDep($2[env], $$[env]);
  inst($2[env], IDEN);
  inst($$[val], IDEN);
  release($2);
}

```

Fig 2. Encoding of the attribute grammar in Fig. 1.

- Finally, we include a `release` action for each symbol in the rule's RHS having semantic attributes.



Finally, in addition to the translation scheme, we need to provide a suitable semantic function manager. It is depicted by the pseudo-code in Fig. 3. Basically, it is a dispatcher that, according to the function's integer identifier, applies the actual function on the sequence of semantic attribute instances<sup>1</sup>.

```

procedure exec(FUN, ARGS) {
case FUN of
  IDEN →
    return val(ARGS[0]);
  ADD →
    return val(ARGS[0]) + val(ARGS[1]);
  MUL →
    return val(ARGS[0]) * val(ARGS[1]);
  TONUM →
    return toNum(val(ARGS[0]));
  VALOF →
    return valOf(val(ARGS[0]), val(ARGS[1]));
end case
}

```

Fig 3. Semantic function manager for the attribute grammar in Fig. 1.

### III. INCORPORATING A DEMAND-DRIVEN EVALUATION FRAMEWORK

In order to make possible the execution of the encodings proposed in the previous section, we need to implement the basic attribution operations. In this section we describe a straightforward implementation supporting a *demand-driven* evaluation style (see, for instance [5] [9]). In this implementation, semantic evaluation starts once the sentence has been completely parsed. In this point, there is an in-memory representation of the part of the dependency graph required for performing semantic evaluation. During evaluation, the values of the attribute instances will be calculated only when they are required.

In the following subsections we describe the resulting framework explaining how attribute instances are represented (subsection III.A). Then, we specify how the attribution operations work (subsection III.B). Finally, we illustrate the complete framework with an example (subsection III.C). For the sake of simplicity, we will ignore the detection of potential circularities in the underlying dependency graphs, although it would not be difficult to extend the framework to support it.

#### A. Representing the instances of the semantic attributes

The instances of the semantic attributes can be conceived as records. Table 2 outlines the fields required together with their intended purposes. Thus, this representation makes it possible to build a dependency structure in which:

- Each attribute instance points to those attribute instances required to compute it (in a similar way to the *reversed* dependency graph used in [5]).
- In addition, it explicitly stores the identifier of the semantic function to be used in this computation.

<sup>1</sup>Remark that, by the sake of generality, we are intentionally using a minimal set of programming language features. Indeed, by using a more sophisticated programming paradigm (e.g., a language equipped with higher-order features), it could be possible to give more elegant solutions to these basic conceptualizations.

TABLE II.

STRUCTURE OF ATTRIBUTE INSTANCES IN THE DEMAND-DRIVEN EVALUATION FRAMEWORK.

Field	Purpose	Initial value
value	It keeps the value of the instance of the semantic attribute.	$\perp$
available	A boolean flag that indicates whether the value is available.	false
deps	It keeps the links to those attribute instances required to compute the value.	The empty list
semFun	It stores the integer code of the semantic function required to compute the value.	$\perp$
refcount	A counter of references to this attribute instance (used to enable garbage collection).	1

#### B. Implementing the attribution operations

Table 3 outlines, using pseudo-code, the implementation of the attribution operations. In this pseudo-code, references are intended to work like in Java, although we do not assume automatic garbage collection (instead, a `delete` primitive is explicitly invoked). Indeed, this is why we explicitly include `release` attribution operations.

The different operations behave as follows:

- `mkCtx` collects in a list many fresh attribute instances as needed.
- `mkDep` adds the second attribute instance in the `deps` list of the first one.
- `inst` stores the semantic function code in the `semFun` field.
- `release`, when applied to a list of semantic attribute instances, releases each instance and de-allocates the list itself.
- On the other hand, when `release` is applied to an attribute instance, decreases in 1 its reference count. If this count becomes 0, the instances on which it depends are released; finally, the original instance itself is de-allocated.
- `set` sets the `value` field and records its availability.
- `val` recovers the value of an attribute instance as follows: (i) if the value is available, it returns such a value, (ii) otherwise, it calls the semantic function manager to compute such a value and sets and returns it.

Thus, the demand-driven evaluation process arises from the interplay of the `val` attribution operation and the semantic function manager. Notice that, in our minimalistic conceptualization, we assume this manager has the pre-estab-

lished `exec` name, and its implementation is changed from encoding to encoding<sup>2</sup>.

TABLE III.

IMPLEMENTATION OF THE ATTRIBUTION OPERATIONS FOR ALLOWING A DEMAND-DRIVEN EVALUATION STYLE.

Operation	Implementation
<code>mkCtx(n)</code>	<code>as := new list</code> <code>for i := 0 to n-1 do</code> <code>add(as, new attribute)</code> <code>end for</code> <code>return as</code>
<code>mkDep(a<sub>0</sub>, a<sub>1</sub>)</code>	<code>add(a<sub>0</sub>.deps, a<sub>1</sub>)</code> <code>a<sub>1</sub>.refcount := a<sub>1</sub>.refcount + 1</code>
<code>inst(a,f)</code>	<code>a.semFun := f</code>
<code>release(as)</code>	<code>foreach a in as do</code> <code>release(a)</code> <code>end foreach</code> <code>delete as</code>
<code>release(a)</code>	<code>a.refcount := a.refcount - 1</code> <code>if a.refcount = 0 then</code> <code>foreach a' in a.deps do</code> <code>release(a')</code> <code>end foreach</code> <code>delete a.deps</code> <code>end if</code> <code>delete a</code>
<code>set(a,val)</code>	<code>a.value := val</code> <code>a.available := true</code>
<code>val(a)</code>	<code>if ¬ a.available then</code> <code>set(a, exec(a.semFun, a.deps))</code> <code>release(a.deps)</code> <code>end if</code> <code>return a.value</code>

Also notice how explicit garbage collection can be readily interleaved in the implementation of the attribution operation by appropriately managing the reference counters and by deallocating lists and records as soon as they become unreachable. Although in this evaluation style, most of the dependency graph remains in memory until parsing is finished, automatic garbage collection makes it possible to de-allocate useless parts of the graph when they becomes unreachable. It can be due to attribute instances that are not finally required in any computation (e.g., `F.env` in a `F ::= id` context), or to successive evolutions of the implementation combining pure attribute grammar features with implementation-oriented optimizations (e.g., global variables, on-the-fly evaluation of semantic attributes, ...).

### C. Example

In order to illustrate the internal functioning of the framework, we will use the example developed in subsection II.D, together with the input sentence `5 * (6 + x)`.

<sup>2</sup>Again it is possible to achieve more elegant solutions by using a programming language with a minimal higher-order support (e.g., a conventional object-oriented language). Nevertheless, our conceptualization keeps the essence of this evaluation approach.

Action	Input	Parse Stack
1. init	<code>*(6+x)\$</code>	
2. shift	<code>(6+x)\$</code>	<code>n<sup>[1]</sup></code>
3. reduce <code>F ::= n</code>	<code>(6+x)\$</code>	<code>F<sup>[2,3]</sup></code>
4. reduce <code>T ::= F</code>	<code>(6+x)\$</code>	<code>T<sup>[4,5]</sup></code>
5. shift	<code>6+x)\$</code>	<code>T<sup>[4,5]</sup>*</code>
6. shift	<code>+x)\$</code>	<code>T<sup>[4,5]</sup>*(</code>
7. shift	<code>x)\$</code>	<code>T<sup>[4,5]</sup>*(n<sup>[6]</sup></code>
8. reduce <code>F ::= n</code>	<code>x)\$</code>	<code>T<sup>[4,5]</sup>*(F<sup>[7,8]</sup></code>
9. reduce <code>T ::= F</code>	<code>x)\$</code>	<code>T<sup>[4,5]</sup>*(T<sup>[9,10]</sup></code>
10. reduce <code>E ::= T</code>	<code>x)\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[11,12]</sup></code>
11. shift	<code>)\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[11,12]</sup> +</code>
12. shift	<code>\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[11,12]</sup>+id<sup>[13]</sup></code>
13. reduce <code>F ::= id</code>	<code>\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[11,12]</sup>+F<sup>[14,15]</sup></code>
14. reduce <code>T ::= F</code>	<code>\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[11,12]</sup>+T<sup>[16,17]</sup></code>
15. reduce <code>E ::= E+T</code>	<code>\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[18,19]</sup></code>
16. shift	<code>\$</code>	<code>T<sup>[4,5]</sup>*(E<sup>[18,19]</sup>)</code>
17. reduce <code>F := (E)</code>	<code>\$</code>	<code>T<sup>[4,5]</sup>*(F<sup>[20,21]</sup></code>
18. reduce <code>T := T*F</code>	<code>\$</code>	<code>T<sup>[22,23]</sup></code>
19. reduce <code>E := T</code>	<code>\$</code>	<code>E<sup>[24,25]</sup></code>
20. reduce <code>S := E</code>	<code>\$</code>	<code>S</code>

Fig 4. Evolution of the translator generated from Figure 2 while analyzing `5 * (6 + x)`

Fig. 4 illustrates the evolution of the parser. Each symbol in the parse stack is superscripted by the list of the references to their attribute instances. Fig. 5 outlines the dependency structure created in the heap. In this structure, nodes correspond to attribute instances, while dependencies are indicated by mean of arrows. Each instance is accompanied by a numeric identifier (it is also used to indicate references in the parse stack), and by its *duration* (i.e., the parse action in which the instance was created, followed by the parse action in which it was deleted). For example, the instance 16 was created in the action 14 (reduction of the `T ::= F` rule), and it was destroyed in action 20 (once the parsing concluded and semantic evaluation was activated as a consequence of consulting `val` in the parse tree root).

Finally, it is important to remark several points. On one hand, it should be notice the parse tree is never explicitly built, since the process only requires the underlying dependency graph. Additionally, dependencies in Fig. 5 are reverted with respect to the usual convention, according to which, when `a` is used to compute `b`, an arc starting in `a` and finishing in `b` is used [8]. Indeed, dependencies actually represent the contents of the `deps` field. Additionally, the fact they appear reversed with respect to the usual convention em-

phasized the demand-driven nature of this evaluation strategy. Also, while most of the instances exists until the end of the process, this example makes apparent how those instances that become unreachable are readily de-allocated. For example, consider action 4: when rule  $T ::= F$  is reduced, the instance for  $F.env$  (i.e., the instance 3) is no longer needed, and therefore it can be de-allocated. Actually, this instance is de-allocated as consequence of releasing the  $F$  attribute instance list. By last, notice how lexical attributes for terminal symbols are created in the *shift* action that precedes the actual shift of such a symbol, or during parse initialization, in the case of the first shift. It is due to lexical attributes are actually created by the scanner, and we suppose 1-lookahead parsers, as those generated by LALR(1) parser generators.

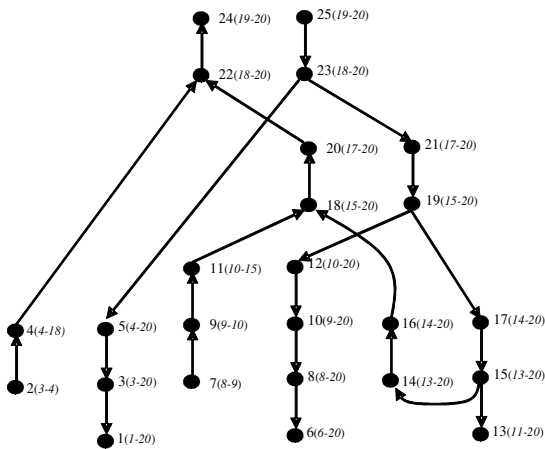


Fig 5. Dependency structure created in the heap as by the process outlined in Fig. 4.

#### IV. INCORPORATING A DATA-DRIVEN EVALUATION FRAMEWORK

In this section we describe an alternative implementation of the attribution operations, which leads to a *data-driven* evaluation style (see, for instance, [7]). In this evaluation style, attribute instances are scheduled for being evaluated as soon as the values for all the instances on which it depends are available. Thus, this method can shorten the durations of attribute instances. Additionally, it can interleave evaluation with parsing. These features can result of interest to process very long sentences, or sentences made available asynchronously (e.g., on a network communication channel). However, this method can do useless evaluations on attribute instances not required to yield the final results.

As in the previous section, we outline the representation of attribute instances (subsection IV.A), the implementation of attribute operations (subsection IV.B), and we illustrate how the method works with an example (subsection IV.C).

##### A. Representing the instances of the semantic attributes

Table 4 outlines the representation of attribute instances in the data-driven style. Notice that, in addition to the list of in-

stances on which an instance depends, it is needed to maintain the reverse relationship (i.e., each attribute instance must refer to those instances which depend on it). Indeed, this representation is similar to the used by networks of *observables-observers* in the *observer* object-oriented pattern [4]<sup>3</sup>.

TABLE IV.

STRUCTURE OF ATTRIBUTE INSTANCES IN THE DATA-DRIVEN EVALUATION FRAMEWORK.

Field	Purpose	Initial value
value	It keeps the value of the instance of the semantic attribute.	$\perp$
available	A boolean flag that indicates whether the value is available.	false
deps	It keeps the links to those attribute instances required to compute the value.	The empty list
obs	It keeps the links to those attribute instances observing it (i.e., which depend on it to compute their values).	The empty list
required	Counter which records the number of attribute instances in <i>deps</i> whose values have not yet been determined.	0
semFun	It stores the integer code of the semantic function required to compute the value.	$\perp$
instrumented	<i>True</i> if <i>semFun</i> was set, <i>false</i> otherwise.	false
refcount	A counter of references to this attribute instance (used to enable garbage collection).	1

##### B. Implementing the attribution operations

Table 5 outlines the pseudo-code of the attribution operations whose implementation differs from those in the demand-driven style. In this way, we only need to redefine *mkDep*, *inst*, *set* and *val*:

- In addition to updating *deps* in the first instance, *mkDep* must test whether the second instance was already computed. If it is not available, the first instance must be added to its *obs* list, since such an instance depends on its value, a value which is not yet available.
- On its hand, *inst* must take care of whether the value can be computed. Indeed, if the corresponding attribute instance has all the instances on which it depends computed, it can thereby be computed. It assumes the establishment of all the required dependencies before instrumentation, which is ensured by our encoding pattern.
- *Set* must take care of decrementing the *required* counters in all the instances depending

<sup>3</sup>As with the demand-driven style, this representation could be simplified, inferring the values of flags (in this case, *available* and *instrumented*) from the other fields. However, we prefer to explicitly preserve these flags to increase the readability of pseudo-code.

of the current one. In addition, if a counter becomes 0, it must enforce the evaluation of the corresponding instance.

- Finally, `val` immediately computes the value, unless the instance has not been yet instrumented.

TABLE V.

IMPLEMENTATION OF THE ATTRIBUTION OPERATIONS FOR ALLOWING A DATA-DRIVEN EVALUATION STYLE (ONLY THOSE IMPLEMENTATIONS DIFFERING FROM TABLE III ARE PRESENTED).

Operation	Implementation
<code>mkDep(<math>a_0, a_1</math>)</code>	<pre> <b>add</b> (<math>a_0</math>.deps, <math>a_1</math>) <math>a_1</math>.refcount := <math>a_1</math>.refcount + 1 <b>if</b> <math>\neg a_1</math>.available <b>then</b>   <b>add</b> (<math>a_1</math>.obs, <math>a_0</math>)   <math>a_0</math>.required := <math>a_0</math>.required + 1   <math>a_0</math>.refcount := <math>a_0</math>.refcount + 1 <b>end if</b> </pre>
<code>inst(<math>a, f</math>)</code>	<pre> <math>a</math>.semFun := <math>f</math> <math>a</math>.instrumented := <b>true</b> <b>if</b> <math>a</math>.required = 0 <b>then</b>   <math>val(a)</math> <b>end if</b> </pre>
<code>set(<math>a, val</math>)</code>	<pre> <math>a</math>.value := <math>val</math> <math>a</math>.available := <b>true</b> <b>foreach</b> <math>a'</math> <b>in</b> <math>a</math>.obs <b>do</b>   <math>a'</math>.required := <math>a'</math>.required - 1   <b>if</b> <math>a'</math>.required = 0 <b>then</b>     <math>val(a')</math>   <b>end if</b> <b>end foreach</b> release(<math>a</math>.obs) </pre>
<code>val(<math>a</math>)</code>	<pre> <b>if</b> <math>a</math>.instrumented <b>then</b>   set(<math>a</math>, exec(<math>a</math>.semFun, <math>a</math>.deps))   <math>a</math>.available := <b>true</b>   release(<math>a</math>.deps) <b>end if</b> </pre>

Notice how, in this case, evaluation can be interleaved with parsing. Indeed, evaluation is fired when the values of attribute instances are explicitly set, and also when attributes are instrumented. As a consequence, garbage collection also interplays with parsing, and, therefore, this method can incur in less heap overhead. It can be realized by considering the implementation of an *s-attributed* grammar (i.e., a grammar with only synthesized attributes) [1]. In this case, LHS attribute instances are computed when they are instrumented, and RHS attribute instances are garbage collected immediately before the reduction of the corresponding rules. On other cases, the behavior strongly depends on the nature of inherited attributes. In the extreme case (e.g., the example developed in subsection II.D), the dependency structure will be fully constructed, and evaluation will be delayed until the end of parsing, as in the demand-driven style. Still in these cases, it is possible to apply some straightforward optimizations on the resulting encoding, based on the use of *marker* non-terminals [1], in order to improve performance.

```

def env=0; def val=0;
def IDEN=0; def ADD=1; def MUL=2;
def TONUM=3; def VALOF=4;
S ::= M0 E {
  print (val ($2[val]));
  release ($1); release ($2); }
M0 ::=  $\lambda$  {
  $$ := mkCtx (1);
  set ($$[env], getEnv ()); }
E ::= E + M1 T {
  $$ := mkCtx (1);
  mkDep ($$[val], $1[val]); mkDep ($$[val], $4[val]);
  inst ($$[val], ADD);
  release ($1); release ($3); release ($4); }
M1 ::=  $\lambda$  {
  $$ := mkCtx (1);
  mkDep ($$[env], $-2[env]);
  inst ($$[env], IDEN) }
E ::= T {
  $$ := mkCtx (1);
  mkDep ($$[val], $1[val]);
  inst ($$[val], IDEN);
  release ($1); }
T ::= T * M1 F {
  $$ := mkCtx (1);
  mkDep ($$[val], $1[val]); mkDep ($$[val], $4[val]);
  inst ($$[val], MUL);
  release ($1); release ($3); release ($4); }
T ::= F {
  $$ := mkCtx (1);
  mkDep ($$[val], $1[val]);
  inst ($$[val], IDEN);
  release ($1); }
F ::= n {
  $$ := mkCtx (1);
  mkDep ($$[val], $1[lex]);
  inst ($$[val], TONUM);
  release ($1); }
F ::= id {
  $$ := mkCtx (1);
  mkDep ($$[val], $0[env]); mkDep ($$[val], $1[lex]);
  inst ($$[val], VALOF);
  release ($1); }
F ::= ( M2 E ) {
  $$ := mkCtx (1);
  mkDep ($$[val], $3[val]);
  inst ($$[val], IDEN);
  release ($2); release ($3); }
M2 ::=  $\lambda$  {
  $$ := mkCtx (1);
  mkDep ($$[env], $-1[env]);
  inst ($$[env], IDEN) }

```

Fig 6. Result of optimizing the translation scheme of Figure 2 with the help of markers to get the most of the data-driven evaluation style.

### C. Example

In order to illustrate the potential advantages of the data-driven method with respect to heap requirements, we will slightly modify the encoding of Fig. 2 by introducing marker non-terminals (i.e., new non-terminal symbols defined by empty rules) in strategic places<sup>4</sup>.

These marker non-terminals will allocate references to the `env` attribute instance for their immediate successors in the parse stack. It lets us discard equations to propagate the environment along the parse tree left-spines. The resulting encoding is shown in Fig. 6.

Fig. 7 illustrate the evolution of the parser while analyzing the sentence  $5 * (6 + x)$ . Fig. 8 shows the dependency structure created in the heap. Notice how the marker-based optimization performed on the encoding makes it possible to get a behavior equivalent to a *one-pass, on-the-fly*, evaluation of the semantic attributes, as the durations indicated in Fig. 8 make apparent.

<sup>4</sup>It must be carefully done, as the resulting context-free grammar can lose its desired character -e.g., LALR(1).

Action	Input	Parse Stack
1. init	*(6+x)\$	
2. reduce $M0 ::= \lambda$	*(6+x)\$	$M0^{[2]}$
3. shift	(6+x)\$	$M0^{[2]}n^{[1]}$
4. reduce $F ::= n$	(6+x)\$	$M0^{[2]}F^{[3]}$
5. reduce $T ::= F$	(6+x)\$	$M0^{[2]}T^{[4]}$
6. shift	6+x)\$	$M0^{[2]}T^{[4]}*$
7. reduce $M1 ::= \lambda$	6+x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}$
8. shift	+x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}($
9. reduce $M2 ::= \lambda$	+x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}$
10. shift	x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}n^{[6]}$
11. reduce $F ::= n$	x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}F^{[8]}$
12. reduce $T ::= F$	x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}T^{[9]}$
13. reduce $E ::= T$	x)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[10]}$
14. shift	)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[10]}+$
15. reduce $M1 ::= \lambda$	)\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[10]}+M1^{[12]}$
16. shift	\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[10]}+M1^{[12]}id^{[11]}$
17. reduce $F ::= id$	\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[10]}+M1^{[12]}F^{[13]}$
18. reduce $T ::= F$	\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[10]}+M1^{[12]}T^{[14]}$
19. reduce $E ::= E+MIT$	\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[15]}$
20. shift	\$	$M0^{[2]}T^{[4]}*M1^{[5]}(M2^{[7]}E^{[15]})$
21. reduce $F ::= (M2E)$	\$	$M0^{[2]}T^{[4]}*M1^{[5]}F^{[16]}$
22. reduce $T ::= T*MIF$	\$	$M0^{[2]}T^{[17]}$
23. reduce $E ::= T$	\$	$M0^{[2]}E^{[18]}$
24. reduce $S ::= M0 E$	\$	S

Fig 7. Evolution of the translator generated from Figure 6 while analyzing  $5 * (6 + x)$

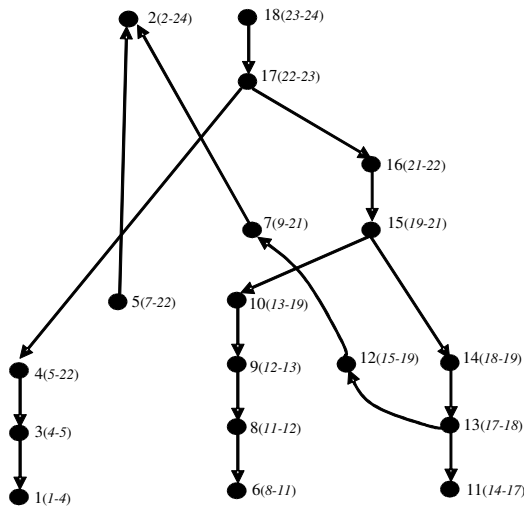


Fig 8. Dependency structure created in the heap as by the process outlined in Fig. 7

## V. CONCLUSIONS AND FUTURE WORK

This paper has shown how to systematically encode arbitrary non-circular attribute grammars in the input languages

of bottom-up, LALR(1) parse generation tools like YACC, BISON or CUP. It is done using a small set of attribution operations. These operations, in their turn, can be implemented of different ways in order to enable different semantic evaluation styles. In particular, this paper has illustrated two alternative implementations: one supporting a demand-driven style, and another one supporting a data-driven one. The results of this work can be useful to promote a systematic method of using conventional bottom-up parse generation tools to yield final implementations. This method starts with the initial encoding of an attribute grammar-based specification, and then it evolves it in a final implementation by applying systematic implementation patterns and techniques. Besides, the method facilitates the incremental introduction of new language features, since they can be described according to attribute grammar conventions, then readily encoded in the implementation, and finally optimized according to implementation-dependent criteria. Therefore, the method transports the attribute grammar amenability for doing modular and extensible specifications incrementally to an implementation process based on parse generation tools.

Currently we have successfully tested our method with several small examples, and we are applying it to the development of a non-trivial translator for a Pascal-like language. As future work, we plan to apply the method to descent parser generation tools (e.g., JavaCC or ANTLR).

## ACKNOWLEDGMENT

Thanks are due to the project grants TIN2010-21288-C02-01.

## REFERENCES

- [1] Aho A.V, Lam M.S, Sethi R, Ullman J.D. 2006. Compilers: principles, techniques and tools (2<sup>nd</sup> Edition). Addison-Wesley.
- [2] Akker, R; Melichar, B.; Tarhio, J. The Hierarchy of LR-attributed grammars. WAGA'90, Paris, France, September 19-21. 1990
- [3] Ekman, T., Hedin, G. The JastAdd system - modular extensible compiler construction. Sc. of Comp. Prog. 69(1-3), 14-26. 2007
- [4] Gamma,E; Helm,R; Jhonson,R; Vlissides,J. Design Patterns. Elements of Reusable Object-Oriented Software. Addison-Wesley. 1995
- [5] Jalili, F. A general linear-time evaluator for attribute grammars. ACM SIGPLAN Notices 18(9), 35-44. 1983
- [6] Katwijk, J. A preprocessor for YACC or a poor man's approach to parsing attributed grammar. ACM SIGPLAN Notices 18(10), 12-15. 1983
- [7] Kennedy, K.; Ramanathan, J. A Deterministic Attribute Grammar Evaluator Based on Dynamic Sequencing. ACM Transaction of Programming Languages and Systems 1(1), 142-160. 1979
- [8] Knuth, D. E. Semantics of Context-free Languages. Math. System Theory 2(2), 127-145. 1968. See also the correction published in Math. System Theory 5, 1, 95-96
- [9] Magnusson, E.; Hedin, G. Circular reference attributed grammars—their evaluation and applications. Sc. of Comp. Prog. 68(1), 21-37. 2007
- [10] Memik, M.,Lenic, M., Acdicausevic, E., Zumer, V. LISA: An Interactive Environment for Programming Language Development. CC 2002, Grenoble, France, April 8-12, 2002
- [11] Paaki, J. Attribute Grammar Paradigms – A High-Level Methodology in Language Implementation. ACM Comp. Surveys, 27, 2, 196-255. 1995

# The embedded left LR parser

Boštjan Slivnik

Faculty of Computer and Information Science

University of Ljubljana

Ljubljana, Slovenia

Email: bostjan.slivnik@fri.uni-lj.si

**Abstract**—A parser called the embedded left LR( $k$ ) parser is defined. It is capable of (a) producing the prefix of the left parse of the input string and (b) stopping not on the end-of-file marker but on any string from the set of lookahead strings fixed at the parser generation time. It is aimed at automatic construction of LL( $k$ ) parsers that use embedded LR( $k$ ) parsers to resolve LL( $k$ ) conflicts. The conditions regarding the termination of the embedded left LR( $k$ ) parser if used within LL( $k$ ) (and similar) parsers are defined and examined in-depth. As the embedded LR( $k$ ) parser produces the prefix of the left parse, the LL( $k$ ) parser augmented with embedded LR( $k$ ) parsers still produces the left parse and the compiler writer does not need to bother with different parsing strategies during the compiler implementation.

## I. INTRODUCTION

Parsing is an important phase of virtually any modern compiler because it represents the backbone upon which syntax-directed translation of the source program to the (intermediate) code is based. Furthermore, syntax errors in the source program can be successfully detected and precisely reported only if the appropriate parsing method is chosen.

The two most widely used parsing methods nowadays, i.e., LL and LR parsing [1], [2], are both relatively old [3], [4]. Nevertheless, the discourse on whether LL or LR parsing is more suitable either in general or in some particular case still goes on decades later after both methods have been simplified or strengthened many times since their discovery.

By careful examination of open source compilers for the most popular programming languages one can conclude only that the race between LL (most often implemented as a hand-written recursive-descent parser) and LR parsing remains open. For instance, Sun/Oracle Java compiler (a part of the standard JDK) employs a recursive-descent parser augmented with operator precedence parsers while Eclipse Java compiler uses Jikes-generated LALR parser. Likewise, the distribution of Google's Go includes two parsers, a recursive-descent one and a Bison-generated LALR one. GCC switched from Bison-generated LALR parser to the recursive-descent parser for parsing C++ in 2004 (gcc 3.4.0) and for C/ObjectiveC in 2006 (gcc 4.1.0). Python is parsed using a hand-written LL(1) parser (augmented with DFAs to select the next production at each step), but Ruby and PHP are parsed using Bison-generated LALR parser. Finally, Haskell is parsed using Happy-generated LALR parser (GHC and JHC) or recursive-descent parser (NHC). (No citation is given in this paragraph:

the findings can be best verified by downloading and examining the appropriate source code.)

The latest spark in this ongoing debate was triggered by the online publication of the paper entitled “Yacc is dead” [5]. Although the authors's original intent was to popularize a new parsing method, the online discourse quickly focused on whether it is better to use (mostly LALR) parser generators or write recursive-descent parsers by hand. As it might have been expected, no definite conclusion has been reached. However, two issues have been made clear (again). First, parser generators are appreciated, and second, both methods, LR and LL, remain attractive.

On one hand, LR parsing is popular for two reasons. First, unlike LL parsing, it is powerful: all deterministic context-free languages (DCFL) can be parsed using this method, and left-recursive productions (necessary for describing the left associativity of arithmetic operators, for instance) can be used. Second, for nearly every widespread programming language, an LALR parser generator is available (itself a consequence of the first reason).

On the other hand, the popularity of LL parsing stems from its simplicity which makes it suitable even for hand-written recursive-descent parsers, and its error recovery capability that allows generating precise error messages. Many LL parser generators are available, but quite a few include some way of producing parsers beyond the strength of LL(1) parsing: ANTLR employs LL(\*) parsing [6] while LISA offers both LL and LR parsing (but the generated parser uses either one method or the other, but never both) [7].

Apart from some major modifications of LL parsing like LL(\*) parsing [6], different techniques are used to bolster LL parsing. One way is to augment an LL(1) parser with DFAs (Python). Another way is to use small simple or operator precedence parsers [1] for parsing those phrases of the language (usually declarations or arithmetic expressions) that are too complicated for LL(1) parsing. However, none of these ways make parsing of all DCFL possible.

In this paper, we present yet another way to make LL parsing stronger: to use small LR parsers to resolve LL conflicts. Instead of the standard LR parser a modified LR parser which (a) produces the left parse and (b) stops as soon as the shortest prefix of the left parse can be computed, are to be used within the main LL parser. From the compiler writer's point of view the combined parser acts like a top-down parser capable of good error recovery [1], [10] while it

is as powerful as an LR parser since it can be constructed for any LR grammar.

An intermediate knowledge of LL and LR parsing is presumed. The notation used in [1] and [2] is adopted and all nonstandard symbols are introduced along the way. Furthermore, it is assumed that the result the parser produces is the *left (right) parse* of the input string, i.e., the list of productions needed to derive the input string from the initial grammar symbol using the leftmost (rightmost) derivation.

The paper is organized as follows. In Section II, the basic method for embedding LR( $k$ ) parsers into LL parsing is described and the embedded left LR( $k$ ) parser is formally defined in Sections III. The termination properties are investigated in Section IV. The paper ends with Conclusion containing a list of issues not cover in this paper due to the lack of space.

## II. EMBEDDING THE LR( $k$ ) PARSER INTO THE LL( $k$ ) PARSER

Consider that an LL( $k$ ) parser is being used for parsing a language generated by an LR( $k$ ) grammar, and that small LR( $k$ ) parsers are used to resolve LL( $k$ ) conflicts in the LL( $k$ ) parser. More precisely, let the backbone parser be an SLL( $k$ ), i.e., strong LL( $k$ ), parser. There are several reasons for using SLL( $k$ ) parser instead of the canonical LL( $k$ ) parser [1], [2]. First, the construction and implementation of the SLL( $k$ ) parser are much simpler and memory efficient than that of the canonical LL( $k$ ) parser. Second, every LL( $k$ ) grammar can be transformed into an equivalent SLL( $k$ ) grammar automatically, so no expressive power is lost. And finally, when  $k = 1$ , the only value of  $k$  used in practice, SLL(1) = LL(1).

Theoretically, the SLL( $k$ ) parser is a produce-shift parser and produces the left parse of the input string. For instance, after reading the prefix  $u$  of the input string  $w = uv$  that is derived by the derivation

$$S \Rightarrow_{G, \text{lm}}^{\pi_u} u\delta \Rightarrow_{G, \text{lm}}^{\pi_v} uv = w \quad ,$$

the SLL( $k$ ) parser for  $G = \langle N, T, P, S \rangle$  reaches configuration  $\$ \delta^R \mathbf{1} v \$$  with viable suffix  $\$ \delta^R$  on the stack and lookahead string  $x = k: v \$$  in the lookahead buffer; the parser's output contains the left parse  $\pi_u \in P^*$ . Furthermore, the parser is said to be in *position*  $X \mathbf{1} x$  if symbol  $X$  is the topmost stack symbol ( $\$ \delta^R = \$ \delta^R X$ ).

By theory [1], [2], the SLL( $k$ ) parser for  $G$  (based on the  $\$$ -augmented grammar  $G'$ ) exhibits produce-produce conflict in *conflicting position*  $A \mathbf{1} x$  if and only if there exist productions  $A \rightarrow \alpha_1, A \rightarrow \alpha_2 \in P$  where  $\alpha_1 \neq \alpha_2$  so that

$$x \in ( \text{FIRST}_k^{G'}(\alpha_1 \text{FOLLOW}_k^{G'}(A)) \cap \text{FIRST}_k^{G'}(\alpha_2 \text{FOLLOW}_k^{G'}(A)) ) \quad .$$

As the conflicting position  $A \mathbf{1} x$  is the result of every production  $B \rightarrow \beta_1 A \beta_2$  where

$$x \in ( \text{FIRST}_k^{G'}(\alpha_1 \beta_2 \text{FOLLOW}_k^{G'}(B)) \cap \text{FIRST}_k^{G'}(\alpha_2 \beta_2 \text{FOLLOW}_k^{G'}(B)) ) \quad ,$$

the basic idea (borrowed from the combination of LL and simple/operator precedence parsing) is to replace the production  $B \rightarrow \beta_1 A \beta_2$  with production  $B \rightarrow \beta_1 \langle\langle A \rangle\rangle \beta_2$  where  $\langle\langle A \rangle\rangle \notin T$  is a marker that triggers an LR( $k$ ) parser for  $A$ .

However, the embedded parser for  $A$  cannot assume that the end-of-input marker (denoted  $\$$ ) is at the end of the substring being parsed, i.e., the substring derived from  $A$ . It must stop when a string  $x \in \text{FIRST}_k^{G'}(\beta_2 \text{FOLLOW}_k^{G'}(B))$  is in the lookahead buffer, and then handle the control back to the backbone SLL( $k$ ) parser. This is not always possible as the following two examples demonstrate.

*Example 1:* Consider the grammar  $G_{\text{ex1}}$  with productions

$$S \rightarrow bAab, A \rightarrow Aa \mid a \quad .$$

The position  $A \mathbf{1} a$  exhibits the conflict, but the LR(1) parser for a grammar with the new symbol  $A$  cannot stop in the right moment. Suppose that an input string starts with  $baa$  and that the LR(1) parser for  $A$  has been triggered in configuration  $\$baA \mathbf{1} a \dots \$$ . After the first  $a$  is shifted and reduced to  $A$ , the lookahead buffer contains the second  $a$ . As  $k = 1$ , the LR(1) parser cannot decide whether the particular  $a$  in the buffer is derived from  $S$  (if  $baab$  is being parsed) or from  $A$  (if  $baaab$  is being parsed, for instance). The solution is to parse not just  $A$  but  $Aa$  using LR(1) parser as  $b$  is never a part of the input for this embedded LR parser and can thus stop on  $b$ . ■

*Example 2:* Consider the grammar  $G_{\text{ex2}}$  with productions

$$S \rightarrow bBab \mid abBb, B \rightarrow A, A \rightarrow Ba \mid a \quad .$$

The conflicting position is again  $A \mathbf{1} a$ . If the input string starts with  $baa$ , the LR(1) parser for a new start symbol  $A$  cannot stop correctly for the same reason as in Example 1. But now  $A$  is the rightmost symbol in production  $A \rightarrow B$  and thus the solution from Example 1 cannot be used. Hence, position  $B \mathbf{1} a$  must be declared conflicting instead.

However, if the input string starts with  $abaa$ , then the LR(1) parser for  $A$  used for resolving the conflict in position  $A \mathbf{1} a$  can stop on  $b$  — stopping of the embedded parsers clearly depends on the wider context within which the conflicting position occurs. ■

To avoid the problem of the context that is exposed in Example 2, the grammar  $G = \langle N, T, P, S \rangle$  for which the SLL( $k$ ) parser using embedded LR( $k$ ) parsers is to be constructed, is transformed into grammar  $\bar{G} = \langle \bar{N}, T, \bar{P}, \bar{S} \rangle$  where each nonterminal occurs in exactly one FOLLOW-context. More precisely, the start symbol becomes  $\bar{S} = \langle S, \{\varepsilon\} \rangle$  and the set  $\bar{N}$  of nonterminals is defined as

$$\bar{N} = \{ \langle A, \mathcal{F}_A \rangle; S \Rightarrow_{\text{lm}}^* uA\delta \wedge \mathcal{F}_A = \text{FIRST}_k^G(\delta) \} \quad .$$

For any nonterminal  $\langle A, \mathcal{F}_A \rangle$  the new set  $\bar{P}$  of productions includes productions

$$\langle A, \mathcal{F}_A \rangle \rightarrow \bar{X}_1 \bar{X}_2 \dots \bar{X}_n$$

where, for any  $i = 1, 2, \dots, n$ ,

$$\bar{X}_i = \begin{cases} X_i & X_i \in T \\ \langle X_i, \text{FIRST}_k^G(X_{i+1} X_{i+2} \dots X_n \mathcal{F}_A) \rangle & X_i \in N \end{cases}$$



provided that  $A \rightarrow X_1 X_2 \dots X_n \in P$ . (This transformation does not introduce any new  $LL(k)$  conflicts; in fact, if  $k > 1$  and  $SLL(k)$  parser is used instead of  $LL(k)$  parser, it even reduces the number of  $LL(k)$  conflicts for some non- $SLL(k)$  grammars [2].)

To resolve the  $SLL(k)$  conflicts during  $SLL(k)$  parsing, every production

$$\langle B, \mathcal{F}_B \rangle \rightarrow \beta_1 \langle A, \mathcal{F}_A \rangle \beta_2 \in \bar{P}$$

must be replaced with

$$\langle B, \mathcal{F}_B \rangle \rightarrow \beta_1 \langle \langle A \beta'_2, \mathcal{F}_{A \beta'_2} \rangle \rangle \beta''_2$$

where  $\beta_2 = \beta'_2 \beta''_2$  and  $\mathcal{F}_{A \beta'_2} = \text{FIRST}_k^G(\beta''_2 \mathcal{F}_B)$ . The new symbol  $\langle \langle A \beta'_2, \mathcal{F}_{A \beta'_2} \rangle \rangle \notin \bar{N}$  acts as a trigger for starting the embedded  $LR(k)$  parser for substrings derived from  $A \beta'_2$  that can stop on strings in  $\mathcal{F}_{A \beta'_2}$ .

Furthermore, as the amount of  $LR$  parsing is to be minimal,  $\beta'_2$  should be as short as possible or even  $\varepsilon$  in the best case. If, on the other hand, not even  $\beta''_2 = \varepsilon$  suffices for the safe termination of the embedded  $LR(k)$  parser,  $\langle B, \mathcal{F}_B \rangle$  must be declared conflicting nonterminal.

*Example 3:* Using the transformation described just above, grammar  $G_{\text{ex1}}$  is transformed to grammar  $\bar{G}_{\text{ex1}}$  with a single production

$$\langle S, \{\varepsilon\} \rangle \rightarrow b \langle \langle Ab, \{b\} \rangle \rangle b \quad .$$

Likewise, grammar  $G_{\text{ex2}}$  is transformed to grammar  $\bar{G}_{\text{ex2}}$  with productions

$$\langle S, \{\varepsilon\} \rangle \rightarrow b \langle \langle Ba, \{b\} \rangle \rangle b \mid ba \langle \langle B, \{b\} \rangle \rangle b$$

despite the fact that symbol  $B$  is not part of any conflicting position. ■

Finally, if marker  $\langle \langle \beta, \mathcal{F} \rangle \rangle$  is given for grammar  $G = \langle N, T, P, S \rangle$ , an embedded  $LR(k)$  parser that stops (no later than) on any lookahead string  $x \in \mathcal{F}$ , is needed. The easiest way to achieve this is to build the  $LR(k)$  parser for the *embedded grammar*

$$\hat{G} = \langle \hat{N}, T, \hat{P}, S_1 \rangle$$

where  $\hat{N} = N \cup \{S_1, S_2\}$  for  $S_1, S_2 \notin N$  and

$$\hat{P} = P \cup \{S_1 \rightarrow S_2 x, S_2 \rightarrow \beta ; x \in \mathcal{F}\} \quad .$$

The trick is obvious: the *embedded*  $LR(k)$  parser for  $\hat{G}$  must accept its input no later than when the reduction on  $S_2 \rightarrow \beta$  is due. In that way, it never pushes any symbol of any string  $x \in \mathcal{F}$  onto the stack. If this cannot be done, the embedded  $LR(k)$  parser for  $\langle \langle \beta, \mathcal{F} \rangle \rangle$  cannot be used.

### III. THE EMBEDDED LEFT $LR(k)$ PARSER

As mentioned in the introduction, the left  $LR(k)$  parser that is embedded into the backbone  $LL(k)$  parser must fulfill two requirements. First, it must produce the prefix of the left parse instead of the right parse so that the compiler writer can concentrate on the implementation of attribute grammar as if the entire input is being parsed using  $LL(k)$  parser. Second, it must stop and handle the control to the backbone parser

as soon as possible, preferably after the first production of the left parse is produced. In that way, the amount of  $LR(k)$  parsing is minimized. Furthermore, the embedded left  $LR(k)$  parser must be able to accept its input and terminate without the end-of-input marker in the lookahead buffer since it parses only a substring of the entire input — but this issue has been resolved in the previous section.

To meet these two goals, the embedded left  $LR(k)$  parser is a modification of the *left*  $LR(k)$  parser. The left  $LR(k)$  parser produces the left parse of the input string during bottom-up parsing using two methods [8].

The first method, first introduced in the Schmeiser-Barnard  $LR(k)$  parser [9], augments each nonterminal pushed on the  $LR$  stack with the left parse of the substring derived from that nonterminal:

- If the parser performs the shift action, no production is pushed on the stack, i.e., the terminal pushed is augmented with the empty left parse  $\varepsilon$ .
- If the parser performs the reduce action, the left parses accumulated in states that are removed from the stack are concatenated, and prefixed by the production the reduction is made on. The resulting left parse is pushed on the stack together with the new nonterminal.

Note that using this method, the first production of the left parse is produced only at the very end of parsing.

In general, take an  $LR(k)$  grammar  $G = \langle N, T, P, S \rangle$  and the input string  $w = uv$  derived by the rightmost derivation

$$S \Rightarrow_{G, \text{rm}}^* \gamma v \Rightarrow_{G, \text{rm}}^* uv \quad . \quad (1)$$

After reading the prefix  $u$ , the canonical  $LR(k)$  parser for grammar  $G$  reaches the configuration

$$[\$][\$X_1][\$X_1 X_2] \dots [\$X_1 X_2 \dots X_n] \mid v \$ \quad (2)$$

where  $X_1 X_2 \dots X_n = \gamma$ ,  $[\$X_1 X_2 \dots X_n]$  is the current parser state and  $x = k : v \$$  is the contents of the lookahead buffer. Note that  $[\$X_1 X_2 \dots X_j]$ , for  $j = 0, 1, \dots, n$ , denotes the state of the canonical  $LR(k)$  machine  $M_G$  reachable from the state  $[\$]$  by string  $X_1 X_2 \dots X_j$  ( $M_G$  is based on the  $\$$ -augmented grammar  $G'$  obtained by adding the new start symbol  $S'$  with production  $S' \rightarrow \$S\$$  to  $G$ ).

On the other hand, the Schmeiser-Barnard  $LR(k)$  parser (which is based on the canonical  $LR(k)$  machine as well) reaches the configuration

$$\begin{aligned} & \langle \langle [\$]; \varepsilon \rangle \langle [\$X_1]; \pi(X_1) \rangle \langle [\$X_1 X_2]; \pi(X_2) \rangle \dots \\ & \dots \langle [\$X_1 X_2 \dots X_n]; \pi(X_n) \rangle \mid v \$ \end{aligned} \quad (3)$$

where  $\pi(X_j)$  denote the left parse of the substring derived from  $X_j$ , i.e.,  $X_1 X_2 \dots X_n \Rightarrow_{G, \text{lm}}^{\pi(X_1) \pi(X_2) \dots \pi(X_n)} u$ .

*Example 4:* Consider the embedded grammar  $G_{\text{ex4}}$  with productions

$$\begin{aligned} S_1 & \rightarrow S_2 c, S_2 \rightarrow A, \\ A & \rightarrow aa \mid aB \mid bBa \mid bBaa, B \rightarrow Bb \mid \varepsilon \quad . \end{aligned}$$

Parsing of the input string  $bbbaac$  using the Schmeiser-Barnard  $LR(1)$  parser is shown in Table I. ■

TABLE I  
PARSING THE STRING  $bbbaac \in L(G_{\text{ex4}})$   
USING THE SCHMEISER-BARNARD LR(1) PARSER.

STACK	INPUT
$\$ \langle [\$]; \varepsilon \rangle$	$bbbaac\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$b]; \varepsilon \rangle$	$bbaac\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$b]; \varepsilon \rangle \langle [\$bB]; \pi_1 = B \rightarrow \varepsilon \rangle$	$bbaac\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$b]; \varepsilon \rangle \langle [\$bB]; \pi_1 = B \rightarrow \varepsilon \rangle \langle [\$bBb]; \varepsilon \rangle$	$baac\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$b]; \varepsilon \rangle \langle [\$bB]; \pi_2 = B \rightarrow Bb \cdot \pi_1 \rangle \langle [\$bBb]; \varepsilon \rangle$	$aac\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$b]; \varepsilon \rangle \langle [\$bB]; \pi_3 = B \rightarrow Bb \cdot \pi_2 \rangle$	$aac\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$b]; \varepsilon \rangle \langle [\$bB]; \pi_3 = B \rightarrow Bb \cdot \pi_2 \rangle \langle [\$bBa]; \varepsilon \rangle$	$ac\$$
$\$ \dots \langle [\$bB]; \pi_3 = B \rightarrow Bb \cdot \pi_2 \rangle \langle [\$bBa]; \varepsilon \rangle \langle [\$bBaa]; \varepsilon \rangle$	$c\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$A]; \pi_4 = A \rightarrow bBaa \cdot \pi_3 \rangle$	$c\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$S_2]; \pi_5 = S_2 \rightarrow A \cdot \pi_4 \rangle$	$c\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$S_2]; \pi_6 = S_2 \rightarrow A \cdot \pi_5 \rangle \langle [\$S_2c]; \varepsilon \rangle$	$\$$
$\$ \langle [\$]; \varepsilon \rangle \langle [\$S_1]; \pi_7 = S_1 \rightarrow S_2c \cdot \pi_6 \rangle$	$\$$

where  $\pi_7 = S_1 \rightarrow S_2c \cdot S_2 \rightarrow A \cdot A \rightarrow bBaa \cdot B \rightarrow Bb \cdot B \rightarrow Bb \cdot B \rightarrow \varepsilon$

The second method, first introduced in the left LR( $k$ ) parser [8], enables the parser to compute the prefix of the left parse of the substring corresponding to the prefix of the input string read so far (although this is not possible in every parser configuration). In other words, if apart from derivation (1) the input string  $w = uv$  is derived by the leftmost derivation

$$S \xRightarrow{\pi(u)}_{G, \text{lm}} u\delta \xRightarrow{*}_{G, \text{lm}} uv \quad , \quad (4)$$

then the left LR( $k$ ) parser computes the left parse  $\pi(u)$  in configuration (3). As this part of the left LR( $k$ ) parser is modified, it deserves more attention.

By theory [2], configurations (2) and (3) imply that machine  $M_G$  contains at least one sequence of valid  $k$ -items

$$\begin{aligned} & [A_0 \rightarrow \bullet \alpha_0 A_1 \beta_0, x_0] \cdot \dots \cdot [A_0 \rightarrow \alpha_0 \bullet A_1 \beta_0, x_0] \cdot \\ & \cdot [A_1 \rightarrow \bullet \alpha_1 A_2 \beta_1, x_1] \cdot \dots \cdot [A_1 \rightarrow \alpha_1 \bullet A_2 \beta_1, x_1] \cdot \\ & \quad \vdots \\ & \cdot [A_\ell \rightarrow \bullet \alpha_\ell A_{\ell+1} \beta_\ell, x_\ell] \dots [A_\ell \rightarrow \alpha_\ell \bullet A_{\ell+1} \beta_\ell, x_\ell] \end{aligned} \quad (5)$$

where  $[A_0 \rightarrow \bullet \alpha_0 A_1 \beta_0, x_0] = [S' \rightarrow \bullet \$\$ \$, \varepsilon]$ ,  $\gamma = \alpha_0 \alpha_1 \dots \alpha_\ell$ ,  $k: v\$ \in \text{FIRST}_k^{G'}(A_{\ell+1} \beta_\ell x_\ell)$ , and  $A_{\ell+1} = \varepsilon$ ; the horizontal dots denote repetitive application of operation **passes** (or **GOTO**) while the vertical dots denote the application of **desc** (or **CLOSURE**).

Sequence (5) induces the (*induced*) *central derivation*

$$\begin{aligned} S' = A_0 & \xRightarrow{G} \alpha_0 A_1 \beta_0 \\ & \xRightarrow{G} \alpha_0 \alpha_1 A_2 \beta_1 \beta_0 \\ & \quad \vdots \\ & \xRightarrow{G} \alpha_0 \alpha_1 \dots \alpha_\ell A_{\ell+1} \beta_\ell \beta_{\ell-1} \dots \beta_0 \quad ; \end{aligned}$$

the name “central” becomes obvious if the corresponding derivation tree presented in Figure 1(a) is observed.

However, if the left parses  $\pi(\alpha_0), \pi(\alpha_1), \dots, \pi(\alpha_\ell)$ , where  $\alpha_j \xRightarrow{\pi(\alpha_j)}_{G', \text{lm}} u_j$  for  $j = 0, 1, \dots, \ell$ , are provided, then

sequence (5) induces the (*induced*) *leftmost derivation*

$$\begin{aligned} S' = A_0 & \xRightarrow{G, \text{lm}} \alpha_0 A_1 \beta_0 \xRightarrow{\pi(\alpha_0)}_{G, \text{lm}} u_0 A_1 \beta_0 \\ & \xRightarrow{G, \text{lm}} u_0 \alpha_1 A_2 \beta_1 \beta_0 \xRightarrow{\pi(\alpha_1)}_{G, \text{lm}} u_0 u_1 A_2 \beta_1 \beta_0 \\ & \quad \vdots \\ & \xRightarrow{G, \text{lm}} u_0 u_1 \dots u_{\ell-1} \alpha_\ell A_{\ell+1} \beta_\ell \beta_{\ell-1} \dots \beta_0 \\ & \xRightarrow{\pi(\alpha_\ell)}_{G, \text{lm}} u_0 u_1 \dots u_\ell A_{\ell+1} \beta_\ell \beta_{\ell-1} \dots \beta_0 \end{aligned}$$

where  $u = u_0 u_1 \dots u_\ell$  and  $k: v\$ \in \text{FIRST}_k^{G'}(\beta_\ell \beta_{\ell-1} \dots \beta_0 \$)$ . The corresponding derivation tree is shown in Figure 1(b) and the left parse of the induced leftmost derivation is therefore

$$\begin{aligned} \pi(u) = A_0 & \longrightarrow \alpha_0 A_1 \beta_0 \cdot \pi(\alpha_0) \cdot \\ & \cdot A_1 \longrightarrow \alpha_1 A_2 \beta_1 \cdot \pi(\alpha_1) \cdot \\ & \quad \vdots \\ & \cdot A_\ell \longrightarrow \alpha_\ell A_{\ell+1} \beta_\ell \cdot \pi(\alpha_\ell) \quad . \end{aligned} \quad (6)$$

(Likewise, if the right parses  $\pi(\beta_1), \pi(\beta_2), \dots, \pi(\beta_\ell)$  are known, then sequence (5) induces the (*induced*) *rightmost derivation* producing the derivation tree in Figure 1(c).)

Subparses  $\pi(\alpha_j)$  of the left parse (6) are available on the parser stack because  $\alpha_0 \alpha_1 \dots \alpha_\ell = \gamma = X_1 X_2 \dots X_n$ , but productions  $A_j \rightarrow \alpha_j A_{j+1} \beta_j$  are not. However, if sequence (5) is known, the missing productions and in fact the entire prefix of the left parse can be computed [8]. Starting with  $\pi = \varepsilon$  and  $i = [A_\ell \rightarrow \alpha_\ell \bullet A_{\ell+1} \beta_\ell, x_\ell]$ , the stack is traversed downwards:

- If  $i = [A \rightarrow \bullet \beta, x]$ , then (a)  $i$  expands the nonterminal  $A$  by production  $A \rightarrow \beta$  and (b)  $i'$ , the item that precedes  $i$  in sequence (5), is in the same state. Hence, let  $\pi := A \rightarrow \beta \cdot \pi$  and  $i' := i$ .
- If  $i = [A \rightarrow \alpha X \bullet \beta, x] \in [\$ \gamma X]$  for some  $\gamma$ , then (a) the left parse  $\pi(X)$  is available on the stack and (b)  $i'$  is in the state  $[\$ \gamma]$  (which is found beneath  $[\$ \gamma X]$ ). Hence, let  $\pi := \pi(X) \cdot \pi$  and  $i' := i$ ; furthermore, proceed one step downwards along the stack, i.e., to the state  $[\$ \gamma]$ .

The downward traversal stops when the item  $[S_2 \rightarrow \bullet \beta, x] \in [\$]$ , for some  $\beta \in (N \cup T)^*$  and  $x \in (T \cup \{\$\})^{*k}$ , is reached (the production  $S_2 \rightarrow \beta$  is not added to the resulting left parse).

This method can be upgraded to compute the prefix of the left parse and the viable suffix  $\delta^R$  in derivation (4) as well since  $\delta = A_{\ell+1} \beta_\ell \beta_{\ell-1} \dots \beta_0$  — see Figure 1(b). Hence, start with  $\delta = A_{\ell+1} \beta_\ell$  and whenever  $i = [A \rightarrow \bullet \beta, x]$ , let  $\delta := \delta \cdot \beta'$  where  $i' = [A' \rightarrow \alpha' \bullet A \beta', x']$  is the item preceding  $i$  in sequence (5).

*Example 5:* Consider again grammar  $G_{\text{ex4}}$  and the input string  $bbbaac \in L(G_{\text{ex4}})$  from Example 4. After the prefix  $bbba$  of the input string has been read, the parser reaches the configuration shown in the 7th line of Table I. But as illustrated in Figure 2, there is only one item active for the current lookahead string  $a$  in state  $[\$bBa]$ , namely  $[S_2 \rightarrow bBa \bullet a, \$]$ . Furthermore, there exist exactly one sequence of LR(1) items starting with  $[S' \rightarrow \bullet \$\$ \$, \varepsilon] \in [\varepsilon]$  and ending

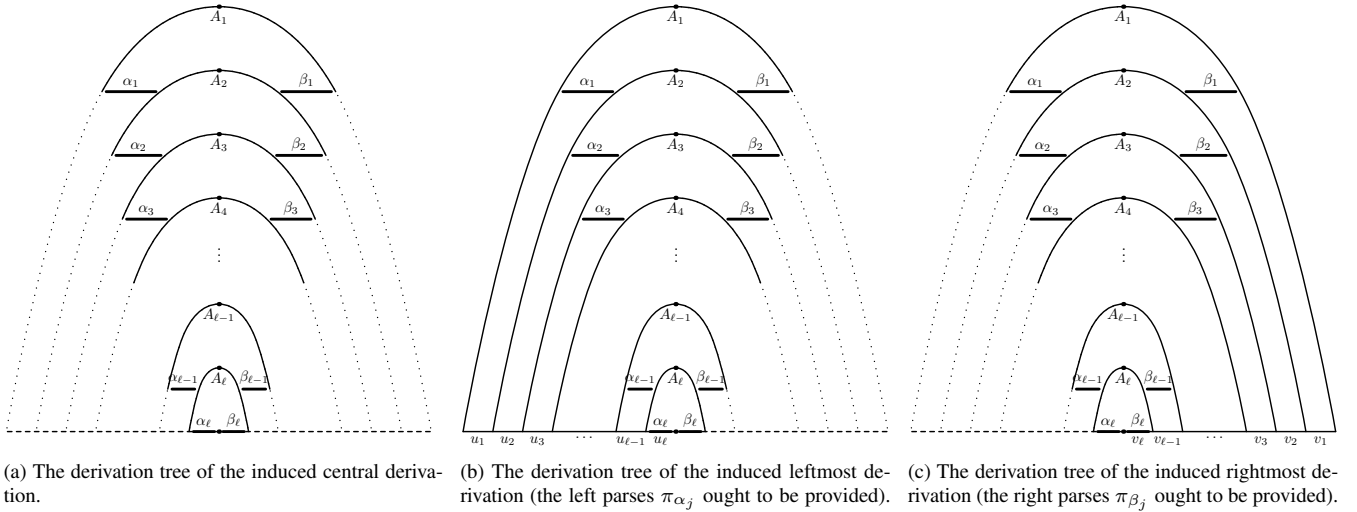


Fig. 1. The derivation trees corresponding to various kinds of induced derivations; remember that  $A_{\ell+1} = \varepsilon$  in all three cases.

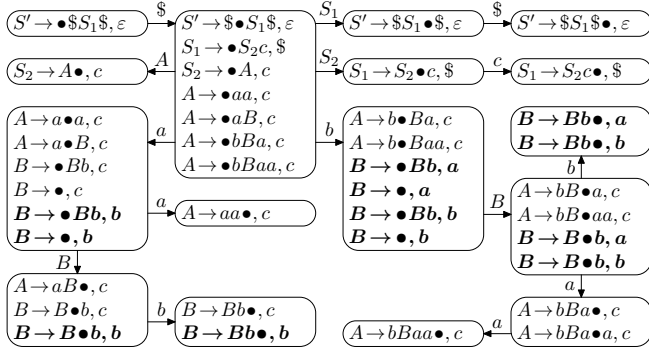


Fig. 2. The canonical LR(1) machine for  $G_{\text{ex4}}$  — items that end multiple sequences starting with  $[S' \rightarrow \bullet \$S$,  $\varepsilon] \in [\varepsilon]$  are shown in bold face.$

with  $[S_2 \rightarrow bBa\bullet a, \$] \in [\$bS_2a]$ :

$$\begin{aligned}
 & [S' \rightarrow \bullet \$S_1$,  $\varepsilon] \cdot [S' \rightarrow \$\bullet S_1$,  $\varepsilon] \cdot [S_1 \rightarrow \bullet S_2c, \$] \cdot \\
 & \cdot [S_2 \rightarrow \bullet bBaa, \$] \cdot [S_2 \rightarrow b\bullet Baa, \$] \cdot \\
 & \cdot [S_2 \rightarrow bB\bullet aa, \$] \cdot [S_2 \rightarrow bBa\bullet a, \$]
 \end{aligned}$$$$

Hence, the prefix of the left parse and the corresponding viable suffix can be computed by the method outlined above as shown in Figure 3. ■

In general, cases where exactly one sequence (5) exists (as in Example 5) are extremely rare, but all sequences (5) that differ only in lookahead strings  $x_j$ , where  $j = 1, 2, \dots, \ell$ , induce the same (leftmost) derivation. In other words, the lookahead strings  $x_j$  are not needed for computing the prefix of the left parse and the viable suffix.

The left LR( $k$ ) parser uses an additional parsing table called LEFT to establish whether the prefix of the left parse can be computed in some state  $[\$ \gamma]$  for some lookahead string  $x$ , and the *left-parse-prefix* automaton (LPP) to actually compute sequence (5) with the lookahead strings omitted.

The LEFT table implements mapping

$$\text{LEFT: } Q_k^G \times (T \cup \{\$\})^{*k} \longrightarrow (I_0^G \cup \{\perp\})$$

where  $Q_k^G$  and  $I_0^G$  denote the set of LR( $k$ ) states and the set of LR(0) items for grammar  $G'$ , respectively. It maps LR( $k$ ) state  $[\$ \gamma]$  and the contents  $x$  of the lookahead buffer to either

- $[A_\ell \rightarrow \alpha_\ell \bullet A_{\ell+1} \beta_\ell]$ , where  $\alpha_\ell \neq \varepsilon$ , if all sequences (5) that are active for  $x$ , i.e., they end with some LR( $k$ ) item  $[A_\ell \rightarrow \alpha_\ell \bullet A_{\ell+1} \beta_\ell, x_\ell]$  (for different  $x_\ell$ ) where  $x \in \text{FIRST}_k^{G'}(A_{\ell+1} \beta_\ell x_\ell)$ , differ in lookahead strings only, or
- $\perp$  otherwise.

Hence, the parser can produce the prefix of the left parse and compute the viable suffix if and only if  $\text{LEFT}([\$ \gamma], x) \neq \perp$ .

The above definition of LEFT works well for the left LR( $k$ ) parser [8]. But as (a)

$$[\$] = \text{desc}^*([\$ \bullet S' \rightarrow \$\bullet S_1$,  $\varepsilon]$ )$$

(note that the embedded grammar is being used) and (b) there is only one path to  $\{[S' \rightarrow \$\bullet S_1$,  $\varepsilon]\} \in [\$]$ , the value of  $\text{LEFT}([\$], x)$  is set to  $[S' \rightarrow \$\bullet S_1$]$  for all  $x \in \text{FIRST}_k^{G'}(S_1 \$)$  — if the definition suitable for the left LR( $k$ ) parser is used. It is valid but useless because if the method outlined in Example 5 is used, the embedded LR( $k$ ) parser would print  $\varepsilon$  and stop before ever producing any production of the left parse.$

Thus, an exception must be made in state  $[\$]$ . Provided that the grammar includes the productions  $S_1 \rightarrow S_2 y$  and  $S_2 \rightarrow A \beta$ , the value of  $\text{LEFT}([\$], x)$  must be set to either

- $[A_\ell \rightarrow \bullet A_{\ell+1} \beta_\ell]$  if all sequences (5) that are active for  $x$ , i.e., they end with some LR( $k$ ) item  $[A_\ell \rightarrow \bullet A_{\ell+1} \beta_\ell, x_\ell]$  (for different  $x_\ell$ ) where  $x \in \text{FIRST}_k^{G'}(A_{\ell+1} \beta_\ell x_\ell)$ , differ in lookahead strings only and

$$[S_2 \rightarrow \bullet A_\ell \beta, y] \text{ desc } [A_\ell \rightarrow \bullet A_{\ell+1} \beta_\ell, x_\ell] \quad ,$$

or

$\$ \langle [\$]; \varepsilon \rangle$	$\langle [\$b]; \varepsilon \rangle$	$\langle [\$bB]; B \rightarrow Bb, B \rightarrow Bb, B \rightarrow \varepsilon \rangle$	$\langle [\$bBa]; \varepsilon \rangle$	<b>l</b> $ac$ <b>\$</b>
$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$[A \rightarrow \bullet bBaa, c] \in [\$]$	$[A \rightarrow b \bullet Baa, c] \in [\$b]$	$[A \rightarrow bB \bullet aa, c] \in [\$bB]$	$[A \rightarrow bBa \bullet a, c] \in [\$bBa]$	
$\pi_4 = A \rightarrow bBaa \cdot \pi_3$	$\pi_3 = \varepsilon \cdot \pi_2$	$\pi_2 = B \rightarrow Bb \cdot B \rightarrow Bb \cdot B \rightarrow \varepsilon \cdot \pi_1$	$\pi_1 = \varepsilon \cdot \pi_0$ where $\pi_0 = \varepsilon$	
$\delta_4 = \delta_3 \cdot \varepsilon$	$\delta_3 = \delta_2$	$\delta_2 = \delta_1$	$\delta_1 = \delta_0$ where $\delta_0 = a$	
$[S_2 \rightarrow \bullet A, c] \in [\$]$				
$\pi_5 = (S_2 \rightarrow A) \cdot \pi_4$				
$\delta_5 = \delta_4$				
The result: $\pi_5 = A \rightarrow bBaa \cdot B \rightarrow Bb \cdot B \rightarrow Bb \cdot B \rightarrow \varepsilon$ ; $\delta_6 = a$				

Fig. 3. Computing the prefix of the left parse of the string  $bbbaac \in L(G_{ex4})$  after  $bbba$  has been read: the computation starts at the top of the stack (right side of the figure) with  $\pi_0 = \varepsilon$  and  $\delta_0 = a$ , and traverses the stack downwards (towards the left side of the figure, and then downwards).

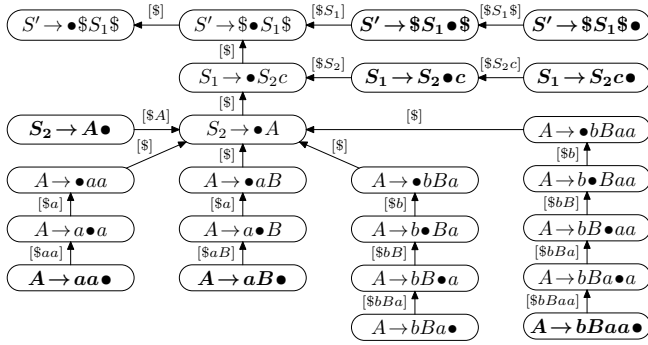


Fig. 4. The left-parse-prefix automaton for  $G_{ex4}$  — items that are not needed during embedded left LR(1) parsing are shown in bold face.

- $\perp$  otherwise.

The left-parse-prefix automaton represents mapping

$$\text{LPP: } I_0^G \times Q_k^G \longrightarrow I_0^G$$

which is a compact representation of all possible sequences (5) with lookahead strings stripped off. Hence,  $\text{LPP}(i_0, [\$ \gamma]) = i_0'$  if and only if there exists some sequence (5) with two consecutive LR( $k$ ) items  $i_k', i_k$ , where  $i_k \in [\$ \gamma]$ , so that  $i_0'$  is equal to  $i_k$  ( $i_k'$ ) without the lookahead string.

*Example 6:* The left-parse-prefix automaton for grammar  $G_{ex4}$  is shown in Figure 4. (In this example, the left-parse-prefix automaton is trivial, i.e., without any loop, but if the grammar is bigger and describes some more complex language, the corresponding LPP gets more complicated — see [8].)

Mapping LEFT for  $G_{ex4}$  is defined as

$$\begin{aligned} \text{LEFT}([\$S_2], c) &= [S_2 \rightarrow A \bullet c] \\ \text{LEFT}([\$a], a) &= [A \rightarrow a \bullet a] \\ \text{LEFT}([\$a], b) &= [A \rightarrow a \bullet B] \\ \text{LEFT}([\$bBa], \$) &= [A \rightarrow bBa \bullet] \\ \text{LEFT}([\$bBa], b) &= [A \rightarrow bBa \bullet a] \end{aligned}$$

(in all other cases, the value of LEFT equals  $\perp$ ). Note that  $\text{LEFT}([\$], a) = \perp$  and  $\text{LEFT}([\$], b) = \perp$  because of  $A \rightarrow aa|aB$  and  $A \rightarrow bBa|bBaa$ , respectively. ■

The algorithms for computing LEFT and LPP can be found in [8]. Once mappings LEFT and LPP are available, the method

for computing the prefix of the left parse and the viable suffix as outlined above and illustrated by Example 5 can be formalized as Algorithm 1. It is basically an algorithm which performs a *long reduction*: a sequence of reductions on productions whose right sides have been only partially pushed on the stack.

If compared with the similar method used by the left LR( $k$ ) parser [8], this one is not only augmented to compute the viable suffix but also simplified in that it does not leave any markers on the stack about which subparses accumulated on the stack have already been printed out. It does not need to do this as after the first long reduction the LR parsing stops, the LR stack is cleared, and the control is given back to the backbone LL( $k$ ) parser.

Finally — for the sake of completeness, the sketch of the embedded left LR( $k$ ) parser is given as Algorithm 2: in essence, it is a Schemiser-Barnard LR( $k$ ) parser [9] with the option of (a) premature termination and (b) computing the viable suffix.

Algorithm 2 always terminates: if not sooner (including cases where it detects a syntax error), the parser eventually reaches the (final) state  $[\$S_2] = \{[S_1 \rightarrow S_2 \bullet x, \$]\}$  where  $\text{LEFT}([\$S_2], \$) = [S_1 \rightarrow S_2 \bullet x]$  causing it to exit the loop in lines 3–5.

To conclude, the embedded left LR( $k$ ) parser is the left LR( $k$ ) parser for the embedded grammar (with a modified mapping LEFT) which (a) produces the left parse of the substring parsed and the remaining viable suffix and (b) terminates after the first (simplified) long reduction.

#### IV. THE TERMINATION OF THE EMBEDDED LEFT LR PARSER

Determining whether the embedded LR( $k$ ) parser does not contain any LR( $k$ ) conflicts is time consuming if a brute-force approach of using testing whether  $\hat{G} \in \text{LR}(k)$  is used. However, the method based on the following theorem significantly reduces the time complexity of testing the embedded LR( $k$ ) parser for LR( $k$ ) conflicts.

*Theorem 1:* Let  $G = \langle N, T, P, S \rangle$  be an LR( $k$ ) grammar with the derivation

$$S \Longrightarrow_{G, \text{lm}}^* uB\delta \Longrightarrow_{G, \text{lm}} u\beta_1\beta_2\delta \quad .$$

---

**Algorithm 1** *long-reduction*: computing the prefix of the left parse and the viable suffix.

---

*long-reduction*  $(\Gamma, [A \rightarrow \alpha \bullet \beta]) = \langle \pi, \beta \cdot \delta \rangle$   
 where  $\langle \pi, \delta \rangle = \text{long-reduction}'(\Gamma, [A \rightarrow \alpha \bullet \beta])$   
 $\text{long-reduction}'(\Gamma, [S' \rightarrow \$ \bullet S \$]) = \langle \varepsilon, \varepsilon \rangle$   
 $\text{long-reduction}'(\Gamma \cdot \langle [\$ \gamma X], \pi(X) \rangle, [A \rightarrow \bullet \beta]) = \langle A \rightarrow \beta \cdot \pi, \delta \cdot \beta' \rangle$   
 where  $[A' \rightarrow \alpha' \bullet A \beta'] = \text{LPP}([A \rightarrow \bullet \beta], [\$ \gamma X])$   
 $\langle \pi, \delta \rangle = \text{long-reduction}'(\Gamma \cdot \langle [\$ \gamma X], \pi(X) \rangle, [A' \rightarrow \alpha' \bullet A \beta'])$   
 $\text{long-reduction}'(\Gamma \cdot \langle [\$ \gamma X'], \pi(X') \rangle \cdot \langle [\$ \gamma X' X], \pi(X) \rangle, [A \rightarrow \alpha \bullet \beta]) = \langle \pi(X) \cdot \pi, \delta \rangle$   
 where  $\langle \pi, \delta \rangle = \text{long-reduction}'(\Gamma \cdot \langle [\$ \gamma X'], \pi(X') \rangle, \text{LPP}([A \rightarrow \alpha \bullet \beta], [\$ \gamma X]))$

---



---

**Algorithm 2** Embedded LR( $k$ ) parsing.

---

1: let  $q \in Q_k^G$  denote the topmost state  
 2: let  $x \in (T \cup \{\$\})^{*k}$  denote the LA buffer contents  
 3: **while**  $(i \leftarrow \text{LEFT}(q, x)) = \perp$  **do**  
 4:   perform a step of the Schmeiser-Barnard LR( $k$ ) parser  
 5: **end while**  
 6:  $\langle \pi, \delta \rangle \leftarrow \text{long-reduction}(\text{stack}, i)$   
 7: **PRINT**  $\pi$   
 8: **return**  $\delta$

---

Grammar  $\hat{G} = \langle \hat{N}, T, \hat{P}, S_1 \rangle$  where

$$\hat{N} = N \cup \{S_1, S_2\} \text{ for } S_1, S_2 \notin N \text{ and}$$

$$\hat{P} = P \cup \{S_1 \rightarrow S_2 x, S_2 \rightarrow \beta_2 ; x \in \text{FIRST}_k^G(\delta)\} \quad ,$$

is not an LR( $k$ ) grammar if and only if

$$[S_2 \rightarrow \bullet \beta_2, x'] \text{ desc}^* [B \rightarrow \bullet \beta_2, x']$$

where  $x' = k: x\$$  for some  $x \in \text{FIRST}_k^G(\delta)$ .

*Proof:* First, the structure of grammar  $\hat{G}$  implies that items  $[S_1 \rightarrow \bullet S_2 x, \$]$  and  $[S_2 \rightarrow \bullet \beta_2, x']$ , where  $x' = k: x\$$ , appear in the state  $[\$]_{\hat{G}} = \text{desc}^*([S' \rightarrow \$ \bullet S_1 \$, \varepsilon])$  only. Hence, any item  $[S_1 \rightarrow \psi_1 \bullet \psi_2, \$]$  or  $[S_2 \rightarrow \psi_1 \bullet \psi_2, x']$  can appear in state  $[\$ \psi_1]_{\hat{G}}$  only.

Second, because of the leftmost derivation above, there exists the rightmost derivation

$$S \Longrightarrow_{G, \text{rm}}^* \gamma B v \Longrightarrow_{G, \text{rm}} \gamma \beta_1 \beta_2 v$$

and thus

$$\{[B \rightarrow \beta_1 \bullet \beta_2, x] ; x \in \text{FIRST}_k^{G'}(\delta \$)\} \subseteq [\$ \gamma \beta_1]_G$$

where  $[\$ \gamma \beta_1]_G$  is a state of the canonical LR( $k$ ) machine for grammar  $G$ . Therefore,

- $[S_1 \rightarrow \bullet S_2 x, \$] \in [\$]_{\hat{G}}$  implies  $[B \rightarrow \beta_1 \bullet \beta_2, x'] \in [\$ \gamma \beta_1]_G$  where  $x' = k: x\$$ , and
- $[S_2 \rightarrow \hat{\gamma} \bullet \psi, x'] \in [\$ \hat{\gamma}]_{\hat{G}}$  implies  $[B \rightarrow \beta_1 \hat{\gamma} \bullet \psi, x'] \in [\$ \gamma \beta_1 \hat{\gamma}]_G$ .

Consider any two items  $i_1$  and  $i_2$  (except items based on the production  $S' \rightarrow \$ S_1 \$$  as these items are never involved in an LR( $k$ ) conflict) in any state  $[\$ \hat{\gamma}]_{\hat{G}}$  of the canonical LR( $k$ ) machine for  $\hat{G}$ , i.e.,  $i_1, i_2 \in [\$ \hat{\gamma}]_{\hat{G}}$ :

- If  $i_1$  and  $i_2$  are based on productions in  $P$ , then  $i_1, i_2 \in [\$ \gamma \beta_1 \hat{\gamma}]_G$  and there is no LR( $k$ ) conflict between  $i_1$  and  $i_2$  since  $G \in \text{LR}(k)$ .
- If  $i_1$  and  $i_2$  are based on productions in  $\hat{P} \setminus P$ , three cases must be considered:
  - If  $i_1 = [S_1 \rightarrow \hat{\gamma} \bullet \alpha, \$]$  and  $i_2 = [S_1 \rightarrow \hat{\gamma} \bullet \alpha', \$]$ , then either  $\hat{\gamma} = \varepsilon$  and both items imply the shift action since  $\alpha, \alpha' \neq \varepsilon$  or  $k: \alpha \$ \neq k: \alpha' \$$  so no conflict is possible.
  - If  $i_1 = [S_1 \rightarrow \hat{\gamma} \bullet \alpha, \$]$  and  $i_2 = [S_2 \rightarrow \hat{\gamma} \bullet \alpha', y']$  (or vice-versa), then  $\hat{\gamma} = \varepsilon$  (otherwise  $\hat{\gamma} = S_2 \hat{\gamma}'$  because of  $i_1$  but  $\hat{\gamma} \neq S_2 \hat{\gamma}'$  because of  $i_2$ ) and both items imply the shift action since  $\alpha, \alpha' \neq \varepsilon$ .
  - If  $i_1 = [S_2 \rightarrow \hat{\gamma} \bullet \alpha, y]$  and  $i_2 = [S_2 \rightarrow \hat{\gamma} \bullet \alpha', y']$ , then  $\alpha = \alpha'$  and both items imply either the reduce action on  $S_2 \rightarrow \beta_2$  or the shift action.
- If  $i_1$  is based on a production in  $\hat{P} \setminus P$  and  $i_2$  is based on a production in  $P$  (or vice versa), two cases must be considered:
  - If  $i_1 = [S_1 \rightarrow \hat{\gamma}_1 \hat{\gamma}_2 \bullet \alpha, \$]$  and  $i_2 = [A \rightarrow \hat{\gamma}_2 \bullet \alpha', y']$ , then obviously  $\hat{\gamma}_1 \hat{\gamma}_2 = \varepsilon$  and  $\alpha = S_2 x$  for some  $x \in \text{FIRST}_k^G(\delta)$  (otherwise  $\hat{\gamma}_1 \hat{\gamma}_2 = S_2 \hat{\gamma}'$  because of  $i_1$  but  $\hat{\gamma}_1 \hat{\gamma}_2 \neq S_2 \hat{\gamma}'$  because of  $i_2$ ). Thus

$$[S_1 \rightarrow \bullet S_1 x, \$], [A \rightarrow \bullet \alpha', y'] \in [\$]_{\hat{G}}$$

implies

$$[B \rightarrow \beta_1 \bullet \beta_2, x'], [A \rightarrow \bullet \alpha', y'] \in [\$ \gamma \beta_1]_G$$

where  $x' = k: x\$$ . But as

$$\text{FIRST}_k^{\hat{G}'}(S_2 x \$) = \text{FIRST}_k^{G'}(\beta_2 x')$$

and no items in  $[\$ \gamma \beta_1]_G$  exhibit any LR( $k$ ) conflict, the items  $[S_1 \rightarrow \bullet S_1 x, \$]$  and  $[A \rightarrow \bullet \alpha', y']$  do not exhibit LR( $k$ ) conflict either.

- If  $i_1 = [S_2 \rightarrow \hat{\gamma}_1 \hat{\gamma}_2 \bullet \alpha, x']$  and  $i_2 = [A \rightarrow \hat{\gamma}_2 \bullet \alpha', y']$  where  $x' = k: x\$$  for some  $x \in \text{FIRST}_k^{\hat{G}'}(\delta)$ , then  $\hat{\gamma}_1 \hat{\gamma}_2 = \beta_2$  because of  $i_1$  and  $[B \rightarrow \beta_1 \hat{\gamma}_1 \hat{\gamma}_2 \bullet \alpha, x'], [A \rightarrow \hat{\gamma}_2 \bullet \alpha', y'] \in [\$ \gamma \beta_1 \hat{\gamma}_1 \hat{\gamma}_2]_G$ . If  $\alpha \neq \varepsilon$  and  $\alpha' \neq \varepsilon$ , then items  $i_1$  and  $i_2$  both imply the shift action.  
 If  $\alpha \neq \varepsilon$  but  $\alpha' = \varepsilon$ , then  $y' \notin \text{FIRST}_k^{\hat{G}'}(\alpha x')$  as otherwise  $[B \rightarrow \beta_1 \hat{\gamma}_1 \hat{\gamma}_2 \bullet \alpha, x']$  and  $[A \rightarrow \hat{\gamma}_2 \bullet, y']$

would exhibit a shift-reduce conflict; hence, items  $i_1$  and  $i_2$  do not exhibit any LR( $k$ ) conflict.

If  $\alpha' \neq \varepsilon$  but  $\alpha = \varepsilon$ , then  $x' \notin \text{FIRST}_k^{\hat{G}'}(\alpha'y')$  as otherwise  $[B \rightarrow \beta_1\hat{\gamma}_1\hat{\gamma}_2\bullet, x']$  and  $[A \rightarrow \hat{\gamma}_2\bullet\alpha', y']$  would exhibit a reduce-shift conflict; hence, items  $i_1$  and  $i_2$  do not exhibit any LR( $k$ ) conflict.

If  $\alpha = \varepsilon$  and  $\alpha' = \varepsilon$ , then

$$[S_2 \rightarrow \beta_2\bullet, x'], [A \rightarrow \hat{\gamma}_2\bullet, y'] \in [\$ \beta_2]_{\hat{G}}$$

implies

$$[B \rightarrow \beta_1\beta_2\bullet, x'], [A \rightarrow \hat{\gamma}_2\bullet, y'] \in [\$ \gamma \beta_1 \beta_2]_G$$

Therefore, if and only if

$$[B \rightarrow \beta_1\beta_2\bullet, x'] = [A \rightarrow \hat{\gamma}_2\bullet, y']$$

where  $\hat{\gamma}_2 = \beta_1\beta_2$  (and thus  $\beta_1 = \varepsilon$ ) can there be a (reduce-reduce) conflict in  $[\$ \beta_2]_{\hat{G}}$  without a conflict in  $[\$ \gamma \beta_1 \beta_2]_G$ .

As determined, the only possibility for an LR( $k$ ) conflict in the canonical LR( $k$ ) machine for  $\hat{G}$  is the reduce-reduce conflict exhibited by items

$$[S_2 \rightarrow \beta\bullet, x'], [B \rightarrow \beta_2\bullet, x'] \in [\$ \beta_2]_{\hat{G}}$$

which are derived from items

$$[S_2 \rightarrow \bullet\beta, x'], [B \rightarrow \bullet\beta_2, x'] \in [\$]_{\hat{G}} .$$

But as

$$\begin{aligned} [\$]_{\hat{G}} &= \text{desc}^*\{[S' \rightarrow \$\bullet S_1 \$, \varepsilon]\} \\ &= \{[S' \rightarrow \$\bullet S_1 \$, \varepsilon]\} \\ &\quad \cup \{[S_1 \rightarrow \bullet S_2 x, \$] ; x \in \text{FIRST}_k^G(\delta)\} \\ &\quad \cup \{[S_2 \rightarrow \bullet\beta_2, x'] ; x' \in \text{FIRST}_k^{G'}(\delta \$)\} \\ &\quad \cup \text{desc}^*\{[S_2 \rightarrow \bullet\beta_2, x'] ; x' \in \text{FIRST}_k^{G'}(\delta \$)\} , \end{aligned}$$

item  $[B \rightarrow \bullet\beta_2, x']$  can belong to the state  $[\$]_{\hat{G}}$  only if  $[S_2 \rightarrow \bullet\beta, x'] \text{ desc}^* [B \rightarrow \bullet\beta_2, x']$ .

Finally, proving the theorem in the opposite direction is trivial: items  $[S_2 \rightarrow \bullet\beta_2, x']$  and  $[A \rightarrow \bullet\beta_2, x']$  in  $[\$]_{\hat{G}}$  imply a reduce-reduce conflict between items  $[S_2 \rightarrow \beta_2\bullet, x']$  and  $[A \rightarrow \beta_2\bullet, x']$  in  $[\$ \beta_2]_{\hat{G}}$ , so  $\hat{G} \notin \text{LR}(k)$ . ■

Theorem 1 provides two important insights into the embedded left LR( $k$ ) parsing. First, it guarantees that by simply checking the state  $[\$ \beta_2]$  for reduce-reduce conflicts one can tell whether the embedded left LR( $k$ ) parser for parsing substrings derived from the sentential form  $\beta_2$  can be made. As the sentential form  $\beta_2$  is a part of the right side of a production, it is usually short and therefore the method based on Theorem 1 is significantly faster than the brute-force approach.

Second, Theorem 1 illustrates that once again it is the left recursion that causes problems. But this is not to be worried about since it is clear that any substring derived from the left recursive nonterminal must be parsed entirely by an LR parser. In other words, Theorem 1 indicates that if the grammar is made so that the left recursive nonterminals are kept as low as possible in the resulting derivative trees, the substrings actually parsed using the embedded left LR( $k$ ) parsers tend to be short.

## V. CONCLUSION

The embedded left LR( $k$ ) parser has been obtained by modifying the left LR( $k$ ) parser in two ways. First, the left LR( $k$ ) parser was made capable of computing the viable suffix which the unread part of the input string is derived from. Second, it was simplified not to leave any markers on the stack about which subparses accumulated on the stack have been printed out already — as the parser stops after the first “long” reduction anyway. However, the algorithm for minimizing the embedded left LR( $k$ ) parser, i.e., for removing states that are not reachable before the first long reduction is performed, is still to be formalized.

At present, both, the backbone LL parser and the embedded LR parsers, need to use the lookahead buffer of the same length. However, if the LL parser was built around LA( $k$ )LL( $\ell$ ) parser (where  $k \geq \ell$ ) as defined in [2], then the combined parsing could most probably be formulated as the combination of LL( $\ell$ ) and LR( $k$ ) parsing (note that  $\text{LL}(\ell) \subseteq \text{LA}(\ell')\text{LL}(\ell)$  for any  $\ell' \geq \ell$ ). This would make the combined parser even more memory efficient.

Furthermore, the left LR( $k$ ) parser could be based on the LA( $k$ )LR( $\ell$ ) parser (most likely for  $\ell = 0$ ) instead of on the canonical LR( $k$ ) parser. This would further reduce the parsing tables while the strength of the resulting combined parser would be reduced from LR( $k$ ) to LA( $k$ )LR( $\ell$ ): not a significant issue as today LA(1)LR(0) is used instead of LR(1) whenever LR parsing is applied.

Finally, by using an LL( $k$ ) parser augmented by the embedded left LR( $k$ ) parsers instead of the left LR( $k$ ) parser the error recovery can be made much better — especially if the error recovery of the embedded left LR( $k$ ) parsers is made using the method described in [10].

## REFERENCES

- [1] S. Sippu and E. Soisalon-Soininen, *Parsing Theory, Vol. I: Languages and Parsing*, Berlin Heidelberg, Germany: Springer-Verlag, 1988.
- [2] S. Sippu and E. Soisalon-Soininen, *Parsing Theory, Vol. II: LL( $k$ ) and LR( $k$ ) Parsing*, Berlin Heidelberg, Germany: Springer-Verlag, 1990.
- [3] D. E. Knuth, *On the translation of languages from left to right*, Information and Control (1965), vol. 8, no. 6, pp. 607–639.
- [4] P. M. Lewis II and R. E. Stearns, *Syntax-directed transduction*, Proceedings of the 7th Annual IEEE Symposium on Switching and Automata Theory, New York, USA (1966), pp. 21-35.
- [5] M. Might and D. Darais, *Yacc is dead*, available online at Cornell University Library (arXiv.org:1010.5023), 2009.
- [6] T. Parr and K. S. Fischer, *LL(\*): The Foundation of the ANTLR Parser Generator*, accepted at the 32nd ACM SIGPLAN conference PLDI 2011.
- [7] P. R. Henriques, M. J. Varando Pereira, M. Mernik, M. Lenič, J. G. Gray, H. Wui, *Automatic generation of language-based tools using the LISA system*, IEE Proceedings - Software (2005), vol. 152, no. 2, pp. 54–69.
- [8] B. Slivnik and B. Vilfan, *Producing the left parse during bottom-up parsing*, Information Processing Letters (2005), vol. 96, no. 6, pp. 220–224.
- [9] J. P. Schmeiser and D. T. Barnard, *Producing a top-down parse order with bottom-up parsing*, Information Processing Letters (1995), vol. 54, no. 6, pp. 323–326.
- [10] B. Slivnik and B. Vilfan, *Improved error recovery in generated LR parsers*, Informatica (2004), vol. 28, no. 3, pp. 257–263.

# Indexing Trees by Pushdown Automata for Nonlinear Tree Pattern Matching

J. Trávníček, J. Janoušek, B. Melichar  
Department of Theoretical Computer Science  
Faculty of Information Technology  
Czech Technical University in Prague  
Thákurova 9, 160 00 Prague 6, Czech Republic

Email: travnja3@fit.cvut.cz, Jan.Janousek@fit.cvut.cz, melichar@fit.cvut.cz

This research has been partially supported by the Czech Ministry of Education, Youth and Sports under research program MSMT 6840770014, and by the Czech Science Foundation as project No. 201/09/0807.

**Abstract**—A new kind of an acyclic pushdown automaton for an ordered tree is presented. The *nonlinear tree pattern pushdown automaton* represents a complete index of the tree for nonlinear tree patterns and accepts all nonlinear tree patterns which match the tree. Given a tree with  $n$  nodes, the number of such nonlinear tree patterns is  $\mathcal{O}((2+v)^n)$ , where  $v$  is the number of variables in the patterns. We discuss time and space complexities of the nondeterministic nonlinear tree pattern pushdown automaton and a way of its implementation. The presented pushdown automaton is input-driven and therefore can be determinised.

## I. INTRODUCTION

TREES are one of the fundamental data structures used in Computer Science. Finding occurrences of tree patterns in trees is an important problem with many applications such as compiler code selection, interpretation of nonprocedural languages, implementation of rewriting systems, or various tree finding and tree replacement systems. Tree patterns containing variables which represent specific subtrees are called nonlinear tree patterns. Nonlinear tree pattern matching is used especially in the implementation of term rewriting systems, in which the terms can be represented as tree structures with nonlinear variables.

Generally, there exist two basic approaches to the problem of pattern matching. The first approach is represented by the use of a pattern matcher which is constructed for patterns. In other words, the patterns are preprocessed. Given a tree of size  $n$ , such tree pattern matcher typically perform the search phase in time linear in  $n$  [10]. The second basic approach is represented by the use of an indexing structure constructed for the subject in which we search. In other words, the subject is preprocessed. Examples of such indexing structures are suffix or factor automata [6, 7, 17, 19] for strings or subtree pushdown automaton [13], which represents a complete index of a tree for subtrees.

Trees can also be seen as strings, for example in their prefix (also called preorder) or postfix (also called postorder) notation. A linear notation of a tree can be obtained by the corresponding traversing of the tree. Moreover, every sequential algorithm on a tree traverses nodes of the tree in

a sequential order and follows a linear notation of the tree. [15] shows that the deterministic pushdown automaton (PDA) is an appropriate model of computation for labelled ordered trees in postfix notation and that the trees in postfix notation acceptable by deterministic PDA form a proper superclass of the class of regular tree languages [9], which are accepted by finite tree automata.

This paper describes a new kind of acyclic pushdown automaton, *nonlinear tree pattern pushdown automaton*, which represents a complete index of the tree for nonlinear tree patterns and accepts all nonlinear tree patterns which match the tree. Given a tree with  $n$  nodes, the number of such nonlinear tree patterns is  $\mathcal{O}((2+v)^n)$ , where  $v$  is the number of variables in the patterns. We describe the construction of the nonlinear tree pattern pushdown automaton and discuss its time and space complexities. The presented nondeterministic pushdown automaton is input-driven and therefore can be determinised. The deterministic version would accept an input nonlinear tree pattern of size  $m$  in time linear in  $m$  and not depending on  $n$ , but its disadvantage would be its large space complexity.

The presented nonlinear tree pattern pushdown automaton is analogous to string nondeterministic factor automaton [17]. The pushdown symbol alphabet contains just one pushdown symbol and therefore the pushdown store can be implemented by a single integer counter. Therefore, efficient methods for implementing nondeterministic string factor automata, such as [8], can easily be used also for the implementation of nondeterministic nonlinear tree pattern pushdown automata.

We note that the presented PDAs can be easily transformed to counter automata, which are a weaker and simpler model of computation than the PDA. We present the automata in this paper as PDAs, because the PDA is a more fundamental and more widely-used model of computation than the counter automaton.

Since the tree indexing data structure is to accept a finite language, a finite automaton could also be constructed. However, this automaton would have significantly more states than the PDA, in which the underlying tree structure is processed by the pushdown store.



The paper is organised as follows. The second section discusses a related work describing a nonlinear tree pattern matching algorithm. The third section contains basic definitions. Three types of pushdown automata that indexes trees for nonlinear pattern matching are presented. The fourth section describes special case of pushdown automaton. The first type, described in the fifth section, is a basic pushdown automaton that represents the basic idea of indexing trees for nonlinear pattern matching with one variable. The second type, described in the sixth section, is an optimisation of the basic pushdown automaton. The optimised pushdown automaton called nonlinear tree pattern pushdown automaton is smaller but accepts the same language as the basic one. The subsequent section is devoted to the indexing for more than one variable in nonlinear tree patterns. The last section is a conclusion.

## II. RELATED WORK

Some algorithms for nonlinear tree pattern matching are known. Nonlinear tree pattern matching algorithm described in [18] uses the approach which is represented by the pre-processing of the nonlinear input tree pattern. The algorithm reads Euler notation of both a subject tree and a nonlinear tree pattern. Euler notation is a tree linear notation, which contains a node each time it is visited during the preorder traversing of the tree. This means that every node appears exactly  $1 + \text{arity}(\text{node})$ -times in the Euler notation. Our method presented in this paper uses a standard tree prefix notation, which contains every node just once, for the first visit during the preorder traversing of the tree and of the input pattern.

In [18] factors which represent some subtrees in a subject tree in Euler notation are constructed. Aho-Corasick automaton is then constructed for these factors. The subject tree in Euler notation is processed by the constructed Aho-Corasick automaton and a binary array is constructed for each factor of the nonlinear tree pattern. If symbol 1 is at position  $i$  in the binary array it means that the corresponding factor of the pattern string is a suffix of the prefix (to  $i$ -th symbol) of Euler notation of the subject tree. In this way the nonlinear variables are matched. In our method presented in this paper we construct a complete index of the subject tree for a given maximal number of variables and do not construct any additional matching automata.

## III. BASIC NOTIONS

We define notions on trees similarly as they are defined in [1, 9, 10].

### A. Alphabet

An *alphabet* is a finite nonempty set of *symbols*. A *ranked alphabet* is a finite nonempty set of symbols each of which has a unique nonnegative *arity* (or *rank*). Given a ranked alphabet  $\mathcal{A}$ , the arity of a symbol  $a \in \mathcal{A}$  is denoted  $\text{Arity}(a)$ . The set of symbols of arity  $p$  is denoted by  $\mathcal{A}_p$ . Elements of arity 0, 1, 2,  $\dots$ ,  $p$  are respectively called nullary (constants), unary, binary,  $\dots$ ,  $p$ -ary symbols. We assume that  $\mathcal{A}$  contains at least

one constant. In the examples we use numbers at the end of identifiers for a short declaration of symbols with arity. For instance,  $a_2$  is a short declaration of a binary symbol  $a$ .

### B. Tree, tree pattern, tree template

Based on concepts from graph theory (see [1]), a tree over an alphabet  $\mathcal{A}$  can be defined as follows:

A *directed graph*  $G$  is a pair  $(N, R)$ , where  $N$  is a set of nodes and  $R$  is a set of edges such that each element of  $R$  is of the form  $(f, g)$ , where  $f, g \in N$ . This element will indicate that, for node  $f$ , there is an edge leaving  $f$ , entering node  $g$ .

A sequence of nodes  $(f_0, f_1, \dots, f_n)$ ,  $n \geq 1$ , is a *path* of length  $n$  from node  $f_0$  to node  $f_n$  if there is an edge which leaves node  $f_{i-1}$  and enters node  $f_i$  for  $1 \leq i \leq n$ . A *cycle* is a path  $(f_0, f_1, \dots, f_n)$ , where  $f_0 = f_n$ . An ordered *dag* (dag stands for Directed Acyclic Graph) is an ordered directed graph that has no cycle. A *labelling* of an ordered graph  $G = (N, R)$  is a mapping of  $N$  into a set of labels. In the examples we use  $a_f$  for a short declaration of node  $f$  labelled by symbol  $a$ .

Given a node  $f$ , its *out-degree* is the number of distinct pairs  $(f, g) \in R$ , where  $g \in N$ . By analogy, the *in-degree* of node  $f$  is the number of distinct pairs  $(g, f) \in R$ , where  $g \in N$ .

A *tree* is an acyclic connected graph. Any node of a tree can be selected as a *root* of the tree. A tree with a root is called *rooted tree*. Nodes of the tree with out-degree 0 are called *leaves*.

A tree can be *directed*. A *rooted and directed tree*  $t$  is a dag  $t = (N, R)$  with a special node  $r \in N$ , called the *root*, such that (1)  $r$  has in-degree 0, (2) all other nodes of  $t$  have in-degree 1, (3) there is just one path from the root  $r$  to every  $f \in N$ , where  $f \neq r$ .

A *labelled, (rooted, directed) tree* is a tree having the following property: (4) every node  $f \in N$  is labelled by a symbol  $a \in \mathcal{A}$ , where  $\mathcal{A}$  is an alphabet.

A *ranked, (labelled, rooted, directed) tree* is a tree labelled by symbols from a ranked alphabet and out-degree of a node  $f$  labelled by symbol  $a \in \mathcal{A}$  equals to  $\text{Arity}(a)$ . Nodes labelled by nullary symbols (constants) are leaves.

An *ordered, (ranked, labelled, rooted, directed) tree* is a tree where direct descendants  $a_{f_1}, a_{f_2}, \dots, a_{f_n}$  of a node  $a_f$  having an  $\text{Arity}(a_f) = n$  are ordered.

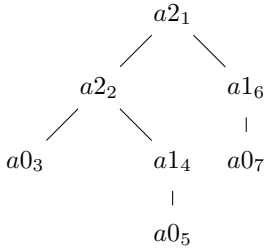
*Example 1* Consider a ranked alphabet  $\mathcal{A} = \{a_2, a_1, a_0\}$ . Consider an ordered, ranked, labelled, rooted, and directed tree  $t_1 = (\{a_{2_1}, a_{2_2}, a_{0_3}, a_{1_4}, a_{0_5}, a_{1_6}, a_{0_7}\}, R_1)$  over  $\mathcal{A}$ , where  $R_1$  is a set of the following ordered pairs:

$$R_1 = \{(a_{2_1}, a_{2_2}), (a_{2_1}, a_{1_6}), (a_{2_2}, a_{0_3}), (a_{2_2}, a_{1_4}), (a_{1_4}, a_{0_5}), (a_{1_6}, a_{0_7})\}.$$

The tree  $t_1$  written in prefix notation is  $\text{pref}(t_1) = a_2 a_2 a_0 a_1 a_0 a_1 a_0$ .

Trees can be represented graphically, and tree  $t_1$  is illustrated in Figure 1.

The height of a tree  $t$ , denoted by  $\text{Height}(t)$ , is defined as the maximal length of a path from the root of  $t$  to a leaf of  $t$ .


 Fig. 1: Tree  $t_1$  from Example 1

To define a *tree pattern*, we use a special nullary symbol  $S$ , not in alphabet  $\mathcal{A}$ ,  $\text{Arity}(S) = 0$ , which serves as a placeholder for any subtree. A tree pattern is defined as a labelled ordered tree over an alphabet  $\mathcal{A} \cup \{S\}$ . We will assume that the tree pattern contains at least one node labelled by a symbol from  $\mathcal{A}$ . A tree pattern containing at least one symbol  $S$  will be called a *tree template*.

A tree pattern  $p$  with  $k \geq 0$  occurrences of the symbol  $S$  *matches* a subject tree  $t$  at node  $n$  if there exist subtrees  $t_1, t_2, \dots, t_k$  (not necessarily the same) of the tree  $t$  such that the tree  $p'$ , obtained from  $p$  by substituting the subtree  $t_i$  for the  $i$ -th occurrence of  $S$  in  $p$ ,  $i = 1, 2, \dots, k$ , is equal to the subtree of  $t$  rooted at  $n$ .

The *nonlinear tree pattern* uses nullary symbols  $X, Y, \dots$  which are not in alphabet  $\mathcal{A}$ . These symbols have arity equal to zero. These symbols serve as a placeholders for any subtree. Additionally every occurrence of for example symbol  $X$  in *nonlinear tree pattern* is matched with the same subject subtree. A nonlinear tree pattern has to contain at least one symbol from  $\mathcal{A}$ . A nonlinear tree pattern which contains at least two symbols  $X$  will be called *nonlinear tree template*. Symbol  $X$  is called a *nonlinear variable*.

A nonlinear tree pattern  $np$  with  $k \geq 2$  occurrences of the symbol  $X$  *matches* an subject tree  $t$  at node  $n$  if there exist the same subtrees  $t_1, t_2, \dots, t_k$  of the tree  $t$  such that the tree  $np'$ , obtained from  $np$  by substituting the subtree  $t_i$  for the  $i$ -th occurrence of  $X$  in  $np$ ,  $i = 1, 2, \dots, k$ , is equal to the subtree of  $t$  rooted at  $n$ .

*Example 2* Consider a tree  $t_1 = (\{a21, a22, a03, a14, a05, a16, a07\}, R_1)$  from Example 1, which is illustrated in Figure 1.

Consider a tree template  $p_1$  over  $\mathcal{A} \cup \{S\}$ ,  $p_1 = (\{a21, S_2, a13, S_4\}, R_{p_1})$ , where  $R_{p_1}$  is a set of lists of the following ordered pairs:

$$R_{p_1} = \{(a21, S_2), (a21, a13), (a13, S_4)\}.$$

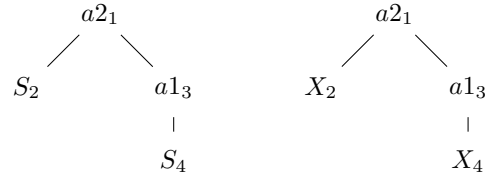
The tree template  $p_1$  written in prefix notation is  $\text{pref}(p_1) = a2 S a1 S$ .

Consider a nonlinear tree template  $p_2$  over  $\mathcal{A} \cup \{S, X\}$ ,  $p_2 = (\{a21, X_2, a13, X_4\}, R_{p_2})$ , where  $R_{p_2}$  is a set of lists of the following ordered pairs:

$$R_{p_2} = \{(a21, X_2), (a21, a13), (a13, X_4)\}.$$

Note that symbol  $S$  can occur in nonlinear tree template and it serves as unbounded variable.

The tree template  $p_2$  written in prefix notation is  $\text{pref}(p_2) = a2 X a1 X$ .


 Fig. 2: Tree template  $p_1$  (left) and nonlinear tree template  $p_2$  (right) from Example 2

Tree templates  $p_1$  and  $p_2$  are illustrated in Figure 2. Tree template  $p_1$  has two occurrences in tree  $t_1$  – it matches at nodes 1 and 2 of  $t_1$ . Nonlinear tree template  $p_2$  has one occurrence in tree  $t_1$  – it matches at node 2 of  $t_1$ .

### C. Language, finite and pushdown automata

We define notions from the theory of string languages similarly as they are defined in [1, 11].

A *language* over an alphabet  $\mathcal{A}$  is a set of strings over  $\mathcal{A}$ . Symbol  $\mathcal{A}^*$  denotes the set of all strings over  $\mathcal{A}$  including the empty string, denoted by  $\varepsilon$ . Set  $\mathcal{A}^+$  is defined as  $\mathcal{A}^+ = \mathcal{A}^* \setminus \{\varepsilon\}$ . Similarly, for string  $x \in \mathcal{A}^*$ , symbol  $x^m$ ,  $m \geq 0$ , denotes the  $m$ -fold concatenation of  $x$  with  $x^0 = \varepsilon$ . Set  $x^*$  is defined as  $x^* = \{x^m : m \geq 0\}$  and  $x^+ = x^* \setminus \{\varepsilon\} = \{x^m : m \geq 1\}$ .

A *nondeterministic pushdown automaton* (nondeterministic PDA) is a seven-tuple  $M = (Q, \mathcal{A}, G, \delta, q_0, Z_0, F)$ , where  $Q$  is a finite set of *states*,  $\mathcal{A}$  is an *input alphabet*,  $G$  is a *pushdown store alphabet*,  $\delta$  is a mapping from  $Q \times (\mathcal{A} \cup \{\varepsilon\}) \times G$  into a set of finite subsets of  $Q \times G^*$ ,  $q_0 \in Q$  is an initial state,  $Z_0 \in G$  is the initial pushdown store symbol, and  $F \subseteq Q$  is the set of final (accepting) states.

Triple  $(q, w, x) \in Q \times \mathcal{A}^* \times G^*$  denotes the configuration of a pushdown automaton. We will write the top of the pushdown store  $x$  on its left hand side. The initial configuration of a pushdown automaton is a triple  $(q_0, w, Z_0)$  for the input string  $w \in \mathcal{A}^*$ . The relation  $\vdash_M \subseteq (Q \times \mathcal{A}^* \times G^*) \times (Q \times \mathcal{A}^* \times G^*)$  is a *transition* of a pushdown automaton  $M$ . It holds that  $(q, aw, \alpha\beta) \vdash_M (p, w, \gamma\beta)$  if  $(p, \gamma) \in \delta(q, a, \alpha)$ . The  $k$ -th power, transitive closure, and transitive and reflexive closure of the relation  $\vdash_M$  is denoted  $\vdash_M^k, \vdash_M^+, \vdash_M^*$ , respectively.

A pushdown automaton is *input-driven* if each of its pushdown operations is determined only by the input symbol.

A language  $L$  accepted by a pushdown automaton  $M$  is defined in two distinct ways:

- 1) *Accepting by final state*:  $L(M) = \{x : (q_0, x, Z_0) \vdash_M^* (q, \varepsilon, \gamma) \wedge x \in \mathcal{A}^* \wedge \gamma \in G^* \wedge q \in F\}$ .
- 2) *Accepting by empty pushdown store*:  $L_\varepsilon(M) = \{x : (q_0, x, Z_0) \vdash_M^* (q, \varepsilon, \varepsilon) \wedge x \in \mathcal{A}^* \wedge q \in Q\}$ .

If the pushdown automaton accepts the language by empty pushdown store, then the set  $F$  of final states is the empty set.

Unreachable states are states  $p \in Q$  from automaton  $M = (Q, \mathcal{A}, G, \delta, q_0, Z_0, F)$  which are not reachable from the initial state because there is no sequence of transitions from the initial state to that particular state  $p$ . Formally, there are no transitions that allow  $(q_0, kw, Z_0) \vdash_M^+ (p, w, \gamma)$ .

Unnecessary states are states  $p \in Q$  from automaton  $M = (Q, \mathcal{A}, G, \delta, q_0, Z_0, F)$  which are not connected to any final state  $f \in F$  if automaton accepts by final states, or not

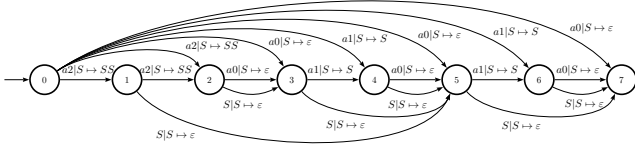


Fig. 3: Transition diagram of nondeterministic tree pattern pushdown automaton  $M_{npt}(pref(t_1))$  from Example 3

connected to any state, where  $\gamma \in G^*$  may be  $\varepsilon$ , if automaton accepts by empty pushdown store.

Pushdown automaton  $M = (Q, \mathcal{A}, G, \delta, q_0, Z_0, F)$  is acyclic if it does not contain transitions  $\delta(q, x_1, \gamma_1) \vdash_M^+ (q, x_2, \gamma_2)$ , where  $xx_2 = x_1$ ,  $x \neq \varepsilon$  and  $q \in Q$ .

#### IV. INDEXING TREES FOR TREE PATTERN MATCHING

*Tree pattern pushdown automata* are introduced in [14, 16] as an extension of subtree PDA. The tree pattern pushdown automaton represents a complete index of a tree for linear tree patterns and accepts all tree patterns that match the tree.

*Example 3* Consider a tree  $t_1$  in prefix notation  $pref(t_1) = a2 a2 a0 a1 a0 a1 a0$  from Example 1, which is illustrated in Figure 1. The tree pattern pushdown automaton accepting all tree patterns matching tree  $t_1$  is nondeterministic pushdown automaton  $M_{npt}(pref(t_1)) = (\{0, 1, 2, 3, 4, 5, 6, 7\}, \mathcal{A}, \{S\}, \delta_5, 0, S, \emptyset)$ . Its transition diagram is illustrated in Figure 3.

#### V. INDEXING TREES BY BASIC NONLINEAR TREE PATTERN PUSHDOWN AUTOMATON

A nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b = (\{0, 1, 2, \dots, n, x_1, \dots, n_1, y_2, \dots, n_2, \dots, z_m, \dots, n_m\}, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$  accepts all nonlinear tree patterns which can occur in a subject tree.

##### A. Construction of basic indexing automaton for nonlinear tree pattern matching

In our indexing pushdown automata we construct special parts called tails, which represent parts accessible after reading an input symbol of a nonlinear variable. Such a symbol selects a particular tail.

The  $tail(M, q_t, Z_t)$  of an automaton  $M = (Q, \mathcal{A}, G, \delta, q_0, Z_0, F)$ , where  $M$  is an acyclic tree pattern pushdown automaton, is defined as  $tail(M, q_t, Z_t) = (Q_t, \mathcal{A}, G, \delta_t, q_t, Z_t, F)$ .  $Q_t = Q \setminus Q_{us}$ ,  $Q_{us}$  is a set of unreachable states from  $q_t$  when pushdown store operations are omitted,  $q_t \in Q_t$  is a new initial state of an automaton,  $\delta_t = \delta \setminus \delta_{us}$ ,  $\delta_{us}$  are transitions leading from or to state  $q_n \in Q_{us}$ .

*Example 4* Given a tree pattern pushdown automaton  $M_{npt}(pref(t_1))$ , which is an index of tree  $t_1$  from Example 1 shown in Figure 3. The tail of automaton with initial state 3 is  $tail(M_{npt}, 3, S) = (Q, \mathcal{A} \cup \{S\}, \{S\}, \delta, 3, S, \emptyset)$  constructed from tree pattern pushdown automaton shown in Figure 3,  $S$  is the initial symbol of the pushdown store. The corresponding transition diagram is illustrated in Figure 4.

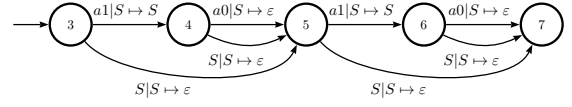


Fig. 4: Tail of tree pattern pushdown automaton  $M_{tail}(M_{npt}, 3, \varepsilon)$  from Example 4

We note that every node of a tree  $t$  is the root of just one complete subtree  $S$ . Prefix notation of such subtree  $pref(S)$  is a factor of  $pref(t_1)$ . These factors are in the tree pushdown automaton "skipped" by transitions for input symbol  $S$ .

A labelled path in the automaton  $M_{npt}$  between states  $q$  and  $q_t$ , where  $q_t$  is defined by transition  $(q_t, \varepsilon) \in \delta(q, S, S)$  will be denoted  $sst(q) = b_1 b_2 \dots b_m$  and it represents a subtree in the linear notation.  $sst(q)$  is used in Algorithm 2 to determine which subtree of subject tree was "assigned" to particular automaton tail.

The construction consists of two algorithms. Algorithm 2 constructs tails from the original tree pattern pushdown automaton. Algorithm 1 connects recursively these created tails to the automaton being created.

**Algorithm 1** Construction of nondeterministic basic nonlinear tree pattern pushdown automaton.

**Input:** Nondeterministic tree pattern pushdown automaton  $M_{npt}$ .

**Output:** Nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b$ .

**Method:**

1. For each transition  $(q_t, \varepsilon) \in \delta(q, S, S)$  in automaton  $M_{npt}$  do:
  - 1.1. Create  $M_{tmp} = nta(tail(M_{npt}, q_t, S), sst(q))$  using Algorithm 2.
  - 1.2. Add new state  $q_{id}$  to  $M_{npt}$  where  $q_{id}$  is copy of state  $q_t$ .
  - 1.3. Add new transition  $(q_{id}, \varepsilon) \in \delta(q, X, S)$  to  $M_{npt}$ .
  - 1.4. Add  $M_{tmp}$  to  $M_{npt}$  and merge initial state of  $M_{tmp}$  with  $q_{id}$ .
2.  $M_b$  is  $M_{npt}$ .

**Algorithm 2** Recursive construction of tail of nondeterministic basic nonlinear tree pattern automaton.

**Input:** Tail of nondeterministic tree pattern pushdown automaton  $M_{tnpt}$ , string representing subtree skipped by transition  $x = sst(q)$ .

**Output:** Recursively created tail  $nta(M_{tnpt}, x)$ .

**Method:**

1. For each transition  $(q_t, \varepsilon) \in \delta(q, S, S)$  in automaton  $M_{tnpt}$  where  $sst(q) = x$  do:
  - 1.1. Create  $M_{tmp} = nta(tail(M_{tnpt}, q_t, S), x)$  using Algorithm 2.
  - 1.2. Add new state  $q_{id}$  to  $M_{tnpt}$  where  $q_{id}$  is copy of state  $q_t$ .
  - 1.3. Add new transition  $(q_{id}, \varepsilon) \in \delta(q, X, S)$  to  $M_{tnpt}$ .

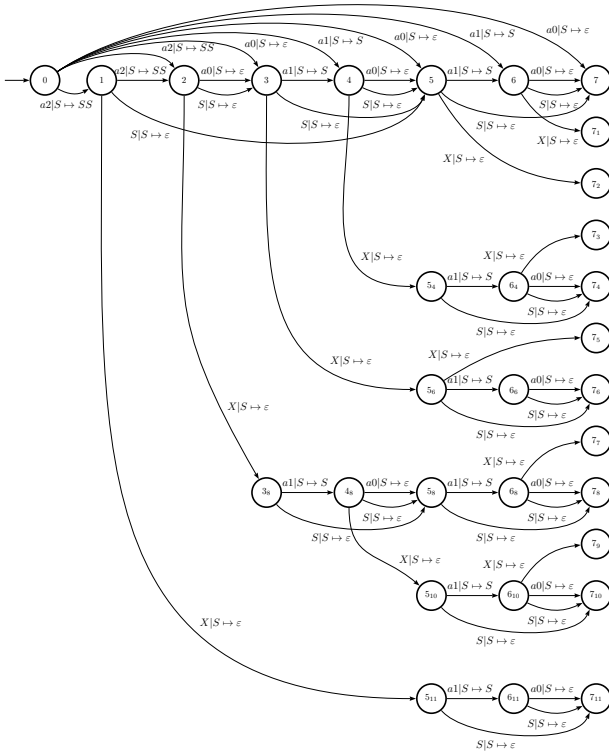


Fig. 5: Nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b(t_1)$  from Example 5 constructed for subject tree shown in Figure 1

1.4. Add  $M_{tmp}$  to  $M_{tnpt}$  and merge initial state of  $M_{tmp}$  with  $q_{id}$ .

2.  $nta(M_{tnpt}, x)$  is  $M_{tnpt}$ .

The difference between Algorithm 2 and Algorithm 1 is that Algorithm 2 calls itself only when processing transition for symbol  $S$  leading from state  $q$ , where  $sst(q)$  equals its subtree parameter. On the other hand, Algorithm 1 calls Algorithm 2 for each transition for symbol  $S$ .

*Example 5* Given a string  $p = a2 a2 a0 a1 a0 a1 a0$ , which is a prefix notation of tree  $t_1$  from Example 1, the corresponding nondeterministic basic nonlinear tree pattern pushdown automaton is  $M_b(t_1) = (Q, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$ , where its transition diagram is illustrated in Figure 5.

## VI. INDEXING TREES BY NONLINEAR TREE PATTERN PUSHDOWN AUTOMATON

Some states of an automaton created by Algorithm 1 may be merged so that states in nondeterministic nonlinear tree pattern pushdown automaton  $M_o = (\{0, 1, 2, \dots, n, x_1, \dots, x_n, y_2, \dots, y_2, \dots, z_m, \dots, n_m\}, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$  will still track both virtually assigned subtree and the same number of nonlinear variables read from the pattern. Merged states are those from tails with the same virtually assigned subtree and the same number of nonlinear variables read.

### A. Constructing indexing automaton for nonlinear tree pattern matching

*Definition 1* A tree node state label  $tnsl(q)$  is the sequence number of tree node of subject tree written in prefix notation. The  $tnsl(q)$  is equivalent to automaton state label, where  $q \in Q$  from automaton  $M_b = (Q, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, q_0, S, \emptyset)$ . The  $tnsl(q)$  is the main number from state label.

**Algorithm 3** Algorithm for counting the  $tnsl$ .

**Input:** Nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b$ , state  $q$  for which count the  $tnsl$ .

**Output:** Number representing  $tnsl$ .

**Variables:** Temporary number  $n$ , State *initial*.

**Method:**

1.  $n = 0$ . *initial* is initial state of automaton  $M_b$ .
2. While  $q \neq \text{initial}$  do:
  - 2.1. If exists transition  $(q, S^{arity(a)}) \in \delta(q_{prev}, a, S)$  where  $a \in \mathcal{A}$  and  $q_{pref}$  is preferably not  $q_0$  do:
    - 2.1.1.  $n = n + 1$ ,  $q = q_{prev}$ .
    - 2.1.2. Continue with step [2.].
  - 2.2. If exists transition  $(q, \varepsilon) \in \delta(q_{prev}, X, S)$  where  $X$  is nonlinear variable do:
    - 2.2.1.  $n = n + |sst(q_{prev})|$ ,  $q = q_{prev}$ .
    - 2.2.2. Continue with step [2.].

*Example 6* Given a nonlinear nondeterministic basic tree pattern pushdown automaton for pattern matching is  $M_b(t_1) = (Q, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$ , having its transition diagram shown in Figure 5.

The  $tnsl(3) = 3$ ,  $tnsl(5_4) = 5$ ,  $tnsl(7_{11}) = 7$ ,  $tnsl(7_9) = 7$ .

*Definition 2* A number of nonlinear variables  $nnv(q, X)$  is the number of transitions for nonlinear variable  $X$  on the path from the initial state  $q_0$  to state  $q$ , where  $q$  and  $q_0 \in Q$  of a nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b = (Q, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$  created by Algorithm 1.

**Algorithm 4** Algorithm for counting the  $nnv$ .

**Input:** Nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b$ , state  $q$  for which count the  $tnsl$ .

**Output:** Number representing  $nnv$ .

**Variables:** Temporary number  $n$ , State *initial*.

**Method:**

1.  $n = 0$ . *initial* is initial state of automaton  $M_b$ .
2. While  $q \neq \text{initial}$  do:
  - 2.1. If exists transition  $(q, S^{arity(a)}) \in \delta(q_{prev}, a, S)$  where  $a \in \mathcal{A}$  and  $q_{pref}$  is preferably not  $q_0$  do:
    - 2.1.1.  $q = q_{prev}$ .
    - 2.1.2. Continue with step [2.].
  - 2.2. If exists transition  $(q, \varepsilon) \in \delta(q_{prev}, X, S)$  where  $X$  is nonlinear variable do:
    - 2.2.1.  $n = n + 1$ ,  $q = q_{prev}$ .
    - 2.2.2. Continue with step [2.].

**Definition 3** The starting states of optimisation  $sso(M_b)$  is a collection of (key, value) pairs, where key is a triplet  $(sst(q), nnv(u, X), tnsi(u))$  and value is a set of states. The  $sso(M_b)$  stores sets of states with the same number of transitions for nonlinear variable  $X$   $nnv(q, X)$  and subtree skipped by transition  $sst(q)$ , which denotes the sets of states from nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b$  created by Algorithm 1. The  $sso(M_b) = \{(sst(q_x), nnv(s_{a1}, X), tnsi(s_{a1})), \{s_{a1}, s_{a2}, \dots\}, (sst(q_y), nnv(s_{b1}, X), tnsi(s_{b2})), \{s_{b1}, s_{b2}, \dots\}, \dots\}$ , where the first state  $s_1$  from each set is the main state. State  $v$  is  $sst(v)$  denoting state for state  $s_1$  given by  $(v, X\omega, S\gamma) \vdash (s_1, \omega, \gamma)$ , where  $\omega = (\mathcal{A} \cup \{S, X\})^*$ . All states from that set are given by following:  $\{\forall s : nnv(s, X) = nnv(s_1, X) \text{ and } sst(v) = sst(u) \text{ and } tnsi(s) = tnsi(s_1); s, s_1, u, v \in Q\}$ , where state  $u$  is  $sst(u)$  denoting state for state  $s$  given by  $(u, X(\mathcal{A} \cup \{S\})^*\omega, S^\alpha) \vdash^* (s, \omega, S^\beta)$ , where  $\omega = (\mathcal{A} \cup \{S, X\})^*$ .

Each set from the collection of sets of states  $sso(M_b)$  defines states from nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b$  that can be merged and the resulting automaton is called nondeterministic nonlinear tree pattern pushdown automaton  $M_o$ . States from each set defines the start of a merging process so that states that are reachable by the same sequence of transitions are also merged.

**Example 7** Given a string  $p = a2 a2 a0 a1 a0 a1 a0$ , which is the prefix notation of tree  $t_1$  from Example 1. The corresponding nondeterministic basic nonlinear tree pattern pushdown automaton is  $M_b(t_1) = (Q, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$ , where its transition diagram and states are illustrated in Figure 5.

All states that occur in one of the set in the collection  $sso(M_b)_o$  are target states from all transitions for a symbol  $X$  and the transitions for a symbol  $S$  which shares the source state.

$$sso(M_b) = \{((a0, 1, 5), \{5_4, 5_8\}), ((a0, 1, 7), \{7_1, 7_4, 7_8\}), ((a0, 2, 7), \{7_3, 7_7, 7_{10}\}), ((a1a0, 1, 7), \{7_2, 7_6\})\}.$$

**Algorithm 5** Construction of the nondeterministic nonlinear tree pattern pushdown automaton.

**Input:** Nondeterministic basic nonlinear tree pattern pushdown automaton  $M_b$ .

**Output:** Nondeterministic nonlinear tree pattern pushdown automaton  $M_o$ .

**Variables:** Collection of sets of states  $sso(M_b)$ .

**Method:**

1. For all transitions  $(u_1, \varepsilon) \in \delta(q, X, S)$  do:
  - 1.1. If the collection  $sso(M_b)$  does not contain a set on a key  $(sst(q), nnv(u_1, X), tnsi(u_1))$  create that set as an empty set.
  - 1.2. Add  $u_1$  to the collection  $sso(M_b)$  to the set on the key  $(sst(q), nnv(u_1, X), tnsi(u_1))$ .
2. For all transitions  $(u_2, \varepsilon) \in \delta(q, S, S)$ , where exists a transition  $(u_1, \varepsilon) \in \delta(q, X, S)$  do:

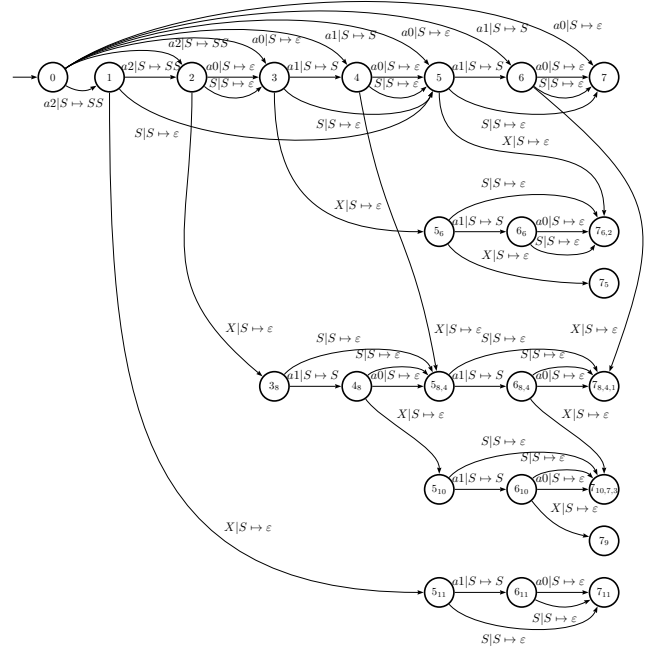


Fig. 6: Nondeterministic nonlinear tree pattern pushdown automaton  $M_o(t_1)$  from Example 8 constructed by Algorithm 5 for subject tree shown in Figure 1

2.3. If  $nnv(u_2, X) \neq 0$  and the collection  $sso(M_b)$  does not contain a set on a key  $(sst(q), nnv(u_2, X), tnsi(u_2))$  create that set as an empty set.

2.4. Add  $u_2$  to the collection  $sso(M_b)$  to the set on the key  $(sst(q), nnv(u_2, X), tnsi(u_2))$ .

3. For each set in the collection  $sso(M_b)$  do:

3.1. Merge all states in this set, along with all states that follows-up.

**Example 8** Given a string  $p = a2 a2 a0 a1 a0 a1 a0$ , which is the prefix notation of tree  $t_1$  from Example 1, the corresponding nondeterministic nonlinear tree pattern pushdown automaton is  $M_o(t_1) = (Q, \mathcal{A} \cup \{S, X\}, \{S\}, \delta, 0, S, \emptyset)$ , where merged states are in Example 7 and its transition diagram and states are illustrated in Figure 6.

The nondeterministic nonlinear tree pattern pushdown automaton can be even minimalised by omitting the  $nnv(q, X)$  part of the key value pairs of the collection  $sso(M_b)$ . A resulting automaton would represent an index of the subject tree for nonlinear tree pattern matching but would not be able to say how many nonlinear variables has been read during processing the nonlinear tree pattern.

## B. Time and Space Complexity Analysis

**Lemma 1** Time complexity of accepting pattern by automaton created by Algorithm 5 is  $\mathcal{O}(m)$ , where  $m$  is the number of nodes of a subject tree.

*Proof:* Automaton created for nonlinear pattern matching reads just one symbol from the input by every transition. Automaton accepts or rejects the input pattern at the latest

after reading the last symbol of pattern tree written in prefix notation. ■

*Lemma 2* Time complexity of accepting pattern by automaton created by Algorithm 5 is  $\mathcal{O}(\sum_S rs_i)$ , where  $S$  is the set of all prefixes except  $\varepsilon$  and  $rs_i$  is the number of distinct sequences of transitions in automaton  $M_o$  for  $s_i \in S$  which ends in valid state.

*Proof:* Automata has to try all possible sequences of transitions according to tree template which occur in nondeterministic nonlinear tree pattern automaton. Sequences of symbols of these transitions form a prefix of tree template. Prefix of size of one symbol from tree template is handled by exactly  $n$  steps where  $n$  is the number of all possible sequences of transitions in automaton for that prefix. Prefix of size of two symbols is handled by  $n + m$  steps where  $m$  is the number of all possible sequences of transitions in automaton for that prefix. Note that to handle two symbol prefix two transitions has to be processed, however the first transition is already accounted by prefix of size of one symbol.

Exact time complexity is then sum of all possible sequences of transitions in automaton for all prefixes of nonlinear tree template, which is  $\mathcal{O}(\sum_S rs_i)$ . ■

*Lemma 3* The number of states of nondeterministic nonlinear tree pattern pushdown automaton  $M_o$  (space complexity) created by Algorithm 5 is  $\mathcal{O}(m(\sum_{i=0}^s r_i))$ , where  $m$  is the number of nodes of a subject tree and  $\sum_{i=0}^s r_i$ , where  $s$  is the number of distinct subtrees, is the number of automaton tails, where  $r_i$  is the number of repetitions of each unique subtree.

*Proof:* Each occurrence of each unique subtree in tree increments the number of automaton tails, that were created for this subtree. The exact number of tails created for particular subtree is then  $r_i$ , where  $r_i$  is the number of repetitions of that subtree. Then the total number of tails for one nonlinear variable in automaton is the number of tails created for each unique subtree of indexed tree which is  $\sum_{i=0}^s r_i$ . The total number of tails does not count original automaton. The exact space complexity of automaton for one nonlinear variable is  $\mathcal{O}(m(\sum_{i=0}^s r_i + 1)) = \mathcal{O}(m(\sum_{i=0}^s r_i))$ . ■

*Lemma 4* The number of transitions of nondeterministic nonlinear tree pattern pushdown automaton  $M_o$  (space complexity) created by Algorithm 5 is  $\mathcal{O}(m^2 + m + \sum_{i=0}^s (\frac{r_i^2 + r_i}{2}))$ , where  $m$  is the number of nodes of a subject tree,  $s$  is the number of distinct subtrees and  $r_i$  is the number of repetitions of each unique subtree.

*Proof:* Given all tails for one nonlinear variable there are transitions for symbol  $X$  between these tails. There is one transition heading to the last tail. There are two transitions heading to the previous tail and so on. The number of transitions for symbol  $X$  is  $\sum_{i=0}^s (\frac{r_i^2 + r_i}{2})$ .

Using Lemma 3 the number of transitions for symbol  $S$  is  $\frac{1}{2}m^2$  and the number of transitions for symbol  $a \in \mathcal{A}$  is  $\frac{1}{2}m^2 + m$ .

The number of transitions then is  $\mathcal{O}(\sum_{i=0}^s (\frac{r_i^2 + r_i}{2}) + m^2 + m)$ . ■

*Lemma 5* Language defined by nondeterministic nonlinear tree pattern pushdown automaton  $M_o$  is  $\mathcal{O}(3^m)$ , where  $m$  is the number of nodes of a subject tree.

*Proof:* Consider a tree with  $m + 1$  nodes. Arity of the first node is  $m$ . Remaining nodes are labelled with the same nullary symbol. It is possible to create tree template where on each position of nullary symbol a nonlinear variable  $X$ , symbol  $S$  or the original symbol can be placed. Therefore on  $m$  positions it is possible to chose from three symbols. Language size is then  $\mathcal{O}(3^m)$ , where  $m + 1$  is the number of nodes of a subject tree. ■

## VII. CONSTRUCTION OF AUTOMATA FOR PATTERNS WITH MORE NONLINEAR VARIABLES

Patterns that contain more than one nonlinear variable are more common than those with one nonlinear variable. The algorithm for construction of nondeterministic nonlinear tree pattern pushdown automaton for more nonlinear variables  $M_{mo}$  or  $M_{mb}$  is basically algorithm for union of automata. Automaton for two nonlinear variables is union of two automata for one nonlinear variable.

### A. An algorithm of joining automata

*Definition 4* The nonlinear variable from automaton  $nva(M)$  is the nonlinear variable for which the nondeterministic nonlinear tree pattern pushdown automaton  $M_o$  was created for.

Consider that nondeterministic basic nonlinear tree pattern pushdown automaton  $M_{mo}$  for two nonlinear variables determined by  $X, Y$  is to be constructed. Nondeterministic nonlinear tree pattern pushdown automaton  $M_o$  for nonlinear variable determined by symbol  $X$  and second for nonlinear variable determined by symbol  $Y$  are constructed by Algorithm 5. The first automaton for nonlinear variable determined by symbol  $X$  handles nonlinear variable determined by symbol  $Y$  as linear variable usually determined by symbol  $S$ . The second automaton handles nonlinear variables similarly.

**Algorithm 6** Construction of Indexing automaton for more nonlinear variables.

**Input:** Set of nondeterministic nonlinear tree pattern pushdown automata  $M_o$ .

**Output:** Nondeterministic nonlinear tree pattern pushdown automaton for more nonlinear variables  $M_{mo} = (Q, \mathcal{A}, \{S\}, \delta', q_I, S, \emptyset)$ .

**Method:**

1. Create set of symbols of nonlinear variables  $nvs$  using  $nva$  from input set of automata.
2. For  $i = 0$  to  $sizeof(M_o)$ , step 1 do:
  - 2.1.  $M_{oi}$  is automaton from set  $M_o$  on index  $i$ .
  - 2.2. For each transition  $(u, \varepsilon) \in \delta(q, S, S)$  in automaton  $M_{oi}$  do:
    - 2.2.1. For each symbol  $s$  in  $nvs \setminus nva(M_{oi})$  add transition  $(u, \varepsilon) \in \delta(q, s, S)$  to automaton  $M_{oi}$ .
2. Create automaton  $M_{mo}$  as union of all automata in input set.

### B. Time and space complexity analysis

*Lemma 6* The space complexity of a nondeterministic nonlinear pattern pushdown automaton for more nonlinear variables  $M_{mo}$  is  $\mathcal{O}(t^v * m)$ , where  $t$  is the number of tails of the nondeterministic nonlinear tree pattern pushdown automaton for one nonlinear variable  $M_o$ ,  $v$  is the number of nonlinear variables and  $m$  is the number of nodes of a subject tree.

*Proof:* The automaton complexity is clear for one nonlinear variable. The number of tails in automaton for more nonlinear variables is result from the Cartesian product of tails in each automaton for one nonlinear variable. Each tail of the first automaton allows the second automaton to move across tails depending on the others nonlinear variables and so on for more automata. The number of tails is  $t^v$  for the Cartesian product, where  $t$  is the number of tails of original one nonlinear variable automaton. The number of tails is given by union of  $v$  automata for  $v$  nonlinear variables. So the exact space complexity of nondeterministic nonlinear tree pattern pushdown automaton  $M_{mo}$  for more nonlinear variables is  $\mathcal{O}(t^v * m)$ . ■

*Lemma 7* The language accepted by nondeterministic nonlinear tree pattern pushdown automaton  $M_{mo}$  for more nonlinear variables contains  $\mathcal{O}((2 + v)^n)$  sentences, where  $n$  is the number of nodes of a subject tree.

*Proof:* Proof is constructed similarly as in Lemma 5.

Consider a tree with  $n+1$  nodes. Arity of the first node is  $n$ . Remaining nodes are labelled with the same nullary symbol. It is possible to create tree template where on each position of nullary symbol a nonlinear variable  $X, Y, \dots$ , symbol  $S$  or the original symbol can be placed. Therefore,  $m$  positions is possible to choose from  $2 + v$  symbols. Language size is then  $\mathcal{O}((2 + v)^n)$ , where  $n + 1$  is the number of nodes of a subject tree. ■

## VIII. CONCLUSION

Algorithms creating pushdown automata for nonlinear tree indexing have been presented. Since these pushdown automata are input-driven, they can be determined. It is shown that a nondeterministic nonlinear tree pattern pushdown automaton for one nonlinear variable has a space complexity polynomial to the size of the subject tree. The algorithm for constructing nondeterministic nonlinear tree pattern pushdown automaton for more nonlinear variables using the union of automata (Cartesian product of tails of automata) have also been presented.

The exact space complexity of the deterministic nonlinear indexing pushdown automata is an open question.

## REFERENCES

- [1] Alfred V. Aho and Jeffrey D. Ullman. *The theory of parsing, translation, and compiling*. Prentice-Hall Englewood Cliffs, N.J., 1972.
- [2] *Arbology* [www pages](http://www.arbology.org), Available on: <http://www.arbology.org>, June 2011.
- [3] Blumer, A., Blumer, J., Haussler, D., Ehrenfeucht, A., Chen, M. T., Seiferas, J. I., 1985. The smallest automaton recognizing the subwords of a text. *Theor. Comput. Sci.* 40, 31–55.
- [4] Crochemore, M., 1986. Transducers and repetitions. *Theor. Comput. Sci.* 45 (1), 63–86.
- [5] H. Comon, M. Dauchet, R. Gilleron, C. Löding, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. *Tree automata techniques and applications*. Available on: <http://www.grappa.univ-lille3.fr/tata>, 2007. release October, 12th 2007.
- [6] Crochemore, M., Hancart, C., 1997. Automata for matching patterns. In: Rozenberg, G., Salomaa, A. (Eds.), *Handbook of Formal Languages. Vol. 2 Linear Modeling: Background and Application*. Springer-Verlag, Berlin, Ch. 9, pp. 399–462.
- [7] Crochemore, M., Rytter, W., 1994. *Jewels of Stringology*. World Scientific, New Jersey.
- [8] Domenico Cantone, Simone Faro and Emanuele Giacquinta: A Compact Representation of Nondeterministic (Suffix) Automata for the Bit-Parallel Approach, In: *CPM 2010, LNCS 6129*, Springer, Berlin, 2010.
- [9] F Gecseg and M. Steinby. Tree languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3 Beyond Words. Handbook of Formal Languages, pages 1–68. Springer-Verlag, Berlin, 1997.
- [10] Christoph M. Hoffmann and Michael J. O'Donnell. Pattern matching in trees. *J. ACM*, 29(1):68–95, 1982.
- [11] Hopcroft, J. E., Motwani, R., Ullman, J. D., 2001. *Introduction to automata theory, languages, and computation*, 2nd Edition. Addison-Wesley, Boston.
- [12] J. W. Klop. *Term Rewriting Systems*, Handbook of Logic in Computer Science, 1992.
- [13] Janousek, J. String Suffix Automata and Subtree Pushdown Automata. In: *Proceedings of the Prague Stringology Conference 2009*, pp. 160–172, Czech Technical University in Prague, Prague, 2009.
- [14] Janousek, J.: *Arbology: Algorithms on Trees and Pushdown Automata*. Habilitation thesis, TU FIT, Brno, 2010.
- [15] Janousek, J., Melichar, B. On Regular Tree Languages and Deterministic Pushdown Automata. In *Acta Informatica*, Vol. 46, No. 7, pp. 533-547, Springer, 2009.
- [16] Melichar, B. Arbology: Trees and pushdown automata. In: *LATA 2010 (LNCS 6031)*, invited speaker, pp. 32-49, Springer, 2010.
- [17] Melichar, B., Holub, J., Polcar, J., 2005. Text searching algorithms. Available on: <http://stringology.org/athens/>, release November 2005.
- [18] R. Ramesh, I. V. Ramakrishnan. *Nonlinear Pattern Matching in Trees*, Journal of the Association for Computing Machinery, Vol 39, No 2, April 1992.
- [19] Smyth, B., 2003. *Computing Patterns in Strings*. Addison-Wesley-Pearson Education Limited, Essex, England.



# A Type and Effect System for Implementing Functional Arrays with Destructive Updates

Georgios Korfiatis  
 Email: gkorf@softlab.ntua.gr

Michalis Papakyriakou  
 Email: mpapakyr@softlab.ntua.gr

Nikolaos Papaspyrou  
 Email: nickie@softlab.ntua.gr

School of Electrical and Computer Engineering  
 National Technical University of Athens  
 Polytechniupoli, 15780 Zografou, Athens, Greece

**Abstract**—It can be argued that some of the benefits of purely functional languages are counteracted by the lack of efficient and natural-to-use data structures for these languages. Imperative programming is based on manipulating data structures destructively, e.g., updating arrays in-place; however, doing so in a purely functional language violates the language’s very nature. In this paper, we present a type and effect system for an eager purely functional language that tracks array usage, i.e., read and write operations, and enables the efficient implementation of purely functional arrays with destructive update.

## I. INTRODUCTION

ARRAYS are ubiquitous data structures in imperative programming, offering constant-time storing and retrieval of data. On the contrary, their use in the purely functional programming paradigm is not equally natural. A key property in purely functional programs is referential transparency, which guarantees that an expression has always the same value in any context. Referential transparency facilitates reasoning about program properties and also enables many compiler optimizations. Yet this entails that an operator intended to update the contents of a given array should actually yield a fresh (updated) copy of the array, leaving the original untouched, and thus adding a significant time and space complexity overhead to the program.

Naïve implementations of such an update operator would require an additional  $O(n)$  complexity for each update, both in time and space, where  $n$  is the size of the array. Better implementations could be based on building a set of differences between the original and the updated array, in the same spirit more or less as Haskell’s `DiffArray`. In principle, such implementations could be tailored to specific use patterns for arrays, (e.g., single-threaded updates); however, as Haskell’s bug reports reveal, the overall performance is very poor.

Pippenger has shown that every algorithm using strict impure data structures that runs in  $O(n)$  can be translated to an algorithm using pure data structures that runs in  $O(n \log n)$  time, simulating random access memory with appropriate algebraic data structures such as balanced binary trees [12]. He has also shown that there are algorithms for which this is the best one can do; in other words, there are algorithms for which an impure language performs asymptotically better than a pure language. This result has caused a significant

stir in the functional programming community. It has been discussed whether it is valid for lazy pure languages [2], or for algorithms without “on line” requirements.

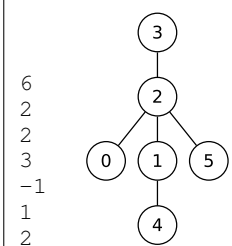
On the other hand, two practical questions have been raised on this subject:

- What good does this study of asymptotic behavior make? Even if the book-keeping cost of using purely functional data structures can be amortized throughout the program and does not increase the overall program complexity, it still induces a constant factor slowdown that may not be negligible.
- How easy is it to work with (and analyze the performance of) purely functional data structures? In other words, how easily can one reuse a long-standing imperative algorithm in a purely functional language and will it be the same algorithm after all?

Let us consider a motivating example. Suppose we are given as input a tree structure which we need to subsequently process, e.g., perform a depth-first traversal. Suppose also that the input is given in the following form: the first line contains the number of nodes  $n$  (numbered from 0 to  $n - 1$ ) and the following  $n$  lines contain the nodes’ parents. The root of the tree is given a parent of  $-1$ . The following program snippet in C++ can be used to read such a tree. For each node  $i$ , the list `c[i]` contains the node’s children. Variable `r` contains the tree’s root.

```
int n, r;
scanf("%d\n", &n);
list<int> c[n];
for (int i=0; i<n; i++) {
    int p;
    scanf("%d\n", &p);
    if (p >= 0)
        c[p].push_back(i);
    else
        r = i;
}
```

**Input**



Now consider an equivalent program snippet in a purely functional programming language. We use here OCaml syntax, but we assume purely functional arrays with the following signature. Function `upd` returns the updated array; semantically, the result of `upd a i x` can be thought of as a new array

whose contents equal those of  $a$ , with the exception of value  $x$  occupying position  $i$ .

```
type 'a array
val newArray : int -> 'a -> 'a array
val get : 'a array -> int -> 'a
val upd : 'a array -> int -> 'a -> 'a array
```

We skip the reading of data and assume that both  $n$  and the list of the nodes' parents  $l$  have already been read. Our goal is to build a data structure of type `tree`, which we can then process in a purely functional fashion.

```
type tree = T of int * tree list
```

Function `build` does precisely this. It walks down the list  $l$ , using array  $c$  to collect for each node a list of its children, exactly as the imperative program does. Argument  $r$  propagates the tree's root node. Finally, function `mkTree` builds the tree structure using the information that has been collected in  $c$ .

```
let build n l =
  let rec walk i c r = function
    | [] -> (c, r)
    | p :: ps when p >= 0 ->
      let c = upd c p (i :: get c p) in
      walk (i+1) c r ps
    | _ :: ps ->
      walk (i+1) c i ps in
  let c = newArray n [] in
  let (c, r) = walk 0 c 0 l in
  let rec mkTree u =
    T (u, map mkTree (get c u)) in
  mkTree r
```

With a naïve implementation of functional arrays, copying the array every time it is updated, function `build` will require  $O(n^2)$  time, while its imperative counterpart takes just  $O(n)$ . Even smarter implementations, e.g., using a balanced binary tree for  $c$ , will take  $O(n \log n)$ . There doesn't seem to be a natural way to bring the complexity down to  $O(n)$  without updating the array in-place. Notice that, in the program snippet above, the array  $c$  is updated and used only in a single-threaded way; after the updating takes place, the original array is never used again. In this program, all array updates can be done destructively without altering its semantics, thus obtaining an  $O(n)$  time complexity. This, however, need not be the case, and this is the real difficulty in optimizing purely functional arrays.

It is desirable to combine the best of the two worlds, imperative and purely functional, in a language that implements array operations efficiently while complying with the referential transparency principle of purely functional programming. In this paper, we present a type and effect system for a purely functional language that enables such an optimization in a safe manner. In particular, the language includes an update operator that can be implemented destructively, but is semantically equivalent to the pure `upd` function that we have just seen. In the sections that follow, we present our approach in an

$$\begin{aligned}
 e &::= w \mid \text{let } x = e_1 \text{ in } e_2 \mid w_1 w_2 \mid \text{if } w \text{ then } e_1 \text{ else } e_2 \\
 &\quad \mid \text{newArray } w_1 w_2 \mid \text{get } w_1 w_2 \mid \text{upd } w_1 w_2 w_3 \mid \dots \\
 w &::= x \mid v \\
 v &::= f \mid \ell \mid \text{true} \mid \text{false} \mid i \\
 f &::= \lambda x : \tau. e \mid \text{fix } x : \tau. f \\
 \tau_g &::= \text{Bool} \mid \text{Int} \\
 \tau &::= \tau_g \mid \text{Array } \tau_g \mid (x : \tau_1) \xrightarrow{\gamma \& \delta} \tau_2
 \end{aligned}$$

Fig. 1. Syntax.

informal way through examples (Section II), then the formalization of the type and effect system (Section III). Section IV discusses related work. We finish with some concluding remarks and directions for future work.

## II. AN INFORMAL ACCOUNT OF OUR APPROACH

In Fig. 1 we define a simple eager functional language, whose main deviation from conventional languages is that evaluation order is explicit: all operators take evaluated arguments ( $w$ ), which are values ( $v$ ) or variables that contain values ( $x$ ). More complex expressions must be explicitly “linearized”, forming essentially a sequence of let-bindings with primitive operations. Making the evaluation order explicit is not an important feature of our language, but it facilitates the presentation of our approach. Programs in a traditional eager functional language, like the last example of Section I, can be straightforwardly transformed to this linear form. For example,  $x y z$  is translated to `let  $y' = x y$  in  $y' z$` . Notice also that our language is explicitly typed, but a more-or-less standard type inference can be used to fill in most of the type annotations.

The language includes integer values ( $i$ ), boolean values and standard conditional expressions, function values ( $f$ ) which can be recursive, and location constants ( $\ell$ ) that do not appear in the original program. Although we do not show them, we assume the existence of operators for manipulating integer and boolean values. The language also includes operators for array manipulation:

- `newArray  $i v$`  creates an array of size  $i$  whose cells are all initialized with value  $v$  and returns it;
- `get  $a i$`  returns the  $i$ th element of array  $a$ ; and
- `upd  $a i v$`  returns an array equal to  $a$  except that position  $i$  is now occupied by value  $v$ .

A type ( $\tau$ ) is either a ground type ( $\tau_g$ ), which can be `Bool` or `Int`, an array type whose element is necessarily of ground type, or a function type. The non-standard annotations in function types will be explained shortly. We assume that arrays are 0-indexed and we do not care to use the type system to avoid array out-of-bounds errors.

The main idea is to maintain an effect for each expression, which records the uses of variables that represent arrays within this expression. Let's start by assuming that an effect consists only of a *qualifier assignment* ( $\xi$ ), which assigns a qualifier  $q$  to each of the array variables used in an expression. (In a short while we will extend our notion of effect.) A variable is

assigned qualifier  $\mathbf{W}$  (written) if it has been used in an **upd** operation, or qualifier  $\mathbf{R}$  (read) if it has been used in a **get** operation.

For example, assuming that  $a$  is of array type, consider the following expression:

$$\begin{array}{l} \text{let } v = \text{get } a \ 0 \ \text{in} \\ \quad \text{upd } a \ 1 \ v \end{array} \quad \begin{array}{l} \{a:\mathbf{R}\} \\ \{a:\mathbf{W}\} \end{array}$$

Each subexpression produces the qualifier assignment shown to its right. The expression gets its overall effect by combining the effects of its subexpressions in the order of execution. In this example it yields the assignment  $\{a:\mathbf{R}, a:\mathbf{W}\}$ .

It is crucial that an array be not used, either for writing or for reading, after it has been written. Therefore, computing the overall effect of the following expression should fail.

$$\begin{array}{l} \text{let } a' = \text{upd } a \ 1 \ 0 \ \text{in} \\ \quad \text{get } a \ 0 \end{array} \quad \begin{array}{l} \{a:\mathbf{W}\} \\ \{a:\mathbf{R}\} \end{array}$$

However, due to the possibility of aliasing, it is not guaranteed that two variables in the effect of an expression refer to distinct arrays. It is thus necessary to keep track of the order in which the effects appear when combining two (seemingly unconnected) effects. In particular, we maintain a *constraint set* ( $\kappa$ ) which records every pair of variables  $x < y$  such that  $x:\mathbf{W}$  has appeared before  $y:q$ , for any qualifier  $q$ . In order to keep track of aliased variables, we additionally annotate each expression with a *propagation set* ( $\delta$ ) which records variables of array type that may be the result of evaluating this expression.

To summarize all this, we relate each expression with a tuple  $\gamma \& \delta$ , where the *effect*  $\gamma$  is itself a tuple  $\langle \xi, \kappa \rangle$ , consisting of an assignment of qualifiers to variables and a set of constraints. Apart from the standard typing environment, we also keep an aliasing environment  $A$  mapping variables to all their possible aliases; this environment is updated in every **let** binding, using the propagation set  $\delta$  of the bound expression.

Consider the following example, which is equivalent to the previous one and should therefore also be rejected:

$$\begin{array}{l} \text{let } b = a \ \text{in} \\ \text{let } a' = \text{upd } a \ 1 \ 0 \ \text{in} \\ \quad \text{get } b \ 0 \end{array} \quad \begin{array}{l} \langle \emptyset, \emptyset \rangle \& \{a\} \\ \langle \{a:\mathbf{W}\}, \emptyset \rangle \& \emptyset \\ \langle \{b:\mathbf{R}\}, \emptyset \rangle \& \emptyset \end{array}$$

The first **let** binding leads to nothing more than the aliasing  $A(b) = \{a\}$ . The subsequent operators **upd** and **get** result in the qualifier assignments for  $a$  and  $b$ , respectively. What is important, however, is the combination of all these effects in the result of this expression. The overall effect is the qualifier assignment  $\{a:\mathbf{W}, b:\mathbf{R}\}$ , along with the constraint set  $\{a < b\}$ , which is produced because  $a:\mathbf{W}$  is executed before  $b:\mathbf{R}$ . When getting out of its scope,  $b$  is replaced by its propagation set  $\{a\}$  both in the qualifier assignment and in the constraint set. This leads to the constraint set  $\{a < a\}$ , which is unsatisfiable, and the program is rejected.

Lambda abstractions have no effects themselves. Nevertheless, they need to remember the possible effects of their body. This is done by annotating the function type with the effect

and the propagation set. When applying a function, the effect on the arrow should be checked for compatibility with respect to the effects collected so far. Thus typing the application  $f \ 4$  in the following example will fail, because  $a$  has been updated before being read by the function.

$$\begin{array}{l} \text{let } f = \lambda x : \text{Int. get } a \ x \ \text{in} \\ \quad \text{let } a' = \text{upd } a \ 0 \ 0 \ \text{in} \\ \quad \quad f \ 4 \end{array} \quad f : \text{Int} \xrightarrow{\langle \{a:\mathbf{R}\}, \emptyset \rangle \& \emptyset} \text{Int} \quad \begin{array}{l} \langle \{a:\mathbf{W}\}, \emptyset \rangle \& \emptyset \\ \langle \{a:\mathbf{R}\}, \emptyset \rangle \& \emptyset \end{array}$$

Annotations on arrow types can also contain references to the formal parameters of the function. To this end, we name the type of each formal parameter, as in the following example. (Of course, only annotations for array types are of any use; we thus omit naming parameters of a different type in the following examples.) When applying a function, the formal parameters mentioned in its effect are substituted with the variable (and its aliases) that correspond to the actual parameter. Typechecking the following example will fail, because application  $f \ r$  updates  $r$  before this is read by the last **get** operation.

$$\begin{array}{l} \text{let } f = \lambda a : \text{Array Int. upd } a \ 0 \ 0 \ \text{in} \\ \quad \quad f : (a : \text{Array Int}) \xrightarrow{\langle \{a:\mathbf{W}\}, \emptyset \rangle \& \emptyset} \text{Array Int} \\ \text{let } r' = f \ r \ \text{in} \\ \quad \quad \text{get } r \ 0 \end{array} \quad \begin{array}{l} \langle \{r:\mathbf{W}\}, \emptyset \rangle \& \emptyset \\ \langle \{r:\mathbf{R}\}, \emptyset \rangle \& \emptyset \end{array}$$

The following example demonstrates how aliasing can be detected in our system. Function  $f$ , which takes two array parameters  $x$  and  $y$ , is applied to the same array  $r$ .

$$\begin{array}{l} \text{let } r = \text{newArray } 5 \ 0 \ \text{in} \\ \text{let } f = \\ \quad \lambda x : \text{Array Int.} \\ \quad \lambda y : \text{Array Int.} \\ \quad \quad \text{let } a = \text{upd } x \ 3 \ 4 \ \text{in} \\ \quad \quad \quad \text{let } b = \text{get } y \ 3 \ \text{in} \\ \quad \quad \quad \quad \text{upd } a \ 2 \ b \\ \quad \text{in} \\ \text{let } f' = f \ r \ \text{in} \\ \quad \quad f' \ r \end{array} \quad \begin{array}{l} \langle \{x:\mathbf{W}\}, \emptyset \rangle \& \emptyset \\ \langle \{y:\mathbf{R}\}, \emptyset \rangle \& \emptyset \\ \langle \{a:\mathbf{W}\}, \emptyset \rangle \& \emptyset \end{array}$$

Function  $f$  has the following type:

$$\begin{array}{l} (x : \text{Array Int}) \xrightarrow{\emptyset \& \emptyset} \\ (y : \text{Array Int}) \xrightarrow{\langle \{x:\mathbf{W}, y:\mathbf{R}\}, \{x < y\} \rangle \& \emptyset} \text{Array Int} \end{array}$$

Notice that the first arrow carries no effect, since it only yields a closure. Application  $f \ r$  has type:

$$(y : \text{Array Int}) \xrightarrow{\langle \{r:\mathbf{W}, y:\mathbf{R}\}, \{r < y\} \rangle \& \emptyset} \text{Array Int}$$

and the last application  $f' \ r$  produces the effect

$$\langle \{r:\mathbf{W}, r:\mathbf{R}\}, \{r < r\} \rangle$$

which contains an unsatisfiable constraint.

As a bigger example, we adapt the core of the motivating example from Section I. Function *walk* collects in array  $c$  the list of children of each node, taking as input the list of nodes' parents ( $l$ ). We assume a ground type **List Int** of lists of integers but, as our language does not support tuples, we skip

the calculation of the tree's root, which can easily be done by a separate function. We also assume the usual list operators: **nil** and **cons**, **isnil**, **head**, and **tail**.

```

let walk =
  fix f : Int  $\xrightarrow{\emptyset \& \emptyset}$  (c : Array (List Int))  $\xrightarrow{\emptyset \& \emptyset}$ 
    List Int  $\xrightarrow{\langle\{c:\mathbf{W},c:\mathbf{R}\},\emptyset\rangle \& \emptyset}$  Array (List Int).
  λi : Int.
  λc : Array (List Int).
  λl : List Int.
  let b = isnil l in
  if b then
    c
  else
    let p = head l in
    let ps = tail l in
    let c' =
      let b = (p >= 0) in
      if b then
        let v = get cp in
        let v' = cons i v in
        upd cp v'
      else
        c in
    let i' = i + 1 in
    let f1 = f i' in
    let f2 = f1 c' in
    f2 ps

```

### III. FORMALIZATION

In this section we formally present the operational semantics of our language and the type and effect system. The syntax of the language has already been presented in Fig. 1.

#### A. Operational semantics

In Fig. 2, we define a standard eager operational semantics of expressions with respect to a memory  $\mu$ , mapping locations  $\ell$  to memory blocks. A memory block  $\{\overline{v}_i^{i < n}\}$  has size  $n$  and contains the values  $v_0, \dots, v_{n-1}$ . The linearized nature of the language ensures that there is only one propagation rule in the semantics (for **let**), with every other rule corresponding to a particular sort of redex. Note that **upd**  $\ell j v$  updates the memory block in-place and returns the original location.

#### B. Type system

Effects  $\gamma$  are defined in Fig. 3 as a pair  $\langle \xi, \kappa \rangle$ , where  $\xi$  is an assignment of qualifiers  $q$  to variables and  $\kappa$  a set of constraints. A propagation set  $\delta$  is simply a set of variables, an aliasing environment  $A$  maps variables to sets of variables, and a typing environment  $\Gamma$  maps variables to types  $\tau$ .

We have opted to presented the type system in a form of input and output effects. This allows us to determine a type error at the exact program point that is to blame for the constraint violation. The typing judgment for an expression  $e$

$$\Gamma; A; \gamma_0 \vdash e : \tau \& \gamma \& \delta$$

$$\begin{array}{c}
\boxed{\mu; e \longrightarrow \mu'; e'} \\
\hline
\mu; e_1 \longrightarrow \mu'; e'_1 \\
\hline
\mu; \text{let } x = e_1 \text{ in } e_2 \longrightarrow \mu'; \text{let } x = e'_1 \text{ in } e_2 \\
\hline
\mu; \text{let } x = v \text{ in } e \longrightarrow \mu; e[v/x] \\
\hline
\mu; (\lambda x : \tau. e) v \longrightarrow \mu; e[v/x] \\
\hline
\mu; \text{if true then } e_1 \text{ else } e_2 \longrightarrow \mu; e_1 \\
\hline
\mu; \text{if false then } e_1 \text{ else } e_2 \longrightarrow \mu; e_2 \\
\hline
\mu; (\text{fix } x : \tau. f) v \longrightarrow \mu; f[(\text{fix } x : \tau. f)/x] v \\
\hline
\ell \text{ fresh in } \mu \\
\forall j. (0 \leq j < i \Rightarrow v_j = v) \\
\hline
\mu; \text{newArray } i v \longrightarrow \mu, \ell \mapsto \{\overline{v}_j^{j < i}\}; \ell \\
\hline
0 \leq j < n \\
\mu, \ell \mapsto \{\overline{v}_i^{i < n}\}; \text{get } \ell j \longrightarrow \mu, \ell \mapsto \{\overline{v}_i^{i < n}\}; v_j \\
\hline
0 \leq j < n \\
\forall j. (0 \leq i < n \wedge i \neq j \Rightarrow v'_i = v_i) \\
v'_j = v \\
\hline
\mu, \ell \mapsto \{\overline{v}_i^{i < n}\}; \text{upd } \ell j v \longrightarrow \mu, \ell \mapsto \{\overline{v}_i^{i < n}\}; \ell
\end{array}$$

Fig. 2. Operational semantics.

$$\begin{array}{l}
q ::= \mathbf{W} \mid \mathbf{R} \\
\xi ::= \emptyset \mid \xi, x : q \\
\kappa ::= \emptyset \mid \kappa, x < y \\
\gamma ::= \langle \xi, \kappa \rangle \\
\delta ::= \emptyset \mid \delta, x \\
A ::= \emptyset \mid A, x \mapsto \delta \\
\Gamma ::= \emptyset \mid \Gamma, x : \tau
\end{array}$$

Fig. 3. Qualifiers, effects, and environments.

yields a type  $\tau$ , effect  $\gamma$ , and propagation set  $\delta$ , given type environment  $\Gamma$ , aliasing environment  $A$ , and input effect  $\gamma_0$ . The output effect  $\gamma$  results from a combination of the input effect and the actual effect of the expression in question. The typing judgment for evaluated arguments ( $w$ ) is similar but lacks the output effect; it will be presented later.

Consider first the rule for **get**:

$$\begin{array}{c}
\Gamma; A; \gamma_0 \vdash w_1 : \mathbf{Array} \tau_g \& \delta_1 \\
\Gamma; A; \gamma_0 \vdash w_2 : \mathbf{Int} \& \emptyset \\
\gamma = \gamma_0 \times \delta_1 : \mathbf{R} \\
\hline
\Gamma; A; \gamma_0 \vdash \text{get } w_1 w_2 : \tau_g \& \gamma \& \emptyset
\end{array}$$

Arguments  $w_1$  and  $w_2$  are first checked and  $w_1$  corresponds to a propagation set  $\delta_1$ . In fact, since locations do not appear in the original program that we typecheck,  $w_1$  can only be a

variable, say  $x$ , thus  $\delta_1$  must be the singleton  $\{x\}$ , as it will become clear in the rule for variables. The actual effect of the `get` expression is  $\langle\{x : \mathbf{R}\}, \emptyset\rangle$ , since variable  $x$  is used for reading. This is expressed concisely, generalized for any  $\delta$ , using the following notation:

$$\delta : q \equiv \langle\{x : q \mid x \in \delta\}, \emptyset\rangle$$

Input and actual effects are combined according to the following function:

$$\begin{aligned} \gamma_1 \times \gamma_2 &\equiv \langle\xi_1, \kappa_1\rangle \times \langle\xi_2, \kappa_2\rangle \equiv \\ &\langle\xi_1 \cup \xi_2, (\kappa_1 \cup \kappa_2) \cup \{x < y \mid (x : \mathbf{W}) \in \xi_1 \wedge (y : q) \in \xi_2\}\rangle \end{aligned}$$

which takes the union of the qualifier assignments ( $\xi_1$  and  $\xi_2$ ) and the union of constraints ( $\kappa_1$  and  $\kappa_2$ ) plus a new set of constraints which results from the combination of every *written* variable in  $\gamma_1$  with every variable in  $\gamma_2$ .

The rule for `upd` is similar; here the actual effect has a  $\mathbf{W}$  assignment instead of  $\mathbf{R}$ .

$$\begin{aligned} \Gamma; A; \gamma_0 \vdash w_1 : \mathbf{Array} \tau_g \&\ \delta_1 \\ \Gamma; A; \gamma_0 \vdash w_2 : \mathbf{Int} \&\ \emptyset \\ \Gamma; A; \gamma_0 \vdash w_3 : \tau_g \&\ \emptyset \\ \gamma &= \gamma_0 \times \delta_1 : \mathbf{W} \end{aligned}$$

$$\frac{}{\Gamma; A; \gamma_0 \vdash \mathbf{upd} \ w_1 \ w_2 \ w_3 : \mathbf{Array} \ \tau_g \ \&\ \gamma \ \&\ \emptyset}$$

In both rules, we have omitted array index out-of-bounds checks. This is orthogonal to our approach and we would not like to diverge from the issue of destructive implementation of arrays to discuss it in this paper.

Creating a new array produces no effect; therefore the rule for `newArray` simply propagates the input effect after checking its arguments.

$$\begin{aligned} \Gamma; A; \gamma_0 \vdash w_1 : \mathbf{Int} \&\ \emptyset \\ \Gamma; A; \gamma_0 \vdash w_2 : \tau_g \&\ \emptyset \end{aligned}$$

$$\frac{}{\Gamma; A; \gamma_0 \vdash \mathbf{newArray} \ w_1 \ w_2 : \mathbf{Array} \ \tau_g \ \&\ \gamma_0 \ \&\ \emptyset}$$

In order to typecheck a `let` expression, we first check the bound expression  $e_1$  given input effect  $\gamma_0$ .

$$\begin{aligned} \Gamma; A; \gamma_0 \vdash e_1 : \tau_1 \&\ \gamma_1 \ \&\ \delta_1 \\ \Gamma, x : \tau_1; A \oplus x \mapsto \delta_1; \gamma_1 \vdash e_2 : \tau_2 \&\ \gamma_2 \ \&\ \delta_2 \\ \tau' &= \tau_2[\delta_1/x] \\ \gamma' &= \gamma_2[\delta_1/x] \\ \delta' &= \delta_2[\delta_1/x] \end{aligned}$$

$$\frac{}{\Gamma; A; \gamma_0 \vdash \mathbf{let} \ x = e_1 \ \mathbf{in} \ e_2 : \tau' \ \&\ \gamma' \ \&\ \delta'}$$

This produces type  $\tau_1$ , effect  $\gamma_1$  and propagation set  $\delta_1$ . The body  $e_2$  is checked on input effect  $\gamma_1$ , since this represents the combination of  $\gamma_0$  with the actual effect of  $e_1$ , that is, all the effects that occurred prior to the execution of  $e_2$ . Apart from adding the binding variable in the type environment, we also add the aliasing information for  $e_1$  in the map  $A$ . Binding variable  $x$  maps to all variables contained in the propagation set  $\delta_1$  and to all their aliases, transitively, as defined below:

$$A \oplus x \mapsto \delta \equiv A, x \mapsto \delta \cup \bigcup \{A(y) \mid y \in \delta\}$$

The resulting type, effect, and propagation set of the `let` expression are those computed for its body. However, these may contain occurrences of  $x$ , which must be replaced by its aliases  $\delta_1$ . The required substitutions are defined below.

$$\delta'[\delta/x] = \begin{cases} \delta' - \{x\} \cup \delta & \text{if } x \in \delta \\ \delta' & \text{otherwise} \end{cases}$$

$$\gamma[\delta/x] = \langle\xi[\delta/x], \kappa[\delta/x]\rangle$$

$$\begin{aligned} \xi[\delta/x] &= \{y : q \mid (y : q) \in \xi \wedge y \neq x\} \\ &\cup \bigcup \{\delta : q \mid (x : q) \in \xi\} \end{aligned}$$

$$\begin{aligned} \kappa[\delta/x] &= \{y < z \mid (y < z) \in \kappa \wedge y \neq x \wedge z \neq x\} \\ &\cup \{w < z \mid (y < z) \in \kappa \wedge y = x \wedge w \in \delta\} \\ &\cup \{y < w \mid (y < z) \in \kappa \wedge z = x \wedge w \in \delta\} \end{aligned}$$

Substituting  $\delta$  for  $x$  in  $\gamma$  amounts to removing any assignment  $x : q$  and any constraint containing  $x$ , and adding the respective assignments and constraints for all variables in  $\delta$ . Substitution of  $\delta$  in a type ( $\tau[\delta/x]$ ) walks through the arrows recursively and applies the substitution for  $\gamma$  and  $\delta$ .

The rule for application is the most elaborate one.

$$\begin{aligned} \Gamma; A; \gamma_0 \vdash w_1 : (x : \tau) \xrightarrow{\gamma \& \delta} \tau' \ \&\ \emptyset \\ \Gamma; A; \gamma_0 \vdash w_2 : \tau \ \&\ \delta_2 \\ A \vdash_{wf} \tau'[\delta_2/x] \\ A \vdash_{wf} \gamma[\delta_2/x] \\ \gamma' = \gamma_0 \times \gamma[\delta_2/x] \\ A \vdash_{wf} \gamma' \\ A; \gamma' \vdash_{compat} \delta[\delta_2/x] \end{aligned}$$

$$\frac{}{\Gamma; A; \gamma_0 \vdash w_1 \ w_2 : \tau'[\delta_2/x] \ \&\ \gamma' \ \&\ \delta[\delta_2/x]}$$

We first typecheck  $w_1$  and  $w_2$ . The effect of the application is the effect  $\gamma$  that the function carries on its type. As explained in the related example in Section II, any occurrence of the formal parameter  $x$  need to be replaced by the aliases of the actual parameter,  $\delta_2$ . This substitution may render some constraints inconsistent; thus the resulting effect needs to be checked for well-formedness. Judgment

$$A \vdash_{wf} \gamma$$

checks whether  $\gamma$  is well-formed, i.e., it contains no constraint of the form  $x < y$  where  $x$  and  $y$  are aliases (or the same variable). Similarly, the effect  $\gamma'$  that results from combining the input effect with the computed one need to be checked, since it contains fresh constraints. Likewise, the output propagation set is the result of substituting  $\delta_2$  for the formal parameter  $x$  in the propagation set of the function,  $\delta$ . Given the output effect  $\gamma'$ , we have to make sure that the resulting propagation set does not include any variable that has already been used for writing. For this purpose we employ the following judgment:

$$A; \gamma \vdash_{compat} \delta$$

which checks whether  $\delta$  is compatible with  $\gamma$  given  $A$ , i.e., it does not contain any variable such that either itself or one of its aliases is assigned qualifier  $\mathbf{W}$  in  $\gamma$ .

Finally, the resulting type  $\tau'$  can itself be an arrow type. Since it can contain occurrences of the formal parameter  $x$ , these have to be replaced by  $\delta_2$ , too. This substitution may break some constraints; we have thus to check the well-formedness of the type. Judgment

$$A \vdash_{wf} \tau$$

checks the well-formedness of a type with respect to aliasing environment  $A$  by checking the well-formedness of all possible  $\gamma$ 's contained in the type as well as the compatibility of all  $\delta$ 's with their respective  $\gamma$ .

The rule for **if** typechecks the two alternatives  $e_1$  and  $e_2$  given the same input effect  $\gamma_0$ .

$$\frac{\begin{array}{l} \Gamma; A; \gamma_0 \vdash w : \mathbf{Bool} \& \emptyset \\ \Gamma; A; \gamma_0 \vdash e_1 : \tau \& \gamma_1 \& \delta_1 \\ \Gamma; A; \gamma_0 \vdash e_2 : \tau \& \gamma_2 \& \delta_2 \\ \gamma = \gamma_1 \cup \gamma_2 \\ \delta = (\delta_1 - \mathbf{W}_A(\gamma_2)) \cup (\delta_2 - \mathbf{W}_A(\gamma_1)) \end{array}}{\Gamma; A; \gamma_0 \vdash \mathbf{if} \ w \ \mathbf{then} \ e_1 \ \mathbf{else} \ e_2 : \tau \& \gamma \& \delta}$$

The output effect  $\gamma$  is the union of the two alternative effects, since we need to conservatively assume that either of the two effects has actually happened. Similarly, the output propagation set  $\delta$  results from the union of the alternative propagation sets. However, extra care is needed here. We need to make sure that  $\delta$  be compatible with  $\gamma$ . We thus subtract all written variables (and their aliases) of  $\gamma_1$  (notation  $\mathbf{W}_A(\gamma_1)$ ) from  $\delta_2$ , and those of  $\gamma_2$  from  $\delta_1$  respectively. This ensures that if a variable is assigned qualifier **W** in either effect, then it cannot be an alias for the **if** expression.

Finally, an evaluated argument  $w$  is considered an expression, too. The following rule simply uses the respective judgment and propagates the input effect.

$$\frac{\Gamma; A; \gamma_0 \vdash w : \tau \& \delta}{\Gamma; A; \gamma_0 \vdash w : \tau \& \gamma_0 \& \delta}$$

Evaluated arguments (values and variables) are typechecked with the following judgment.

$$\Gamma; A; \gamma_0 \vdash w : \tau \& \delta$$

This differs from the judgment for expressions in that there is no output effect. On the other hand, rules expect an input effect  $\gamma_0$ , but only for reasons of checking compatibility, as it will be explained shortly.

The first rule concerns variables of an array type.

$$\frac{\begin{array}{l} (x : \mathbf{Array} \ \tau_g) \in \Gamma \\ A; \gamma_0 \vdash_{\text{compat}} \{x\} \end{array}}{\Gamma; A; \gamma_0 \vdash x : \mathbf{Array} \ \tau_g \& \{x\}}$$

Using the appropriate compatibility judgment, we check that such a variable has not been previously used for writing. The main duty of this rule is to propagate variable  $x$  as an alias.

Likewise, considering a variable of type other than an array, possibly of arrow type:

$$\frac{\begin{array}{l} (x : \tau) \in \Gamma \\ \mathbf{isNotArray} \ \tau \\ A; \gamma_0 \vdash_{\text{compat}} \tau \end{array}}{\Gamma; A; \gamma_0 \vdash x : \tau \& \emptyset}$$

or a lambda abstraction:

$$\frac{\begin{array}{l} \Gamma, x : \tau; A; \emptyset \vdash e : \tau' \& \gamma \& \delta \\ A; \gamma_0 \vdash_{\text{compat}} (x : \tau) \xrightarrow{\gamma \& \delta} \tau' \end{array}}{\Gamma; A; \gamma_0 \vdash \lambda x : \tau. e : (x : \tau) \xrightarrow{\gamma \& \delta} \tau' \& \emptyset}$$

we have to check that the  $\gamma$  and  $\delta$  on the arrow types are compatible with the input effect, in order to rule out closures containing variables that have been previously used for writing.

The rule for **fix** simply checks that the body of **fix**,  $f$ , has the same arrow type as its annotated binder  $x$ .

$$\frac{\begin{array}{l} \tau = (y : \tau_1) \xrightarrow{\gamma \& \delta} \tau_2 \\ \Gamma, x : \tau; A; \gamma_0 \vdash f : \tau \& \emptyset \end{array}}{\Gamma; A; \gamma_0 \vdash \mathbf{fix} \ x : \tau. f : \tau \& \emptyset}$$

Finally, the typing for boolean and numerical constants is standard. Fig. 4 presents the compatibility and well-formedness judgments mentioned in this section.

#### IV. RELATED WORK

Several type-based solutions to the destructive update problem for functional languages have been proposed. In some of them, as in our work, the language uses “functional” arrays and the goal is to identify updates that can be done destructively, to enable compiler optimizations. The lazy purely functional language Clean [13] employs uniqueness typing in order to add side effects without sacrificing referential transparency [1]; a simplified version of uniqueness typing has also been presented [3]. Clean’s type system works by assigning types containing uniqueness information to expressions. Typechecking takes into account annotations assigned to variables by a sharing analysis, which indicates whether a variable has been used only once within its scope or more than once. Related to uniqueness types is also the work of Harrington [8], and Hage *et al.* [7], [10]. The use of uniqueness types instead of monadic arrays in languages like Haskell is also advocated by Diviánszky [5].

Type-based approaches for functional arrays that are more directly related to the classic theory of linearity have also been proposed. Guzmán and Hudak [6] present a type and effect system similar to ours for a non-strict language, which is based on calculating “liabilities”, i.e., mutability, shareness, and linearity attributes for each variable. Their system also cannot handle “unnamed” structures (such as the ones created by ML’s `ref` constructor) and for this reason they apply several syntactic restrictions (which we avoid by using a linearized language). They assume some strictness analysis and disallow higher-order arguments to strict function application. Contrary to our work, arrays of functional values are allowed. Also, marginally related to our work is Tov and Pucella’s programmer-friendly capability-based type system with affine types [17].

In another family of strongly typed functional languages, arrays are used in the “imperative” style. Then, in languages derived from ML, some of the language’s purity is sacrificed for improved efficiency and the researchers’ goal is to minimize the cost of impurity [11], [9], [16]. An interesting solution is the use of monadic computations in Haskell [18], which save purity and referential transparency by defining a “language inside the language” for performing impure operations.

<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A; \gamma \vdash_{compat} \delta</math></div> $\frac{}{A; \gamma \vdash_{compat} \emptyset} \quad \frac{x \notin \mathbf{W}_A(\gamma) \quad A; \gamma \vdash_{compat} \delta}{A; \gamma \vdash_{compat} \delta, x}$ <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A; \gamma \vdash_{compat} \tau</math></div> $\frac{}{A; \gamma \vdash_{compat} \tau_g}$ $\frac{}{A; \gamma \vdash_{compat} \mathbf{Array} \tau_g}$ $\frac{A; \gamma \vdash_{compat} \tau \quad A; \gamma \vdash_{compat} \tau' \quad A; \gamma \vdash_{compat} \delta' \quad A; \gamma \vdash_{compat} \gamma'}{A; \gamma \vdash_{compat} (x : \tau) \xrightarrow{\gamma' \& \delta'} \tau'}$ <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A; \gamma \vdash_{compat} \gamma'</math></div> $\frac{A; \gamma \vdash_{compat} \{x \mid \exists q : (x : q) \in \xi\}}{A; \gamma \vdash_{compat} \langle \xi, \kappa \rangle}$	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A \vdash_{wf} \tau</math></div> $\frac{}{A \vdash_{wf} \tau_g}$ $\frac{}{A \vdash_{wf} \mathbf{Array} \tau_g}$ $\frac{A \vdash_{wf} \tau \quad A \vdash_{wf} \tau' \quad A \vdash_{wf} \gamma \quad A; \gamma \vdash_{compat} \delta}{A \vdash_{wf} ((x : \tau) \xrightarrow{\gamma \& \delta} \tau')}$ <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A \vdash_{wf} \gamma</math></div> $\frac{A \vdash_{wf} \kappa}{A \vdash_{wf} \langle \xi, \kappa \rangle}$ <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A \vdash_{wf} \kappa</math></div> $\frac{\forall x. \forall y. (x < y) \in \kappa \Rightarrow \neg(A \vdash \mathbf{areAlias} x y)}{A \vdash_{wf} \kappa}$ <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;"><math>A \vdash \mathbf{areAlias} x y</math></div> $\frac{(A(x_1), x_1) \cap (A(x_2), x_2) \neq \emptyset}{A \vdash \mathbf{areAlias} x_1 x_2}$
---	--

$\mathbf{aliases}_A x \equiv \{z \mid y \in (A(x), x) \wedge (A \vdash \mathbf{areAlias} y z)\}$   
 $\mathbf{W}_A(\langle \xi, \kappa \rangle) \equiv \bigcup \{\mathbf{aliases}_A x \mid (x : \mathbf{W}) \in \xi\}$

Fig. 4. Compatibility and well-formedness judgments.

On the other hand, several proposed solutions to the same problem are based not on a type system but on static analysis. Sastry *et al.* [14] present a static analysis approach for first-order linearized strict languages with flat arrays, based on abstract interpretation. Liveness analysis determines which updates can be done destructively, using reference counts for abstract locations. Based on this work, Wand and Clinger [19] present a compiler optimization for destructive array updates, focusing on a first-order strict functional language with flat arrays; their approach is based on interprocedural aliasing and liveness analysis, using set constraints. Dimoulas and Wand extend this to an untyped higher-order language [4]. A similar approach is followed by Shankar [15].

## V. CONCLUDING REMARKS

We have presented a type and effect system for a purely functional linearized language that implements array updates destructively. We expect this type system to play an important role in the optimizer of a purely functional language with arrays, deciding whether an array update can be performed destructively. In case it cannot, the update will have to be implemented in a slower way. We have developed a prototypical compiler and type checker for this language.<sup>1</sup> As future

work, we plan to extend this prototype implementation to a fuller functional language, such as ML and to eliminate the existing restrictions (e.g., the lack of support for aggregate data structures, such as tuples, and for arrays of non-ground types). Moving to a full programming language will also require a type inference algorithm that is aware of our effects, as well as effect polymorphism. A different line of research is to investigate how this technique works with lazy functional languages, such as Haskell. Also, we are working on a formal proof of type safety.

## REFERENCES

- [1] E. Barendsen and S. Smetsers, “Uniqueness typing for functional languages with graph rewriting semantics,” *Mathematical Structures in Computer Science*, vol. 6, pp. 579–612, 1996.
- [2] R. Bird, G. Jones, and O. De Moor, “More haste, less speed: lazy versus eager evaluation,” *Journal of Functional Programming*, vol. 7, pp. 541–547, Sep. 1997.
- [3] E. de Vries, R. Plasmeijer, and D. M. Abrahamson, “Uniqueness typing simplified,” in *Implementation and Application of Functional Languages*, O. Chitil, Z. Horváth, and V. Zsóka, Eds. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 201–218.
- [4] C. Dimoulas and M. Wand, “The higher-order aggregate update problem,” in *Proceedings of the 10th International Conference on Verification, Model Checking, and Abstract Interpretation*. Springer, 2009, pp. 44–58.
- [5] P. Diviánszky, “Non-monadic models of mutable references,” *Central European Functional Programming School*, pp. 146–182, 2010.

<sup>1</sup>Available from <http://www.softlab.ntua.gr/~gkorfi/src/puredest.tar.gz>.



- [6] J. Guzmán and P. Hudak, "Single-threaded polymorphic lambda calculus," in *Proceedings of the 5th Annual IEEE Symposium on Logic in Computer Science*, 1990, pp. 333–343.
- [7] J. Hage, S. Holdermans, and A. Middelkoop, "A generic usage analysis with subeffect qualifiers," in *Proceedings of the 12th ACM SIGPLAN International Conference on Functional Programming*, 2007, pp. 235–246.
- [8] D. Harrington, "Uniqueness logic," *Theoretical Computer Science*, vol. 354, no. 1, pp. 24–41, 2006.
- [9] B. Lippmeier, "Type inference and optimisation for an impure world," Ph.D. dissertation, Australian National University, 2009.
- [10] A. Middelkoop, "Improved uniqueness typing for Haskell," Master's Thesis, INF/SCR-06-08, Universiteit Utrecht, 2006.
- [11] M. Odersky, "How to make destructive updates less destructive," in *Proceedings of the 18th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, 1991, pp. 25–36.
- [12] N. Pippenger, "Pure versus impure Lisp," *ACM Transactions on Programming Languages and Systems*, vol. 19, pp. 223–238, Mar. 1997.
- [13] R. Plasmeijer and M. van Eekelen, "Clean language report, version 2.1," Department of Software Technology, University of Nijmegen, 2002.
- [14] A. Sastry, W. Clinger, and Z. Ariola, "Order-of-evaluation analysis for destructive updates in strict functional languages with flat aggregates," in *Proceedings of the Conference on Functional Programming Languages and Computer Architecture*, 1993, pp. 266–275.
- [15] N. Shankar, "Static analysis for safe destructive updates in a functional language," in *Selected papers from the 11th International Workshop on Logic Based Program Synthesis and Transformation*. London, UK: Springer-Verlag, 2001, pp. 1–24.
- [16] T. Terauchi and A. Aiken, "Witnessing side-effects," in *Proceedings of the 10th ACM SIGPLAN International Conference on Functional Programming*, 2005, pp. 105–115.
- [17] J. A. Tov and R. Pucella, "Practical affine types," in *Proceedings of the 38th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, 2011, pp. 447–458.
- [18] P. Wadler, "The essence of functional programming," in *Proceedings of the 19th Annual Symposium on Principles of Programming Languages*, Jan. 1992, pp. 1–14.
- [19] M. Wand and W. Clinger, "Set constraints for destructive array update optimization," *Journal of Functional Programming*, vol. 11, no. 3, pp. 319–346, 2001.

# Checking the Conformance of Grammar Refinements with respect to Initial Context-Free Grammars

Bryan Temprado-Battad, Antonio Sarasa-Cabezuelo, José-Luis Sierra  
 Facultad de Informática. Universidad Complutense de Madrid. 28040 Madrid, Spain  
 Email: {bryan, asarasa, jlsierra}@fdi.ucm.es

**Abstract**—According to this paper, to refine an *initial* context-free grammar supposes to devise an equivalent grammar that preserves the main syntactic structures of the initial one while making explicit other structural characteristics (e.g., associativity and priority of the operators in an expression language). Although, generally speaking, checking the equivalence of two context-free grammars is an undecidable problem, in the scenario of grammar refinement it is possible to exploit the relationships between the initial grammar and the grammar refinement to run a heuristic conformance test. These relationships must be made explicit by associating *core* non-terminal symbols in the initial grammar with *core* non-terminal symbols in the grammar refinement. Once it is made, it is possible to base the heuristic test on searching regular expressions involving both terminal and *core* non-terminal symbols that describe each *core* non-terminal symbol, and on checking the equivalence of carefully chosen pairs of such regular expressions. The paper describes the method and illustrates it with an example.

## I. INTRODUCTION

FROM a language engineering perspective, the necessity to ensure the correctness of the successive refinements of a context-free grammar becomes apparent. Unfortunately, in the last term checking this correctness supposes to check the weak equivalence of two arbitrary context-free grammars<sup>1</sup>, a classical undecidable problem [3]<sup>2</sup>. As a consequence, it is not possible to devise a *complete* checking algorithm, which can work in all the cases. However, since we are not facing arbitrary context-free grammars, but grammars related by a refinement relation, we still can propose some heuristic methods working reasonably well in many practical situations. This paper describes one of such methods.

## II. THE REFINEMENT WORK-FLOW

In order to systematize the process of context-free grammars refinement we propose the work-flow of Fig. 1.

<sup>1</sup>i.e., to check whether the two grammars generate the same language.

<sup>2</sup>Although structural equivalence of context-free grammars (i.e., to check whether two context-free grammars produces structurally equivalent parse trees) is decidable [4], grammar refinements usually do not preserve the structure of the parse trees. Thus, checking this restricted form of structural equivalence is not sufficient in this context.

The work-flow begins with the *providing the initial grammar* activity. The goal of this activity is to get an initial or base grammar to refine.

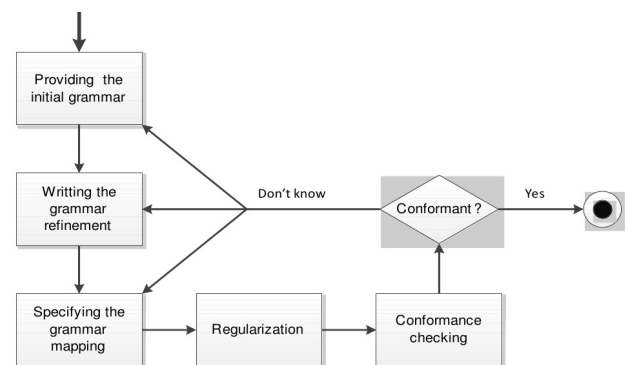


Fig 1. Refinement work-flow

Next activity is *writing the grammar refinement*. This activity is oriented to refine the initial grammar by reflecting structural features and properties not present in such an initial grammar (e.g., associativity and precedence of operators). In this way, the initial grammar should introduce the *core* syntactic constructs of the language, in form of *core* non-terminal symbols, while the refinement should address the structural refinement of these constructs, without changing the generated language.

Once the grammar refinement has been written, next activity is the *specifying the grammar mapping* activity. The goal of this activity is to make explicit a *grammar mapping* identifying which non-terminal symbols in the refinement correspond with each relevant (*core*) non-terminal symbol in the initial grammar.

Next step is *regularization*. This activity, which is carried out automatically, actually is the first part of the conformance checking method. Its goal is to associate, with each *core* non-terminal symbol, a *definitional regular expression*. Definitional regular expressions must involve both terminal and *core* non-terminal symbols. In addition, each sentence in the language described by the definitional regular expression  $\epsilon$  for a *core* non-terminal  $A$  must be derivable from  $A$  (i.e.,  $\alpha \in L(\epsilon) \Rightarrow A \rightarrow^* \alpha$ ). Finally,  $\epsilon$  must completely characterize

one of the intermediary languages derived from  $A$ , in the following sense:  $A \rightarrow^* w \Rightarrow \exists \alpha \in L(\epsilon) (\alpha \rightarrow^* w)^3$ .

Finally, the last step is given by the *conformance checking* activity. This activity tries to actually check the weak equivalence on the basis of the definitional regular expressions associated with each core non-terminal in both grammars, as well as on the basis of the grammar mapping. Therefore, it constitutes the second part of the conformance checking method.

It is worthwhile to notice that, being the checking method necessarily incomplete, during its execution it will be possible to get one of two possible answers: (i) *the grammars are actually equivalent* and (ii) *it has not been possible to prove whether the grammars are equivalent*. While in the first case, the method will ensure the correctness of the performed refinement, in the second case the answer is not conclusive: indeed, the grammars could be actually equivalent in a way not envisioned by the method, and therefore we can't conclude the non-equivalence of the grammar. In this case, it could be possible to re-factor the refinement, to modify simultaneously both the refinement as the initial grammar, and even to rethinking the grammar mapping, as the iterative nature of the work-flow makes apparent. Next sections go inside each activity of this work-flow.

### III. PROVIDING THE INITIAL GRAMMAR, WRITING THE GRAMMAR REFINEMENT AND SPECIFYING THE GRAMMAR MAPPING

The first activity to do in the refinement work-flow is to provide a suitable *initial grammar*. As said before, this grammar formally characterizes the syntax of the computer language addressed (i.e., it is able to generate exactly the sentences of the language). However, it does not necessarily do it in a way which is convenient to undertake a systematic implementation of such a language. Therefore, it could be necessary to modify this initial grammar to yield a *grammar refinement*.

```
(a)
Sents ::= Sents ; Sent | Sent
Sent  ::= id := Exp
Exp   ::= Exp + Exp | Exp * Exp | id | (Exp)

(b)
SS ::= SS ; S | S
S  ::= id := E
E  ::= E + T | T
T  ::= F * T | F
F  ::= id | (E)
```

Fig 2. (a) An initial context-free grammar, (b) a refinement of (a)

In order to illustrate these aspects, let us consider the initial grammar shown in Fig. 2a. This grammar characterizes a simple language of assignment instructions, in which arithmetic expressions can be assigned to identifiers. However, it does not take care of characterizing precedence and associativity of operators. As a collateral consequence, it exhibits ambiguity. In order to solve these shortcomings, it is possible to *refine* the initial grammar, getting an equivalent grammar characterizing the mentioned priority and associativity of op-

erators. Fig. 2b outlines a possible refinement taking these features into account.

In addition to provide the initial grammar and to specify the grammar refinement, in order to get the benefits of our checking approach, grammar writers must make the structural relationships between initial grammars and grammar refinements explicit. It is done by specifying a *grammar mapping*. On one hand, such a mapping supposes to recognize a set of representative syntactic structures in the grammar refinement that result of refining structures in the initial grammar. It is done by identifying a set of *core* non-terminal symbols in the grammar refinement. In particular, the initial symbol must be one of these core symbols. On the other hand, this mapping makes it possible to associate to each core non-terminal symbol in the grammar refinement a distinct non-terminal symbol in the initial grammar. Following a similar convention, these symbols in the initial grammar will be called *core* non-terminal symbols of the initial grammar. Non-terminal symbols that are not core symbols we will be called *auxiliary* non-terminal symbols.

Concerning our example, the establishment of a grammar mapping is straightforward. Indeed, in the refinement of Fig. 2b we can identify three core symbols (SS, S and E), which are mapped respectively to Sents, Sent and Exp in the initial grammar of Figure 2a.

### IV. REGULARIZATION

As said before, the goal of the *regularization* activity is to associate with each core non-terminal a definitional regular expression. In addition, this expression must only comprise terminal symbols, and core non-terminal symbols. For doing so, the algorithm of Fig. 3 is used.

This algorithm successively visits each non-terminal symbol  $A$ , determining and refining a definitional regular expression  $\epsilon_A$  for  $A$ . It starts by determining an arbitrary order for the non-terminal symbols. While the final regular expressions will depend on this order, the results will be equivalent for any order chosen. Then, it visits each non-terminal symbol  $A$ . For this purpose, it begins by establishing a first value for the definitional regular expression  $\epsilon_A$  as  $\bigoplus \{ \alpha \mid A \rightarrow \alpha \in P \}$ , with  $P$  the set of syntax rules. Here, by  $\bigoplus \Gamma$  we denote  $\emptyset$  when  $\Gamma = \emptyset$ ,  $\alpha$  when  $\Gamma = \{ \alpha \}$  and  $\alpha_0 \mid \dots \mid \alpha_k$  when  $\Gamma = \{ \alpha_0, \dots, \alpha_k \}$  ( $k \geq 1$ ). Therefore  $\epsilon_A$  is initially set to the disjunction of the RHSs of the rules for  $A$ . Then it substitutes the expression  $\epsilon_B$  for each already visited *auxiliary* non-terminal  $B$  in  $\epsilon_A$  (it is denoted by  $\epsilon_A[B/\epsilon_B]$ ). Notice that core non-terminals in  $\epsilon_A$  are preserved, since this process only attempts to eliminate auxiliary non-terminals. Next, it simplifies  $\epsilon_A$  to replace several forms of immediate recursion by iteration. For this purpose, a well-known result borrowed from the theory of language equations is used [2], which makes it possible to derive

$$A = (\beta^* \delta \gamma^* \alpha)^* \beta^* \delta \gamma^* \quad (1)$$

from

<sup>3</sup>Here, as usual,  $\alpha$  denotes a sentential form –a string of terminal and non-terminal symbols–, and  $w$  a sentence –a string of terminal symbols–.

$$A = A\alpha A \mid \beta A \mid A\gamma \mid \delta \quad (2)$$

**Input:** (i) A context-free grammar  $G \equiv (S_N, S_T, S, P)$  -  $S_N$  is the set of non terminal symbols,  $S_T$  the set of terminal symbols,  $S$  the initial symbol, and  $P$  the set of syntax rules; (ii) The set of core non-terminals  $K$

**Output:** A set of equations, with an equation of the form  $A = e$  for each core non-terminal  $A$ , with  $e$  a definitional regular expression

**Method:**

```

let  $S_N = \{A_0, A_1, \dots, A_n\}$  in
for  $i = 0$  to  $n$ 
 $\epsilon_i := \Theta\{\alpha \mid A \rightarrow \alpha \in P\}$ 
for  $j = 0$  to  $i-1$ 
if  $A_j \notin K$  then
 $\epsilon_i := \epsilon_i[A_j / \epsilon_j]$ 
end if
end for
 $\epsilon_i := \text{simplify}(\epsilon_i, \text{normalize}(\epsilon_i))$ 
if  $A_i \in K$  then
for  $j = 0$  to  $i-1$ 
 $\epsilon_i := \epsilon_j[A_i / \epsilon_j]$ 
end for
end if
end for
return  $\{A_i = \epsilon_i \mid A_i \in K\}$ 

```

Fig 3. Regularization method.

(a)

**Ordering:** Sents, Sent, Exp

**Regularization:**

```

Sents = ( Sent ; )* Sent
Sent = id := Exp
Exp = ((id | (Exp) ) (+|*))*(id | (Exp) )

```

(b)

**Ordering:** F,T,E,S,SS

**Regularization:**

```

SS = S ( ; S )*
S = id := E
E = ((id | (E) ) *(id | (E) )
( + ((id | (E) ) *(id | (E) ) )*)

```

Fig 4. (a) Regularization of grammar in Fig. 2a; (b) Regularization of grammar in Fig. 2b.

This result is taken as a basic transformation pattern to eliminate several forms of immediate recursion during the regularization stage. In order to make it possible to apply this pattern, we need to start by *normalize* definitional regular expressions  $\epsilon_A$  for non-terminal symbols  $A$  in order to yield regular expressions of the form  $A\alpha A \mid \beta A \mid A\gamma \mid \delta$  equivalent to  $\epsilon_A$ . Such a normalization basically works by applying the identities

$$(\epsilon_0|\epsilon_1)\epsilon_2 = \epsilon_0\epsilon_2|\epsilon_1\epsilon_2 \quad (3)$$

$$\epsilon_2(\epsilon_0|\epsilon_1) = \epsilon_2\epsilon_0|\epsilon_2\epsilon_1 \quad (4)$$

to the begin and the end of the expression in order to push up the left and right recursive positions of  $A$ . Due to space constraints, these normalization details will be omitted here.

In order to exemplify regularization, Fig. 4 shows the results of the regularization activity when carried out on grammars of Fig. 2a and Fig. 2b.

## V. CONFORMANCE CHECKING

The last activity to be considered is *conformance checking* itself, which is carried out by the method of Fig. 5.

The first step involved in the activity is to *align* the core non-terminal symbols in the two regularizations, in such a way those regularizations use the same names for those symbols. It is indicated by the *align* operation (details omitted). In addition to aligning the names of the core non-terminal, this operation is supposed to replace the auxiliary non-terminals that could remain in the definitional expressions by  $\_$  ( $\_$  is assumed to not be allowed as a grammar symbol). Once aligned both regularizations, the next step actually addresses the checking process. For this purpose, it maintains two sets: (i)  $\Gamma$ , which contains the core symbols whose conformance must be checked, and (ii)  $V$ , which contains the core symbols whose conformance has been already undertook (*visited* non-terminals). Then, the checking process proceeds until  $\Gamma$  becomes empty.

**Input:** (i) The regularization of the initial grammar  $R_i$ ; (ii) The regularization of the refinement grammar  $R_r$ ; (iii) The terminal alphabet  $\Sigma_T$ ; (iv) The initial symbol  $S$  of the initial grammar; (v) The grammar mapping  $\Theta$  from the refinement to the initial grammar

**Output:** "yes" if the conformance can be proved, "don't know" otherwise

**Method:**

```

( $R_i, R_r$ ) := align( $R_i, R_r, \Theta, \Sigma_T$ )
 $\Gamma := \{S\}; V := \emptyset$ 
while  $\Gamma \neq \emptyset$ 
pick  $A$  from  $\Gamma$ 
 $\Gamma := \Gamma - \{A\}; V := V \cup \{A\}$ 
pick  $A = e_A^l$  from  $R_i$ 
pick  $A = e_A^r$  from  $R_r$ 
if  $\_ \in e_A^l \vee \_ \in e_A^r$  then
return "don't know"
fi
if  $e_A^l \sim e_A^r$  then
 $\Gamma := \Gamma \cup \{B \mid B \in e_A^l \wedge B \notin V \cup \Sigma_T\}$ 
else
return "don't know"
end if
end while
return "yes"

```

Fig 5. Conformance checking method

In each iteration, a core symbol  $A$  in  $\Gamma$  is chosen, it is recorded as visited, and the definitional expressions for  $A$  in the initial grammar ( $e_A^l$ ) and in the grammar refinement ( $e_A^r$ ) are considered (remember the aligned regularizations shares the names for the core non-terminals). If  $e_A^l$  or  $e_A^r$  contain a  $\_$  symbol, it means regularization failed to produce definitional expressions comprising only terminal and core non-terminal symbols. Thus, the checking process ends with a non-conclusive response. Otherwise,  $e_A^l$  and  $e_A^r$  are checked for equivalence (i.e., it is studied whether  $e_A^l \sim e_A^r$  holds). If the test fails (i.e.,  $e_A^l$  and  $e_A^r$  are not indeed equivalent), the overall process finishes with a non-conclusive answer. Otherwise, the method tries to ensure that, if  $\alpha \in L(e_A^l)$  (and, thus,  $\alpha \in L(e_A^r)$ ), then it is possible to derive exactly the same sentences from  $\alpha$  in the initial grammar than in the grammar refinement. Indeed, if it holds for any core non-terminal in  $\alpha$ , it will hold for the overall sentential form. For this purpose, the method schedules the checking of those core non-terminals in  $\alpha$  not yet visited. Thus, if the set  $\Gamma$  is finally emptied, it is possible to ensure that all the *proof obligations* concern-

ing the core non-terminal symbols scheduled by the method have been satisfied, and, therefore, the equivalence has been effectively proven. In this way, it is possible to finish with a conclusive and positive answer.

Alignment of core non-terminal names	
Aligned regularization for the initial grammar	
Sents = ( Sent; )* Sent Sent = id := Exp Exp = ((id   (Exp) )+(*)*(id   (Exp) )	
Aligned regularization for the grammar refinement	
Sents = Sent ( ; Sent)* Sent = id := Exp Exp = ((id   (Exp) ))**(id   (Exp) ) (+ ((id   (Exp) ))**(id   (Exp) ) )*	
Conformance checking	
Init	$\Gamma = \{\text{Sents}\}$ $V = \emptyset$
It. 1	<b>Symbol to check:</b> Sents $\Gamma = \emptyset$ $V = \{\text{Sents}\}$ $\mathcal{L}(\text{Sent};)^* \text{Sent} \sim \text{Sent} ( ; \text{Sent})^* ?$ : <b>YES</b> $\Gamma = \{\text{Sent}\}$
It. 2	<b>Symbol to check:</b> Sent $\Gamma = \emptyset$ $V = \{\text{Sents}, \text{Sent}\}$ $\mathcal{L} \text{id} := \text{Exp} \sim \text{id} := \text{Exp} ?$ : <b>YES</b> $\Gamma = \{\text{Exp}\}$
It. 3	<b>Symbol to check:</b> Exp $\Gamma = \emptyset$ $V = \{\text{Sents}, \text{Sent}, \text{Exp}\}$ $\mathcal{L}((\text{id}   (\text{Exp}))+(*)^*(\text{id}   (\text{Exp})) \sim$ $((\text{id}   (\text{Exp}))^*)^*(\text{id}   (\text{Exp}))$ $(+ ((\text{id}   (\text{Exp}))^*)^*(\text{id}   (\text{Exp})) )^* ?$ : <b>YES</b>
End	Success (YES answer) !

Fig 6. Checking the conformance of grammar in Fig. 2b with respect to grammar in Fig. 2a

Finally, notice that the conformance checking method relies on checking the equivalence of regular expressions. For this purpose, it is possible to use one of the well-known approaches reported in the literature (see, for instance, [1]), which, in last term, rely on converting regular expressions to equivalent finite automata, and to check equivalence between automata.

Fig. 6 details the application of the conformance checking method to the grammar refinement of Fig. 2b and the initial grammar of Fig. 2a.

## VI. CONCLUSIONS AND FUTURE WORK

Grammar refinement is a usual activity in language engineering. Initial context-free grammars are refined to impose finer structures on the initial syntactic categories, in order to make explicit important features of the target language (e.g., operator associativity and precedence). Grammars are also refined to get equivalents satisfying the constraints imposed by particular development tools (e.g., parser generators). Therefore, checking the correctness of refinements with respect to initial grammars should be a must in any systematic language engineering process. Although the unconstrained problem is undecidable, this paper has shown how it is still possible to provide some useful automatic assistance. For this purpose, it has proposed an interactive approach, focused on checking equivalence of definitional regular expressions for core non-terminals

We are currently improving the efficiency of the different algorithms involved in our proposal. We are also investigating the inclusion of new regularization patterns and strategies. In addition, we want to check the approach with several grammars for non-trivial domain-specific languages. As future work, we want to use this approach to check the conformance of processing-oriented grammars with respect to XML schemas in order to assist the language-oriented processing of XML documents [5].

## ACKNOWLEDGMENT

Thanks are due to the project grants TIN2010-21288-C02-01.

## REFERENCES

- [1] Aho A.V, Ullman J.D. The Theory of Parsing, Translation and Compiling. Vol. I - Parsing. Prentice-Hall. 1972
- [2] Andrei, S., Chind, W-N., Cavadini, S.V. Self-embedded context-free grammars with regular counterparts. Acta Informatica 40(5): 349–365. 2004
- [3] Bar-Hillel, Y., Perles, M., Shamir, E. On formal properties of simple phrase-structure grammars. Z. Phonetik, Sprachwiss. Kommunikationsforsch 14, 143-172. 1961 (Reprinted in Bar-Hillel. Language and Information, Addison-Wesley, 1964)
- [4] Paull, M.C., Unger, S.H. Structural Equivalence of context-free grammars. Journal of Comp. and System Sc., 2(4), 427-463. 1968
- [5] Temprado-Battad, B., Sarasa, A., Sierra, J.L. Modular Specifications of XML Processing Tasks with Attribute Grammars Defined on Multiple Syntactic Views. 5<sup>th</sup> International Workshop on Flexible Database and Information Systems. Bilbao, Spain. 2010

# Identification of Patterns through Haskell Programs Analysis

Ján Kollár, Sergej Chodarev, Emília Pietriková and Ľubomír Wassermann

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics  
 Technical University of Košice, Slovak Republic

E-mail: {jan.kollar, sergej.chodarev, emilia.pietrikova, lubomir.wassermann}@tuke.sk

**Abstract**—Usage of appropriate high-level abstractions is very important for development of reliable and maintainable programs. Abstractions can be more effective if applied at the level of language syntax. To achieve this goal, analysis of programs based on the syntax is needed.

This paper presents Haskell Syntax Analyzer tool that can be used for analysis of Haskell programs from the syntactic perspective. It allows to retrieve derivation trees of Haskell programs, visualize them and perform their statistical analysis. We also propose approach for recognition of recurring patterns in programs that can be used as a basis for automated introduction of abstractions into the language.

## I. INTRODUCTION

**A**BSTRACTION is one of the fundamental concepts in computer science. Abstraction allows expressing things more simple by defining new more abstract concepts, that encapsulate complex expressions. This allows to hide implementation details.

For example, expression  $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$  for computing one root of quadratic equation  $ax^2 + bx + c = 0$  can be simplified by introducing new abstract concept – *discriminant* ( $D$ ). This form can be even more simplified by defining abstraction corresponding to the whole expression (see Fig. 1).

As a disadvantage of domain-specific languages (DSLs), the price of understanding DSL technology is often referred [1], [2]. This is caused by necessity of knowledge of the language design field and of the problem domain. In case of identification of software system field of where it is appropriate to deploy DSL, it is important to design the language suitably (syntax and notations). To increase the usability of a new DSL, the use of terminology and concepts of the target domain is essential.

The purpose of this work is not to derive a grammar from samples of different languages, but taking the full grammar of a language (in our case Haskell), it is evaluated.

## II. ABSTRACTION BASED ON PROGRAM PATTERNS

To achieve the mentioned goals, first we need to solve the problem of recognition of recurring patterns in a code. Manual analysis of the code may be hard and tedious task. On the other hand, tools for automatic patterns recognition can greatly help in this task. Moreover, the recognition needs to be done at the level of program syntax.

Program patterns mean code fragments extracted from a set of sample programs that have equivalent syntactic, and

hence, also semantic structure. Patterns can also contain parts that are different in each program. These parts can be called syntactic variables. After introduction of new abstraction based on a pattern, syntactic variables will become parameters of the abstraction.

Expressiveness of the language can be improved by the recognition of program patterns and introduction of abstractions based on them. Moreover, it allows more natural and strain-forward expression of programs.

This approach can be also useful for development of domain-specific dialects of programming languages. In order to implement this transition from general purpose language to its domain-specific dialect, it is necessary to reflect the fundamental differences between the domain-specific dialect and the corresponding GPL. The main differences lie in the following points:

- focus on a particular domain,
- use of concepts from a domain,
- higher abstraction.

To achieve a connection with particular domain and a shift towards domain specificity, it is suitable to analyze existing programs (or program fragments) of written in the GPL solving various problems from the domain. On the basis of this analysis, a shift from GPL to domain-specific dialect can be achieved.

The goal of this article is to propose a solution for automated introduction of new language abstractions based on patterns found in a code.

## III. HASKELL SYNTAX ANALYZER TOOL

To achieve the goal, *Haskell Syntax Analyzer* tool has been used. The aim of creating *Haskell Syntax Analyzer* tool is to gather needed information from Haskell programs to get a proper knowledge about used constructs in analyzed programs. As a result of the program analysis, derivation tree is produced, consisting of used rules of Haskell grammar [3]. Architecture of Haskell Syntax Analyzer tool consists of two parts – *generating infrastructure* and *analyzing infrastructure*.

The goal of generating infrastructure is to prepare tools that are used during the analysis of Haskell programs by the analyzing infrastructure.

The analyzing infrastructure contains lexer and parser of Haskell programs, intended for analysis of Haskell programs

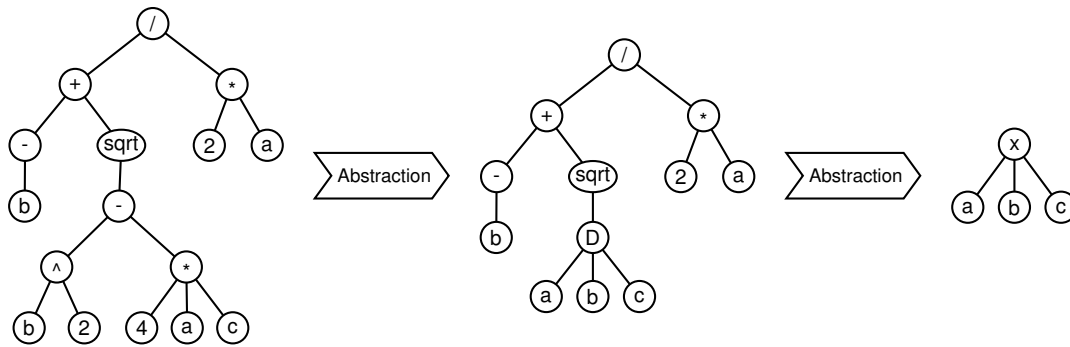


Fig. 1. Simplification of expression structure using abstraction.

into lexical units and then processing them into derivation trees.

Derivation trees are produced in the XML format, subsequently visualized using the Graphviz (Graph Visualization Software) tool [4] and further processed to retrieve statistical data on Haskell programs and to recognize common language patterns.

#### A. Haskell 98 Syntax

Haskell is a general purpose, purely functional programming language that provides higher-order functions, non-strict semantics, static polymorphic typing, user-defined algebraic data types, pattern-matching, list comprehensions, a module system, a monadic I/O system, and a rich set of primitive data types, including lists, arrays, arbitrary and fixed precision integers, and floating-point numbers [5].

In [6], Haskell syntax was processed into HTML format. This form was chosen as a basis for development of the parser. It uses extended Backus-Naur Form with the following additions:

- parentheses for grouping symbols,
- optional symbols marked using question mark (?),
- 0..n repetition marked with star (\*),
- 1..n repetition marked with plus (+).

To speed-up the development, it is necessary to use a parser generator tool. The choice of the appropriate parser generator depends on a class of the processed grammar. In Haskell grammar, it is possible to find several cases using the left recursion, e.g.:

```

aexp ::= qvar
      | gcon
      | literal
      | (exp)
      | (exp (,exp) +)
      | [exp (,exp) *]
      | [exp (,exp) ?.. (exp) ?]
      | [exp | qual (,qual) *]
      | (exp_i qop)
      | (qop exp_i)
      | qcon ( fbind (,fbind) * ) ?
      | aexp fbind (,fbind) *

```

As the goal of the *Haskell Syntax Analyzer* tool is not to transform the language grammar but to process it in its original form, and because of the mentioned left recursion, Haskell grammar has been treated as the *LR* type.

In the grammar, reduce/reduce conflicts may be found, regarding several rules sharing the same right side. Thus, it is difficult for a parser to choose the right nonterminal to reduce. According to the above facts, it is appropriate to use GLR parser generator, as it is capable of parallel reduce of each nonterminal, trying to proceed with multiple possibilities.

In compliance with the project architecture, Haskell Grammar has been transformed into XML form in order to provide better representation of the grammar. The transparent form (XML) of the Haskell grammar is suitable for further processing, including generation of input for a parser generator.

As the parser generator, Bison [7] has been chosen, because of its ability to generate a GLR parser.

The Haskell language allows to use an indentation to define blocks of code. On the other hand, it still allows to define the blocks using braces and to separate the statements using semicolons. For this reason, it is required to introduce a separate step into lexical analysis, during which the layout defined by the white space characters is replaced by semicolons and braces.

Lexical analyzer is generated by the Flex tool [8] that is based on the specification produced according to the Haskell lexical grammar. After this step, the white space characters in the program are analyzed following the algorithm specified in Haskell 98 Report [3]. Based on this analysis, stream of tokens is extended by tokens corresponding to braces and semicolons.

#### B. Transformation of Haskell 98 Grammar

To be able to process the grammar programmatically, it is required to transform HTML form of grammar to more suitable representation. To transform the Haskell grammar to the appropriate form, a grammar transformer was created. This tool first transforms HTML representation of Haskell 98 grammar to XML representation that is more suitable for further processing.

XML grammar is then processed to create Java object model where XML elements are mapped to the instances of the grammar model classes. Java object model of Haskell grammar



provides a better way of how to manipulate and operate on the Haskell grammar.

Mapping is shown in the example of a grammar rule:

```
gdrhs ::= gd = exp ( gdrhs )?
```

This rule is then transformed to the following XML fragment:

```
<nonterm id="n1" label="gdrhs">
  <sequence id="seq1">
    <nonterm id="n2" label="gd"/>
    <term id="equals" label="="/>
    <nonterm id="n3" label="exp"/>
    <option id="opt1">
      <nonterm id="n1" label="gdrhs"/>
    </option>
  </sequence>
</nonterm>
```

Java object model of Haskell grammar is then used to generate grammar specification in a format suitable for the Bison parser generator. During this process, grammar rules need to be transformed from EBNF form into BNF accepted by Bison. For this reason, new helper nonterminals are introduced. They correspond to constructs that can not be expressed directly in BNF, like repetition or optional expression. These nonterminals are specially marked according to their meaning, so it is possible to create derivation tree corresponding to original form of the grammar.

### C. Grammar Ambiguity

Meanwhile the parsing, several ambiguities were detected. For example, rule `pat_i` of Haskell 98 grammar is defined as:

```
pat_i ::= pat_i ( qconop pat_i )?
        | - ( integer | float )
        | pat_10
```

After `pat_10` was reduced to `pat_i` ( $3^{rd}$  alternative), it was possible to reduce `pat_i` to `pat_i` again ( $1^{st}$  alternative). GLR parser generator, that is used within the Haskell Syntax Analyzer tool, disjoined at the mentioned point. As both ways led to the same nonterminal, they joined again. However, the parser was not capable to determine which way to use/throw away.

The problem was solved after modification of the critical rule:

```
pat_i ::= pat_10 ( qconop pat_i )?
```

A branch with ‘-’ was moved to `pat_10`:

```
pat_10 ::= apat
         | gcon ( apat )+
         | - ( integer | float )
```

Another rule `exp_i` was changed by analogy of the previous rule. Original form:

```
exp_i a ::= exp_i ( qop exp_i )?
```

```
| - exp_i
| exp_10
```

After modification:

```
exp_i a ::= exp_10 ( qop exp_i )?
| - exp_i
```

After such modifications, parser was able to process simple Haskell programs. Moreover, number of conflicts in the grammar were decreased.

### D. Fixity Resolution

Another problem, that had been needed to be solved, was a resolution of fixity and precedence of operators. In Haskell 98 Report [3], operators precedence levels were defined using separate grammar rules (like  $exp_i$  for  $0 \leq i \leq 9$ ). In the version of the grammar that had been used as a source for the transformation, the indexed rules were replaced by a single rule  $exp_i$ .

On the other hand, Haskell 2010 Report [9] defines expressions in a different way. It defines a single rule  $infixexp$  for all the precedence levels and associativities. The resolution of expressions is then performed after parsing. This approach is also appropriate for our purposes.

So after the parsing, resulting derivation trees are processed. All occurrences of the `exp_i` element are resolved using the algorithm described in the Haskell 2010 Report [9].

## IV. CODE STATISTICS

Using the developed tool, it was possible to compute some interested statistics based on a set of about 300 Haskell sample programs. Result of the analysis of a program is its derivation tree according to the language grammar. The derivation tree consists of terminal and nonterminal symbols in the grammar, where terminal symbols represent leaves of the tree. The derivation tree also contains helper nodes corresponding to EBNF features like repetition or optional elements.

One of the parameters, that may be investigated, is a relative occurrence of symbols in derivation trees. Relative occurrence of a symbol in a program is defined as:

$$r_{sym} = \frac{n_{sym}}{N}$$

where  $n_{sym}$  means a number of occurrences of the  $sym$  symbol in the derivation tree of a program and  $N$  represents a number of all symbols/nodes of the derivation tree.

Table I represents average occurrences greater than 0.01 of particular symbols in all programs of our sample. As it can be expected, variable names and expressions have the greatest frequency of all symbols.

It is possible to make such statistics for especially selected sample of programs for a specific domain. It will show which language elements are used in programs of the domain and which elements can be omitted from the domain-specific dialect.

Statistical analysis can also be used to partition a sample of programs into groups based on a usage of the language elements.

TABLE I  
PROPORTION NUMBER OF SYMBOL OCCURRENCES

Symbol	Occurrence	Symbol	Occurrence
varid	0,093855	aexp	0,092660
fexp	0,092660	exp_10	0,063059
qvar	0,051154	exp_i	0,049428
exp	0,044523	var	0,037632
apat	0,033349	conid	0,026202
(	0,019259	)	0,019259
=	0,018341	decl	0,017526
;	0,017277	qop	0,016620
topdecl	0,016484	rhs	0,016164
pat_i	0,016099	pat_10	0,016099
qvarop	0,015110	gcon	0,014879
atype	0,013402	varsym	0,012450
qcon	0,011951	btype	0,011516
,	0,011309	funlhs	0,010876
type	0,010080		

## V. PATTERNS RECOGNITION

To find patterns in the program derivation tree, a simple algorithm can be used that is based on the function *findPatterns* defined below:

```

parents ← allParents(elements)
groups ← findGroups(parents)
if groups is empty then
  return [groups]
else
  for all group ∈ groups do
    Add findPatterns(group) to foundGroups
  end for
  return mergeGroups(foundGroups)
end if

```

Function *findPatterns* takes a list of the tree elements and recursively examines their parents to find a set of groups of subtrees that have a similar structure. It uses helper functions with the following meaning:

- *allParents* – returns a set of parents of all tree elements in a group;
- *findGroups* – given a set of tree elements, returns list of groups of elements with similar subtrees;
- *mergeGroups* – merges list of lists of groups into a single list.

To initiate the algorithm, the *findPatterns* function is called on terminal symbols of the tree. Then it tries to walk up to the root of the tree while it can find groups of subtrees with similar structure.

Result of the algorithm is a list of groups of subtrees, where each group corresponds to a found pattern and contains all occurrences of the pattern.

## VI. CONCLUSION

Set of tools called *Haskell Syntax Analyzer* that has been developed within this paper is intended to analyze programs based on the language syntax, resulting in providing appropriate derivation trees. In this paper, usage statistics of the

Haskell syntactic symbols are provided, creating a vision of the language application within specific domains of use.

Moreover, the analysis made possible to accomplish pattern recognition in program codes, with the perspective of development of new language dialects, both general-purpose and domain-specific. The term of program patterns has been used for syntactically, and hence, also semantically equal program fragments occurring in a set of program samples.

The most significant contribution, that we expect based on the partial results presented in this paper, is the contribution for automated software evolution. Clearly, this would mean to shift from a language analyzer to the language abstracter, associating concepts to formal language constructs [10], [11], and formalizing them by means of these associations. In this way, we expect to integrate programming and modeling, associating the general purpose and domain-specific languages [12], [13], [14].

## ACKNOWLEDGMENT

This work was supported by VEGA project No. 1/0015/10 “Principles and methods of semantic enrichment and adaptation of knowledge-based languages for automatic software development.”

## REFERENCES

- [1] M. Mernik, J. Heering, and A. M. Sloane, “When and how to develop domain-specific languages,” *ACM Comput. Surv.*, vol. 37, no. 4, pp. 316–344, 2005.
- [2] M. Crepinsek, M. Mernik, B. Bryant, F. Javed, and A. Sprague, “Inferring context-free grammars for domain-specific languages,” *Electronic notes in theoretical computer science*, vol. 141, no. 4, pp. 99–116, 2005.
- [3] S. Peyton Jones, *Haskell 98 Language and Libraries – The Revised Report*. Cambridge, England: Cambridge University Press, 2003.
- [4] J. Ellson, E. Gansner, L. Koutsofios, S. North, and G. Woodhull, “Graphviz – open source graph drawing tools,” in *Graph Drawing*, ser. Lecture Notes in Computer Science, P. Mutzel, M. Jünger, and S. Leipert, Eds. Springer Berlin / Heidelberg, 2002, vol. 2265, pp. 594–597.
- [5] B. O’Sullivan, J. Goerzen, and D. Stewart, *Real World Haskell*, 1st ed. O’Reilly Media, Inc., 2008.
- [6] P. Herceek, “Haskell 98 report,” Available: <http://www.hck.sk/users/peter/HaskellEx.htm>, 2007.
- [7] C. Donnelly and R. Stallman, *Bison: The Yacc-compatible Parser Generator*, 2010, available: <http://www.gnu.org/software/bison/manual/>.
- [8] V. Paxson, W. Estes, and J. Millaway, *Lexical Analysis With Flex*, 2007, available: <http://flex.sourceforge.net/manual/>.
- [9] S. Marlow, “The Haskell 2010 Language Report,” Available: <http://www.haskell.org/onlinereport/haskell2010/>, 2010.
- [10] J. Porubän and P. Václavík, “Extensible language independent source code refactoring,” in *AEI '2008 : International Conference on Applied Electrical Engineering and Informatics, Greece, Athens, September 8-11*. Košice: FEI TU, 2008, pp. 58–63.
- [11] J. Porubän and M. Sabo, “Jessine: Integrating rules in enterprise software applications,” *Journal of Information, Control and Management Systems*, vol. 7, no. 1, pp. 81–88, 2009.
- [12] M. Sabo and J. Porubän, “Preserving design patterns using source code annotations,” *Journal of Computer Science and Control Systems*, vol. 2, no. 1, pp. 53–56, 2009.
- [13] P. Václavík, “Application domain name-based analysis,” *Journal of Computer Science and Control Systems*, vol. 2, no. 2, pp. 66–69, 2009.
- [14] I. Luković, P. Mogin, J. Pavičević, and S. Ristić, “An approach to developing complex database schemas using form types,” *Software Practice & Experience*, vol. 37, no. 15, pp. 1621–1656, 2007.

# Computer Language Notation Specification through Program Examples

Miroslav Sabo, Jaroslav Porubán, Dominik Lakatoš, Michaela Kreutzová  
Technical University of Košice  
Letná 9

042 00 Košice, Slovakia

Email: {miroslav.sabo, jaroslav.poruban, dominik.lakatos, michaela.kreutzova}@tuke.sk

**Abstract**—It often happens that computer-generated documents originally intended for human recipient need to be processed in an automated manner. The problem occurs if analyzer does not exist and therefore must be created ad hoc. To avoid the repetitive manual implementation of parsers for different formats of processed documents, we propose a method for specification of computer language notation by providing program examples. The main goal is to facilitate the process of computer language development by automating the specification of a notation for recurring well-known language constructs often observed in various computer languages. Hence, we introduce the concept of language patterns, which capture the knowledge of language engineer and enables its automated application in the process of notation recognition. As a result, by using the proposed method, even a user less experienced in the field of computer language construction is able to create a language parser.

## I. INTRODUCTION

**S**OFTWARE systems generate a lot of textual output, either as a main product of their execution or for other secondary purposes such as logging. If the output is intended for information transfer and further processing by another system, its structure must be explicitly defined (e.g. XML and XSD) in order for receiving system to transform the textual content into appropriate structural representation. On the other hand, if the output is intended for human recipient, the structure of textual content is often not defined explicitly but is rather hidden in its human-usable notation. In most of these cases, the explicit specification of structure is not even necessary, as primary purpose of such textual output is to store the information in form which is easily comprehensible to human users just by reading it. However, sometimes it happens that output originally intended for human has to be processed by computer (e.g. to perform an analysis). Since explicit specification of its structure does not exist, it must be defined first.

Considering the generative origin and human-usable notation, we think of such textual output as of collection of programs written in some domain-specific language (DSL). From this perspective, the task of specifying a structure can be translated into a *problem of DSL notation specification*. One of the options is to examine the source code of the system generating the output and construct the grammar of a DSL accordingly. Besides the complexity of such task, it requires access to source code, which might not always be feasible.

That leaves us the specification of a notation by inferring it from the provided program examples.

Although syntax recognition is a well-established area of research and multiple approaches to grammar inference have already been implemented [1]–[3], in this paper we propose a method for example-driven DSL notation specification (EDNS) based on language patterns. The innovative concept of *language patterns* is proposed to capture the well-known recurring notation of common language constructs often seen in many computer languages. Automated identification of such patterns in provided program examples eliminates the necessity to specify the same notation manually and repeatedly for each language being designed. Along the recognition of recurring notations, language patterns are also used to check whether the notation satisfies the conditions (e.g. unambiguity) of a machine-processable computer language.

Our objective is to automate the process of DSL notation specification as much as possible. By using EDNS method, it is possible to perform this task even for a user with little knowledge about the language construction. Moreover, the concept of formalized language patterns allows to incorporate into the process a domain expert with no technical knowledge as well. Besides the analytic way of using language patterns in context of EDNS method, they can also be utilized in the synthetic manner when specifying a notation by their composition [4].

The rest of the paper is organized as follows. In the next section we give a detailed description of approaches to computer language inference, with special emphasis on DSLs. The overview of EDNS method is given in Section III. Following sections discuss the individual artifacts used in the method – Section IV describes the specification of abstract syntax of a DSL, while Section V describes ProgXample, the mechanism for formalization of textual program examples which represent the concrete syntax of a DSL. Section VI elaborates in detail on the proposed concept of language patterns and discusses its application in the proposed EDNS method. Finally, Section VII gives the conclusions of the paper.

## II. RELATED WORKS

The problem of syntax recognition from existing resources has been specifically addressed by grammar inference research community. Successful results have been achieved in inferring

regular languages using various algorithms like EDSM [5] and RPNI [6]. Genetic-programming approach was used for inferring regular grammars in [2]. The methodologies of context-free grammars induction for domain-specific languages are the aim of GenParse Project research group. Their initial research of genetic approach resulted in the successful induction of small grammars [7], however their current focus is on the incremental grammar learning [1] which should allow for bigger DSLs to be inferred as well.

Although yet another various approaches both automatic [8] or semi-automatic [9], [10] exist, they are all targeted towards language design based on the concrete syntax which is recognized solely from artifacts represented as sentences/programs written in the unknown language. The aim of our research is focused, however, on the language development based on abstract syntax since we advocate that language specification with separated abstract and concrete syntaxes is more suitable for DSL design. The fact that many supporting tools for DSL development [11], [12] follow this approach makes for a promising research area to explore.

### III. EXAMPLE-DRIVEN METHOD FOR DSL NOTATION SPECIFICATION

In traditional approach to computer language development [13], central part of the language specification is grammar which defines the concrete syntax (notation) of a language. Abstract syntax is not defined explicitly but can be derived from the grammar.

In model-driven approach to DSL development, it is a usual practice to define abstract and concrete syntaxes separately [14], [15], using some metamodeling and grammar-like languages respectively. The DSL notation is specified formally by writing grammar-like rules containing domain-specific keywords and references to elements of language metamodel. This process is performed by language engineer. As the required domain-specific notation is achieved by amending the rules with domain-specific keywords, we can look at the process of notation specification as a tailoring of the computer-like syntax to meet the requirements on look and feel of the specific domain.

*Example-driven method for DSL notation specification (EDNS)* takes the opposite direction and starts with ideal domain-specific notation, provided informally by domain expert through examples of programs as they would have been written in a desired DSL (Fig. 1). The formal specification is then derived from examples using the concept of language patterns. The method consists of three consecutive phases – *formalization of program examples*, *DSL notation recognition* and *generation of language specifications*.

#### A. Formalization of Program Examples

Formalization of Program Examples can be considered a preprocessing phase to the main phase of EDNS method where the actual process of notation recognition happens. Its purpose is to formalize the program examples which are given

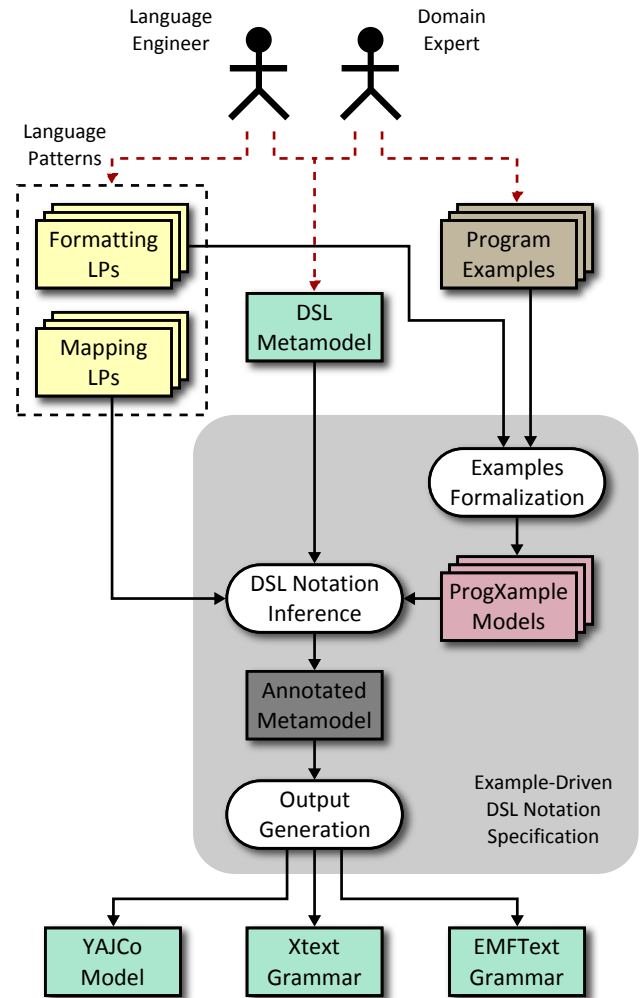


Fig. 1. Overview of the example-driven method for DSL notation specification.

informally as plain text files, so they can be processed in an automated manner in the following phase.

The resulting formal representation is tree models defined in Ecore-based *ProgXample modeling language*. The process of formalization is automated and models are created by transforming the initial models using the *formatting language patterns*. Initial models are constructed as trivial trees with single node holding the whole textual content of an example file. Creating of ProgXample models will be discussed in more detail in Sec. V.

#### B. DSL Notation Inference

DSL Notation Inference is the main phase of EDNS method. It has three inputs – *DSL metamodel*, *ProgXample models* and *mapping language patterns*. DSL metamodel represents the abstract syntax of a language and is the result of common domain analysis conducted by both language engineer and domain expert. ProgXample models are the output of previous preprocessing phase and they are the formalization of concrete syntax proposed by domain expert. The last input is a set

of language patterns which drive the process of notation recognition. Patterns are in an iterative manner compared against metamodel and ProgXample models and if the match is found, the appropriate elements of metamodel are injected with special annotations. The output of the notation recognition process is the metamodel enriched with annotations marking the identified patterns. This metamodel is an internal representation of the language specification in EDNS method.

### C. Output Generation

EDNS method does not directly concern the implementation of languages or their supporting tools. For this purpose, it rather exploits the maturity of existing technologies in this area. There are many tools for language design and implementation, varying from simple parser generators [4] to complex language workbenches [14], [15], which can generate the full implementation of a language given just its specification. Moreover, the more sophisticated tools can generate other supporting tools (editor, debugger, visualizer, etc.) or even a fully-fledged language-specific IDE as well. These tools are in the focus of EDNS method and to utilize their generative capabilities, EDNS provides them with language specifications in the appropriate format.

Output generation is the final phase of EDNS method. It takes a single input produced by the preceding phase of language notation recognition – annotated metamodel of a language. At first, Ecore metamodel serves as a source for instantiating the in-memory Java objects which constitute the internal YAJCo model [16]. This model represents the implementation-independent language specification and itself can be used for generation of the language parser and supporting language-specific tools. The specification files for other tools for language implementation (e.g. Xtext, EMFText) can be generated from YAJCo model using the Generator tool provided by YAJCo framework [16].

## IV. LANGUAGE METAMODEL

*Language metamodel* is the central part of computer language specification in the proposed EDNS method, as it defines the abstract syntax of a language. It is composed of domain concepts, their properties and relations between the concepts. The metamodel is created to formally capture the output of domain analysis conducted together by both language engineer and domain expert. In EDNS method, a metamodel is constructed using the *EMF Ecore Metamodel*. Each domain concept is represented by a single class and concept's properties are modeled using the class attributes. The relations between concepts are in metamodel represented as named connections between classes. Since metamodel is defined in EMF Ecore format, any of the existing EMF editors can be utilized in this phase.

## V. PROGXAMPLE MODELS

In EDNS method, concrete syntax of a language is defined informally by providing the examples of programs as they would have been written in a DSL being specified. For

these examples to be processable in the following phase of automated notation recognition, first they must be formalized. For this purpose, a specialized modeling language, tailored for the needs of EDNS method, has been proposed.

*ProgXample* is a compositional modeling language for tree representation of textual content. It is defined on EMF Ecore platform by its metamodel which specifies the components of tree models that can formally represent the plain text of program examples. The special facet of this language is that its metamodel is not defined as a monolithic structure, but it is rather composed of extensions attached to the core metamodel.

The construction of trivial tree is the starting point of formalization of every program example and it is same for every example file. Comparing the trivial tree model to a flat representation of textual content, it indeed does not bring any additional information on the structure of the program example. However, tree model is iteratively compared against formatting language patterns and if the match is recognized, the model gets transformed into more complex form, possibly being augmented with nodes of other kinds, introduced to metamodel by various patterns. After all patterns have been compared, the final form of tree models is handed as an input to the following phase of DSL notation recognition.

## VI. LANGUAGE PATTERNS

The concept of *language pattern* in the area of computer language design has been inspired by the concept of design pattern in the area of object-oriented software design [17]. In analogy with design patterns, language patterns capture the language design knowledge in a form that can be reused effectively. The captured knowledge serves two purposes:

- 1) captures the widely known and accepted notation of particular language constructs, often used throughout various programming languages (e.g. punctuation, delimiter marks, bracketing)
- 2) controls the specified notation that it satisfies requirements on being computer-processable (e.g. problem of ambiguity in a language)

Language patterns are the foundation of a proposed method for example-driven DSL notation specification [18]–[20]. According to their utilization, they come in two flavors – *formatting language patterns* and *mapping language patterns*.

Formatting language patterns (FLPs) are used in the early phase of EDNS method when ProgXample models are being created. Their purpose is to formalize the plain text of example files into form that will be processable in the following phase of notation inference. FLPs only concern the concrete syntax of a language therefore the only artifacts included in the process of pattern recognition are program examples which define the notation of a language.

Mapping language patterns (MLPs) are used in the main phase of EDNS method when the actual process of DSL notation inference happens. Their purpose is to infer the concrete human-usable notation of a language from the provided program examples. The inferred notation is then defined as mapping between abstract and concrete syntaxes. Since MLPs

concern both syntaxes of a language, the artifacts involved in the process of inference include both DSL metamodel representing the abstract syntax and ProgXample models (formalized program examples) representing the concrete syntax.

## VII. CONCLUSION

Capturing the recurring notation style of common language constructs and its formalization in form of computer language patterns is an unexplored topic in the area of computer language design. In this paper we have elaborated on this novel idea and have discussed its application in context of model-driven language development. The proposed method for example-driven DSL notation specification (EDNS) has been introduced. The paper has presented in detail the concept of formatting and mapping language patterns and its application in EDNS method.

The language patterns open new possibilities in construction of computer languages. They can be utilized in both directions to creating a language specification, analytical and synthetic. Although synthetic approach was not discussed in this paper, more details on this subject can be found in our earlier work [4]. This paper presented the analytical approach, set up in the context of proposed EDNS method. We believe that by using it there is a potential for speeding up the process of language creation since DSL notation is inferred from program examples automatically and the only part of DSL specification that must be performed manually by language engineer boils down to construction of DSL metamodel. Besides the cases where program examples are provided as an output generated by software, the examples can be created manually as well. This gives a possibility to define DSL notation even for a person not versed in language construction, nevertheless the most competent for such task – a domain expert. Since having a domain expert as the direct author of DSL notation can significantly increase the quality and usability of developed DSL, we consider this an important benefit offered by EDNS method.

Currently we are working on development of graphical user interface for EDNS method. This will enable the visualization of identified language patterns in provided program examples. Moreover, it will facilitate the necessary process of amending the examples in situations when proposed notation is not machine-processable and DSL notation can not be inferred. The success of notation inference is directly influenced by number of language patterns that are to be looked for. Indeed, the ones listed in this paper certainly do not encompass all of the patterns that can be observed in computer languages. Since language patterns are still the subject of ongoing research, we expect to identify and formalize more of them in the future.

## ACKNOWLEDGMENT

This work was supported by VEGA Grant No. 1/0305/11 Co-evolution of the artifacts written in domain-specific languages driven by language evolution.

## REFERENCES

- [1] M. Črepinšek, M. Mernik, B. R. Bryant, F. Javed, and A. Sprague, "Inferring Context-Free Grammars for Domain-Specific Languages," *Electronic Notes in Theoretical Computer Science (ENTCS)*, vol. 141, pp. 99–116, December 2005. [Online]. Available: <http://dx.doi.org/10.1016/j.entcs.2005.02.055>
- [2] P. Dupont, "Regular Grammatical Inference from Positive and Negative Samples by Genetic Search: the GIG Method," in *Proceedings of the Second International Colloquium on Grammatical Inference and Applications*. London, UK: Springer-Verlag, 1994, pp. 236–245. [Online]. Available: <http://portal.acm.org/citation.cfm?id=645515.658234>
- [3] M. Mernik, D. Hrcic, B. Bryant, A. Sprague, J. Gray, Q. Liu, and F. Javed, "Grammar inference algorithms and applications in software engineering," in *Information, Communication and Automation Technologies, 2009. ICAT 2009. XXII International Symposium on*, 2009, pp. 1–7.
- [4] J. Porubán, M. Forgáč, M. Sabo, and M. Běhálek, "Annotation Based Parser Generator," *Computer Science and Information Systems*, vol. 7, no. 2, pp. 291–307, 2010.
- [5] K. J. Lang, B. A. Pearlmutter, and R. A. Price, "Results of the Abbadingo One DFA Learning Competition and a New Evidence-Driven State Merging Algorithm," in *Proceedings of the 4th International Colloquium on Grammatical Inference*. London, UK: Springer-Verlag, 1998, pp. 1–12. [Online]. Available: <http://portal.acm.org/citation.cfm?id=645517.655780>
- [6] J. Oncina and P. Garcia, "Inferring regular languages in polynomial update time," in *Pattern Recognition and Image Analysis*, 1991, pp. 49–61.
- [7] M. Črepinšek, M. Mernik, and V. Žumer, "Extracting Grammar from Programs: Brute Force Approach," *SIGPLAN Not.*, vol. 40, pp. 29–38, April 2005. [Online]. Available: <http://doi.acm.org/10.1145/1064165.1064171>
- [8] P. R. Henriques, M. J. V. Pereira, M. J. A. Var, and A. Pereira, "Automatic Generation of Language-based Tools," in *Electronic Notes in Theoretical Computer Science*, M. van den Brand and R. Laemmel, Eds., vol. 65, no. 3. Elsevier Science Publishers, 2002.
- [9] R. Lämmel and C. Verhoef, "Semi-automatic Grammar Recovery," *Softw. Pract. Exper.*, vol. 31, pp. 1395–1448, December 2001. [Online]. Available: <http://portal.acm.org/citation.cfm?id=569229.569230>
- [10] F. Javed, M. Mernik, J. Gray, and B. R. Bryant, "Mars: A metamodel recovery system using grammar inference," *Information and Software Technology*, vol. 50, pp. 948–968, August 2008. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1379905.1379993>
- [11] S. Cook, G. Jones, K. Stuart, and W. A. Cameron, *Domain-Specific Development with Visual Studio DSL Tools*. Addison-Wesley Professional, 2007.
- [12] R. C. Gronback, *Eclipse Modeling Project: A Domain-Specific Language (DSL) Toolkit*, 1st ed. Addison-Wesley Professional, 2009.
- [13] T. Parr, *The Definitive Antlr Reference: Building Domain-Specific Languages*. Pragmatic Bookshelf, 2007.
- [14] EMFText, <http://www.emftext.org>.
- [15] Xtext, <http://www.eclipse.org/Xtext>.
- [16] D. Lakatoš, J. Porubán, and M. Sabo, "Assisted Software Language Creation using Internal Model," in *Proceedings of the International Conference on Engineering of Modern Electric Systems*, ser. EMES'11, 2011.
- [17] E. Gamma, R. Helm, R. Johnson, and J. M. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1994.
- [18] J. Porubán, M. Sabo, J. Kollár, and M. Mernik, "Abstract Syntax Driven Language Development: Defining Language Semantics through Aspects," in *Proceedings of the International Workshop on Formalization of Modeling Languages*, ser. FML '10. New York, NY, USA: ACM, 2010, pp. 2:1–2:5.
- [19] M. Sabo, "Abstract Syntax Driven Concrete Syntax Recognition," *Journal of Information, Control and Management Systems*, vol. 8, no. 4, pp. 393–402, 2010.
- [20] M. Sabo and J. Porubán, "Concrete Syntax Recognition using Language Patterns," in *Proceedings of CSE 2010 International Scientific Conference on Computer Science and Engineering*, 2010, pp. 101–108.



# Tree Indexing by Pushdown Automata and Subtree Repeats

Tomáš Flouri

Czech Technical University in Prague, Czech Republic,  
 Dept. of Theoretical Computer Science

Costas S. Iliopoulos

King's College London, UK, Dept. of Informatics &  
 Curtin University of Technology, Australia, DEBII

Jan Janoušek

Czech Technical University in Prague,  
 Czech Republic,  
 Dept. of Theoretical Computer Science

Bořivoj Melichar

Czech Technical University in Prague,  
 Czech Republic,  
 Dept. of Theoretical Computer Science

Solon P. Pissis

King's College London, UK,  
 Dept. of Informatics

**Abstract**—We consider the problem of finding all subtree repeats in a given unranked ordered tree. We show a new, elegant, and simple method, which is based on the construction of a tree indexing structure called the *subtree pushdown automaton*. We propose a solution for computing all subtree repeats from the deterministic subtree pushdown automaton constructed over the subject tree. The method we present is directly analogous to the relationship between string deterministic suffix automata and factor repeats in a given string.

## I. INTRODUCTION

TREES are one of the fundamental data structures used in Computer Science. Given a tree, finding beforehand unknown subtree repeats of the tree, and the positions of their occurrences, is a problem with many applications – data compression of trees, compiler code optimization, locating code clones in software packages, analysing various data tree structures, such as XML, and so on.

Periodicity in strings has been of interest since the beginning of the 20th century, and efficient methods for finding various kinds of repetitions and repeats in a string form an important part of well-researched stringology theory [1], [2], [3]. Some of these methods are based on principles of constructing and analysing string suffix trees or string suffix automata, which represent complete indexes of the suffixes of a string [4], [5], [6], [7], [8].

Trees can also be seen as strings in their linear notation. A linear notation of a tree can be obtained by the corresponding traversing of the tree. Moreover, every sequential algorithm on a tree traverses nodes of the tree in a sequential order, and follows a linear notation of the tree. In [9], the authors show that the deterministic pushdown automaton (PDA) is an appropriate model of computation for labelled ordered trees in postfix notation, and that the trees in postfix notation, acceptable by deterministic PDA, form a proper superclass of the class of regular tree languages [10], which are accepted by finite tree automata.

This research has been partially supported by the Ministry of Education, Youth and Sports of Czech Republic under research program MSM 6840770014, and by the Czech Science Foundation as project No. 201/09/0807.

In this article, we present a new, elegant, and simple method for finding all subtree repeats in a given unranked ordered tree, and the positions of their occurrences. This problem can be defined as follows.

*Problem 1:* Find all subtree repeats within a given subject tree  $t$ .

The presented method is based on principles of constructing and analysing the deterministic *subtree pushdown automaton* (SPA), similarly as in the case of the above mentioned methods for finding repeats in strings. The SPA for ranked ordered trees was originally introduced in [11]. The construction of an SPA is based on the fact that the postfix bar notation of each subtree is a factor of the postfix bar notation of the tree. The underlying tree structure is processed by the use of the pushdown store. By analogy with the string factor automaton, the SPA represents a complete index of a given tree for all possible subtrees. Given a tree of size  $n$ , the advantages of the deterministic SPA are:

- Given an input subtree of size  $m$ , the deterministic SPA performs the search phase in time linear in  $m$ , and not depending on  $n$ .
- The number of subtrees of the tree is  $n$  and the total size of the deterministic SPA is linear in  $n$ .

We recall that the problem of tree indexing is defined as follows:

*Problem 2:* Construct an indexing structure over a subject tree  $t$ , so that one can efficiently query whether a given subtree  $p$  exists in  $t$ .

## II. PRELIMINARIES

### A. Basic Definitions

An *alphabet*  $\Sigma$  is a finite, nonempty set of symbols. A *string* is a succession of zero or more symbols from an alphabet  $\Sigma$ . The string with zero symbols is denoted by  $\varepsilon$ . The set of all strings over the alphabet  $\Sigma$  is denoted by  $\Sigma^*$ . A string  $x$  of length  $m$  is represented by  $x_1x_2\dots x_m$ , where  $x_i \in \Sigma$  for  $1 \leq i \leq m$ . The length of a string  $x$  is denoted by  $|x|$ . A string  $w$  is a *factor* of  $x$  if  $x = uwv$  for  $u, v \in \Sigma^*$ , and is represented as  $w = x_i\dots x_j$ ,  $1 \leq i \leq j \leq |x|$ .



The number of nodes of a tree  $t$  is denoted by  $|t|$ . The *postfix bar notation*  $\text{bar}(t)$  of a labeled ordered tree  $t$  is obtained by applying *Step* recursively, starting at the root of  $t$ .

*Step*: Let this application of *Step* be node  $v$ . List a bar. If  $v$  is a leaf, list  $v$  and halt. If  $v$  is an internal node having descendants  $v_1, v_2, \dots, v_r$ , apply *Step* to  $v_1, v_2, \dots, v_r$  in that order and then list  $v$ . For simplicity, throughout the article we will refer to the postfix bar notation as bar notation.

An (extended) *nondeterministic pushdown automaton* is a seven-tuple  $M = (Q, \mathcal{A}, G, \delta, q_0, Z_0, F)$ , where  $Q$  is a finite set of *states*,  $\mathcal{A}$  is the *input alphabet*,  $G$  is the *pushdown store alphabet*,  $\delta$  is a mapping from  $Q \times (\mathcal{A} \cup \{\varepsilon\}) \times G^*$  into a set of finite subsets of  $Q \times G^*$ ,  $q_0 \in Q$  is the initial state,  $Z_0 \in G$  is the initial content of the pushdown store, and  $F \subseteq Q$  is the set of final (accepting) states. The triplet  $(q, w, x) \in Q \times \mathcal{A}^* \times G^*$  denotes the configuration of a PDA. In this article, we write the top of the pushdown store  $x$  on its left hand side. The initial configuration of a PDA is a triplet  $(q_0, w, Z_0)$  for the input string  $w \in \mathcal{A}^*$ . The relation  $\vdash_M \subset (Q \times \mathcal{A}^* \times \Gamma^*) \times (Q \times \mathcal{A}^* \times \Gamma^*)$  is a transition of a PDA  $M$ . It holds that  $(q, aw, \alpha\beta) \vdash_M (p, w, \gamma\beta)$  if  $(p, \gamma) \in \delta(q, a, \alpha)$ . For simplicity, in the rest of the text, we use the notation  $p\alpha \xrightarrow[M]{a} q\beta$  when referring to the transition  $\delta_1(p, a, \alpha) = (q, \beta)$  of a PDA  $M$ . A PDA is *deterministic*, if:

- 1)  $|\delta(q, a, \gamma)| \leq 1$  for all  $q \in Q, a \in \mathcal{A} \cup \{\varepsilon\}, \gamma \in G^*$ .
- 2) If  $\delta(q, a, \alpha) \neq \emptyset, \delta(q, a, \beta) \neq \emptyset$  and  $\alpha \neq \beta$  then  $\alpha$  is not a suffix of  $\beta$  and  $\beta$  is not a suffix of  $\alpha$ .
- 3) If  $\delta(q, a, \alpha) \neq \emptyset, \delta(q, \varepsilon, \beta) \neq \emptyset$ , then  $\alpha$  is not a suffix of  $\beta$  and  $\beta$  is not a suffix of  $\alpha$ .

A language  $L$  accepted by a PDA  $M$  is defined in two distinct ways:

- 1) Accepted by final state:  $L(M) = \{x : \delta(q_0, x, Z_0) \vdash_M^* (q, \varepsilon, \gamma) \wedge x \in \mathcal{A}^* \wedge \gamma \in \Gamma^* \wedge q \in F\}$
- 2) Accepted by empty pushdown store:  $L_\varepsilon(M) = \{x : (q_0, x, Z_0) \vdash_M^* (q, \varepsilon, \varepsilon) \wedge x \in \mathcal{A}^* \wedge q \in Q\}$

In the rest of the text, we use the following labeling of edges when illustrating transition diagrams of various PDA: For each transition rule  $\delta_1(p, a, \alpha) = (q, \beta)$  from the transition mapping  $\delta$  of a PDA, we label its edge leading from state  $p$  to state  $q$  by the triplet of the form  $a|\alpha \mapsto \beta$ .

### B. Properties of unranked ordered trees in bar notation

In this section, we present some basic properties of trees in their bar notation.

*Lemma 3*: Given a tree  $t$  and its bar notation  $\text{bar}(t)$ , the bar notations of all subtrees of  $t$  are factors of  $\text{bar}(t)$ .

However, not every factor of the bar notation of a tree represents a subtree.

*Definition 4*: Let  $x = x_1x_2 \dots x_m, m \geq 1$ , be a string over an alphabet  $\Sigma$ . Then, the *bar checksum*  $bc(x) = \sum_{i=1}^m b(x_i)$ , where

$$b(x_i) = \begin{cases} 1 & : x_i = | \\ -1 & : x_i \in \Sigma \end{cases}$$

*Theorem 5*: Let  $\text{bar}(t)$  and  $x$  be a tree  $t$  in bar notation and a factor of  $\text{bar}(t)$ , respectively, over an alphabet  $\Sigma$ . Then,  $x$

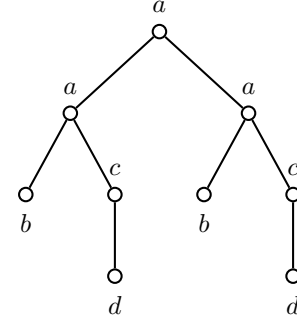


Fig. 2: An unranked ordered tree  $t$  having bar notation  $\text{bar}(t) = |||b||dca||b||dca$

**Input:** Tree  $x = x_1x_2 \dots x_n$  over  $\Sigma$

**Output:** Nondeterministic SPA  $M$

- 1:  $Q \leftarrow \bigcup_{i \leftarrow 0}^n \{i\}$
- 2:  $B \leftarrow \bigcup_{i \leftarrow 1}^n \{i : x_i = |\}$
- 3:  $C \leftarrow \bigcup_{i \leftarrow 1}^n \{i : x_i \neq |\}$
- 4:  $T \leftarrow \bigcup \{(0, \varepsilon, |, S, i) : \forall i \in B\} \cup \bigcup \{(0, S, x_i, \varepsilon, i) : \forall i \in C\} \cup \bigcup \{(i-1, \varepsilon, |, S, i) : \forall i \in B\} \cup \bigcup \{(i-1, S, |, \varepsilon, i) : \forall i \in C\}$
- 5:  $\delta(p, \alpha, x) \leftarrow \{(q, \beta) : \forall (p, \alpha, x, q, \beta) \in T\}$
- 6:  $M \leftarrow (Q, \{S\}, \Sigma, \delta, 0, \varepsilon, \emptyset)$

Fig. 3: Construction of a nondeterministic SPA

is the bar notation of a subtree of  $t$ , if and only if  $bc(x) = 0$ , and  $bc(y) < 0$  for each  $y$ , where  $x = zy, y, z \in \Sigma^+$ .

## III. TREE INDEXING BY PUSHDOWN AUTOMATA

### A. Subtree Pushdown Automaton

The SPA represents a complete index of some tree  $t$ . The language accepted by such automaton is the set of linearised notations of all subtrees of  $t$  – in this case bar notations – and is accepted by an empty pushdown store.

The construction of the nondeterministic SPA is similar to the construction of the classical nondeterministic string suffix automaton (see [2]) and is presented in Fig. 3. However, the transformation to its equivalent deterministic version differs substantially from that of the suffix automaton, as more constraints are placed due to the fact that not every factor of the linearised notation of a tree is a subtree (see Theorem 5).

*Example 6*: Consider the tree  $t$  illustrated in Fig. 2, having bar notation  $\text{bar}(t) = |||b||dca||b||dca$ . The nondeterministic SPA constructed using the algorithm in Fig. 3, is  $M = (\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18\}, \Sigma, \{S\}, \delta, 0, \varepsilon, \emptyset)$ , illustrated in Fig. 1.

### B. Construction of deterministic SPA

We are now in a position to present a simple method for constructing a deterministic SPA to solve Problem 2, by using a subset construction method, transforming the nondeterministic SPA to a deterministic one.

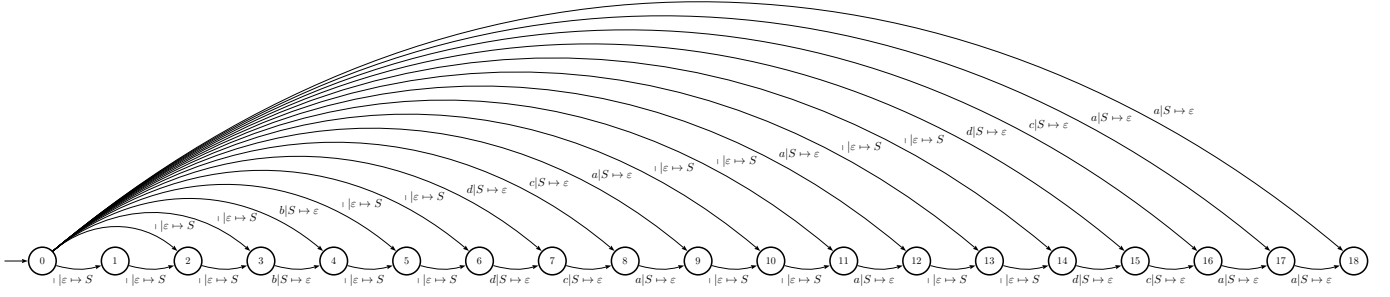


Fig. 1: Non-deterministic SPA constructed from Example 6

**Input:** Nondeterministic SPA  $M = (Q, \Sigma, \{S\}, \delta, 0, \varepsilon, \emptyset)$

**Output:** Deterministic SPA  $M' = (Q', \Sigma, \{S\}, \delta', \{0\}, \varepsilon, \emptyset)$

```

1:  $Q' \leftarrow \{\{0\}\}$ 
2:  $L \leftarrow \{(E, 1) : \delta(0, \varepsilon, |) = E \times \{S\}\}$ 
3: while not empty  $L$  do
4:    $(q, bc) \leftarrow \text{DEQUEUE}(L)$ 
5:    $Q' \leftarrow Q' \cup \{q\}$ 
6:   for all  $x \in \Sigma$  do
7:      $q_x \leftarrow \{E : \delta(p, S, x) = E \times \{\varepsilon\}, \forall p \in q\}$ 
8:     if  $q_x \neq \emptyset$  then
9:        $Q' \leftarrow Q' \cup \{q_x\}$ 
10:       $\delta'(q, S, x) \leftarrow (q_x, \varepsilon)$ 
11:      if  $bc > 1$  then
12:         $\text{ENQUEUE}(L, (q_x, bc - 1))$ 
13:      end if
14:    end if
15:  end for
16:   $q_l \leftarrow \{E : \delta(p, \varepsilon, x) = E \times \{S\}, \forall p \in q\}$ 
17:  if  $q_l \neq \emptyset$  then
18:     $Q' \leftarrow Q' \cup \{q_l\}$ 
19:     $\delta'(q, \varepsilon, |) \leftarrow (q_l, S)$ 
20:     $\text{ENQUEUE}(L, (q_x, bc + 1))$ 
21:  end if
22: end while
23:  $M' \leftarrow (Q', \Sigma, \{S\}, \delta', \{0\}, \varepsilon, \emptyset)$ 

```

Fig. 4: Transforming a nondeterministic SPA to an equivalent deterministic

Fig. 4 presents the algorithm for the transformation, which is based on the well-known technique of subset construction [12]. The nondeterministic SPA constructed from Alg. 3 is input-driven, and thus can be transformed to an equivalent deterministic SPA, which will serve as the indexing structure for the given subject tree.

*Example 7:* Consider the nondeterministic SPA constructed in Example 6. By applying the algorithm in Fig. 4, we obtain a new deterministic SPA with its transition diagram illustrated in Fig. 5.

*Theorem 8:* Given a tree  $t$  of size  $n$ , the deterministic SPA constructed from  $\text{bar}(t)$  is input-driven, has exactly one pushdown symbol, and consists of at most  $4n + 1$  states.

## IV. FINDING SUBTREE REPEATS

### A. Definition of the problem

Problem 1 is to find all subtree repeats of a given tree  $t$ , along with the positions and the types of their occurrences. The positions and the types of the occurrences are summarised in a table called the *subtree repeats table*.

*Definition 9:* Let  $t$  be a tree over an alphabet  $\Sigma$ . A *subtree position set*  $\text{sps}(s, t)$ , where  $s$  is a subtree of  $t$ , is the set  $\text{sps}(s, t) = \{i : \text{bar}(t) = x \text{bar}(s) y, x, y \in (\Sigma \cup \{\})^*, i = |x| + |\text{bar}(s)| + 1\}$ .

Informally, the subtree position set for a subtree  $s$  contains the positions of the roots of all occurrences of the subtree  $s$ .

*Example 10:* Consider the tree  $t$  illustrated in Fig. 2. There are four subtree repeats  $t_1, t_2, t_3$  and  $t_4$  in  $t$  having bar notations  $\text{bar}(t_1) = ||b||dca$ ,  $\text{bar}(t_2) = ||dc$ ,  $\text{bar}(t_3) = |d$  and  $\text{bar}(t_4) = |b$ . It holds that  $\text{sps}(t_1, t) = \{9, 17\}$ ,  $\text{sps}(t_2, t) = \{8, 16\}$ ,  $\text{sps}(t_3, t) = \{7, 15\}$  and  $\text{sps}(t_4, t) = \{4, 12\}$ .

*Definition 11:* Let  $t$  be a tree over an alphabet  $\Sigma$ . Given a subtree  $s$  of  $t$ , the *list of subtree repeats*  $\text{lsr}(s, t)$  is a relation in  $\text{sps}(s, t) \times \{F, S, G\}$  defined as follows:

- $(i, F) \in \text{lsr}(s, t)$  iff  $\text{bar}(t) = x \text{bar}(s) y$ ,  $i = |x| + |\text{bar}(s)|$ ,  $x \neq x_1 \text{bar}(s) x_2$ ,
- $(i, S) \in \text{lsr}(s, t)$  iff  $\text{bar}(t) = x \text{bar}(s) y$ ,  $i = |x| + |\text{bar}(s)|$ ,  $x = x_1 \text{bar}(s)$ ,
- $(i, G) \in \text{lsr}(s, t)$  iff  $\text{bar}(t) = x \text{bar}(s) y$ ,  $i = |x| + |\text{bar}(s)|$ ,  $x = x_1 \text{bar}(s) x_2$

where  $x, y, x_1, x_2 \in \Sigma^*$ .

In other words, the list of subtree repeats can be categorised in three types.  $F$  denotes that the subtree at the specific position is the *first* subtree in the list.  $S$  denotes a *square*, i.e. the specified subtree is a sibling of its previous subtree repeat, and thus their bar notations are consecutive factors in the bar notation of the subject tree.  $G$  stands for *gap*, and denotes that there exists a gap – another different subtree – between its previous repeat. In comparison with the types of repeats found in strings (see [2], [8]), subtree repeats are missing the type *overlapping*, as no two different occurrences of the same subtree can overlap.

*Definition 12:* Given a tree  $t$ , the *subtree repeats table*  $\text{SRT}(t)$  is the set of all triplets  $(\text{sps}(s, t), \text{bar}(s), \text{lsr}(s, t))$ , where  $s$  is a subtree with more than one occurrence in  $t$ .

*Example 13:* Consider the tree  $t$  illustrated in Fig. 2. The subtree repeats table  $\text{SRT}(t)$  is illustrated in Table I.

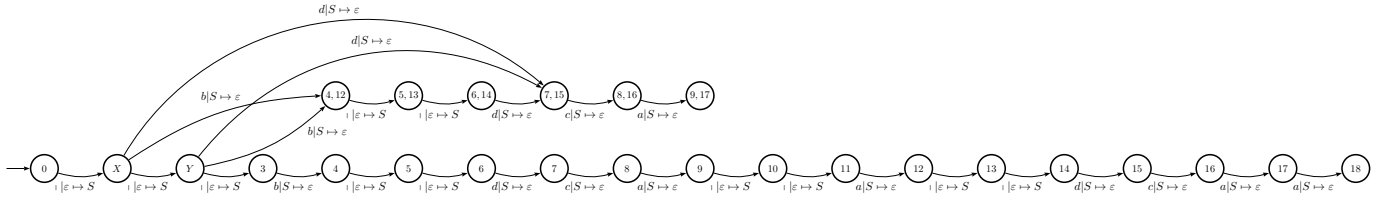


Fig. 5: Deterministic SPA from Example 7. Note that  $X$  denotes the state  $\{1, 2, 35, 6, 10, 11, 13, 14\}$  and  $Y$  the state  $\{2, 3, 6, 11, 14\}$

**Input:** A tree  $t$  in bar notation  $bar(t) = x_1x_2 \dots x_n$

**Output:** Subtree repeats table  $SRT(t)$

- 1: Initialise  $SRT(t) = \emptyset$
- 2: Construct a deterministic SPA  $M = (Q, \Sigma, \{S\}, \delta, \{0\}, \varepsilon, \emptyset)$  using the algorithms in Fig. 3 and 4
- 3: ENQUEUE( $L, \{\{0\}, \varepsilon, 0\}$ )
- 4: **while** notempty  $L$  **do**
- 5:    $(q, x, bc) \leftarrow$  DEQUEUE( $L$ )
- 6:   **for all**  $y \in \Sigma$  **do**
- 7:     **if**  $\delta(q, S, y) \neq \emptyset$  **then**
- 8:       ENQUEUE( $L, \{p, xy, bc - 1\} : \delta(q, S, y) = (p, \varepsilon)$ )
- 9:     **end if**
- 10:   **end for**
- 11:   **if**  $\delta(q, \varepsilon, |) \neq \emptyset$  **then**
- 12:     ENQUEUE( $L, \{p, x|, bc + 1\} : \delta(q, \varepsilon, y) = (p, S)$ )
- 13:   **end if**
- 14:   **if**  $bc = 0 \wedge |x| > 0 \wedge |q| > 1$  **then**
- 15:     Let us denote that  $q = \{r_1, r_2, \dots, r_{|q|}\}$  such that  $r_i > r_{i-1}$
- 16:      $lsr(x, t) \leftarrow \{(r_1, F)\}$
- 17:     **for all**  $r_i \in q, i > 1$  **do**
- 18:       **if**  $r_i - r_{i-1} = |x|$  **then**
- 19:          $lsr(x, t) \leftarrow lsr(x, t) \cup \{(i, S)\}$
- 20:       **else**  $\{r_i - r_{i-1} > |x|\}$
- 21:          $lsr(x, t) \leftarrow lsr(x, t) \cup \{(i, G)\}$
- 22:       **end if**
- 23:     **end for**
- 24:   **end if**
- 25:    $SRT(t) \leftarrow SRT(t) \cup \{(i, y) : (i, y) \in lsr(x, t)\}, x, lsr(x, t)\}$
- 26: **end while**

Fig. 6: Construction of the subtree repeats table for a tree  $t$

$sps(s, t)$	$bar(s)$	List of subtree repeats
9, 17	b dca	(9, F), (17, S)
8, 16	dc	(8, F), (16, G)
7, 15	d	(7, F), (15, G)
4, 12	b	(4, F), (12, G)

TABLE I: Subtree repeats table  $SRT(t)$  from Example 13

### B. Construction of the subtree repeats table

A well-known general property of the deterministic string suffix automaton constructed for a string  $x$  is that the state in which the deterministic string suffix automaton transits after

reading a factor  $y$ , corresponds to the set of ending positions of all occurrences of the factor  $y$  in  $x$  [8].

The transitions of the deterministic SPA  $M$  constructed by the algorithm in Fig. 4 for some tree  $t$  are extensions of the transitions of the deterministic string suffix automaton [11], and therefore the same general property also holds for the SPA  $M$ : the state in which the SPA  $M$  transits after reading the postfix notation  $x$  of some subtree  $s$ , corresponds to the set of ending positions of all occurrences of  $x$  in  $bar(t)$ , which, in turn, corresponds to the positions of the root nodes of all occurrences of  $s$  in  $t$ .

The deterministic SPA  $M$  accepts bar notations of all subtrees of  $t$  by the empty pushdown store. The non-singleton subset of some state  $q$  having its  $bc$  set to 0, which is represented by the empty pushdown store, denotes the positions of the subtree read to transit from the initial state of  $M$  to state  $q$ . The subtree repeats table can therefore be constructed by traversing the deterministic SPA  $M$  and is described by the algorithm in Fig. 6.

### REFERENCES

- [1] M. Crochemore and W. Rytter, *Jewels of Stringology*. New Jersey: World Scientific, 1994.
- [2] B. Melichar, J. Holub, and J. Polcar, "Text searching algorithms," Available on: <http://stringology.org/athens/>, 2005, release November 2005.
- [3] B. Smyth, *Computing Patterns in Strings*. Essex, England: Addison-Wesley-Pearson Education Limited, 2003.
- [4] A. Apostolico and F. P. Preparata, "Optimal off-line detection of repetitions in a string," *Theor. Comput. Sci.*, vol. 22, pp. 297–315, 1983.
- [5] G. S. Brodal, R. B. Lyngsø, C. N. S. Pedersen, and J. Stoye, "Finding maximal pairs with bounded gap," in *CPM*, ser. Lecture Notes in Computer Science, M. Crochemore and M. Paterson, Eds., vol. 1645. Springer, 1999, pp. 134–149.
- [6] M. Crochemore, "An optimal algorithm for computing the repetitions in a word," *Inf. Process. Lett.*, vol. 12, no. 5, pp. 244–250, 1981.
- [7] M. G. Main and R. J. Lorentz, "An  $o(n \log n)$  algorithm for finding all repetitions in a string," *J. Algorithms*, vol. 5, no. 3, pp. 422–432, 1984.
- [8] B. Melichar, "Repetitions in text and finite automata," in *Proceedings of the Eindhoven FASTAR Days 2004*, L. Cleophas and B. Watson, Eds., TU Eindhoven, The Netherlands, 2004, pp. 1–46.
- [9] J. Janoušek and B. Melichar, "On regular tree languages and deterministic pushdown automata," *Acta Inf.*, vol. 46, no. 7, pp. 533–547, 2009.
- [10] F. Gecseg and M. Steinby, "Tree languages," in *Handbook of Formal Languages*, G. Rozenberg and A. Salomaa, Eds. Springer-Verlag, Berlin, 1997, vol. 3 Beyond Words. Handbook of Formal Languages, pp. 1–68.
- [11] J. Janoušek, "String suffix automata and subtree pushdown automata," in *Proceedings of the Prague Stringology Conference 2009*, J. Holub and J. Ždarek, Eds., Czech Technical University in Prague, Czech Republic, 2009, pp. 160–172, available on: <http://www.stringology.org/event/2009>.
- [12] A. V. Aho and J. D. Ullman, *The theory of parsing, translation, and compiling*. Prentice-Hall Englewood Cliffs, N.J., 1972.

# Subtree Oracle Pushdown Automata for Ranked and Unranked Ordered Trees

Martin Plicka, Jan Janoušek, Bořivoj Melichar  
 martin.plicka@fit.cvut.cz, jan.janoušek@fit.cvut.cz, borivoj.melichar@fit.cvut.cz  
 Department of Theoretical Computer Science,  
 Faculty of Information Technology,  
 Czech Technical University in Prague,  
 Thákurova 9, 160 00 Prague 6,  
 Czech Republic

**Abstract**—Oracle modification of subtree pushdown automata for ranked and unranked ordered trees is presented. Subtree pushdown automata [1] represent a complete index of a tree for subtrees. Subtree oracle pushdown automata, as inspired by string factor oracle automaton [2], have the number of states equal to  $n + 1$ , where  $n$  is the length of a corresponding linear notation of the tree. This makes the space complexity very low. By analogy with the string factor oracle automaton the subtree oracle automata can also accept some subtrees which are not present in the given subject tree. However, the number of such false positive matches is smaller than in the case of the string factor oracle automaton. The presented pushdown automata are input-driven and therefore they can be determined.

## I. INTRODUCTION

TREES are one of the fundamental data structures used in Computer Science, e.g. they are used as intermediate forms in compilers or in the form of XML documents. Trees can also be seen as strings, for example in their prefix (also called preorder) or postfix (also called postorder) notation.

One of basic approaches to pattern matching uses data structures which are constructed for a given subject and represent its index. Examples of such data structures for a subject string can be suffix or factor automata [3], [4], [5]. The main advantage of this kind of deterministic finite automata are that they perform the search phase in time linear in  $m$  and not depending on  $n$ , where  $m$  and  $n$  are the length of the input pattern and of the subject string, respectively. However, the implementation of the string factor automaton requires a fairly large amount of memory space. Factor oracle automaton [2] represents a space reduced variant of the string factor automaton. The number of states of factor oracle automaton is equal to  $n + 1$ , where  $n$  is the length of the subject string. In comparison with the string factor automaton, it accepts also some additional subsequences of the subject string. Despite this fact, it can be used for a fast and memory efficient indexing of strings and for backward oracle string matching, in which the factor oracle for a reversed

input pattern is constructed and then it is used for matching the input pattern in a sliding window from right to left [2].

In [1], we have introduced Subtree PDA, a new kind of acyclic PDA for ordered trees, which represents an index of a subject tree for subtrees and is analogous to the string factor automaton and its properties. The construction of a subtree PDA is based on the fact that each subtree in a specific linear notation is a substring of the tree in the linear notation. The underlying tree structure is processed by the use of the pushdown store. In this paper, by analogy with the string processing, we present an oracle modification of the subtree PDA. The deterministic subtree oracle PDA has the number of states equal to  $n + 1$ , where  $n$  is the length of a corresponding linear notation of the tree. It accepts all subtrees of the subject tree and further it may accept also certain subtrees which are not present in the subject tree. Despite this fact, it can be used for a fast and memory efficient indexing of trees, mainly for quick rejection of input patterns that are not subtrees of the subject tree, and for backward oracle subtree matching, which can be done by analogy with the string backward oracle matching algorithm [2]. We present subtree oracle PDAs for both ranked and unranked ordered trees. Although searching thoroughly, we have not found any other existing indexing structure for a tree with the abovementioned space complexity.

## II. SOME BASIC NOTIONS AND NOTATIONS

**Definition 1.** The prefix notation  $pref(t)$  of a tree  $t$  is defined in this way:

- 1)  $pref(t) = a$  if  $a$  is a leaf,
- 2)  $pref(t) = a pref(b_1) \dots pref(b_n)$ , where  $a$  is the root of the tree  $t$  and  $b_1, \dots, b_n$  are direct descendants of  $a$ .

For unranked trees, nodes have no arity so it is not possible to determine the number of node descendants from label. Instead, a bar notation defined below can be used.

**Definition 2.** Let  $]$  be the prefix bar symbol,  $] \notin \mathcal{A}$ . Prefix bar notation  $pref_{bar}(t)$  of a tree  $t$  is defined as follows:

- 1)  $pref_{bar}(t) = a]$  for tree  $t$  with a single node  $a$ .

This research has been partially supported by the Czech Ministry of Education, Youth and Sports under research program MSMT 6840770014, and by the Czech Science Foundation as project No. 201/09/0807.

2)  $pref_{bar}(t) = r pref_{bar}(a_1), \dots, pref_{bar}(a_n)]$  for tree with root  $r$  and direct descendants  $a_1 \dots a_n$  of node  $r$ .

**Example 1.** Consider a ranked alphabet  $\mathcal{A} = \{b_2, a_2, b_0, a_0\}$ . Consider a tree  $t_1$  in prefix notation  $pref(t_1) = b_2 b_0 a_2 a_0 a_2 a_0 a_0$ . Tree  $t_1$  is illustrated in Fig. 1.  $\square$

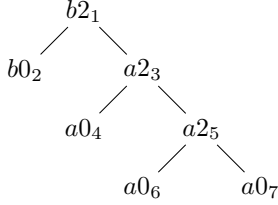


Fig. 1. Tree  $t_1$  from Example 1

Assuming tree  $t_1$  being unranked ( $\mathcal{A} = \{a, b\}$ ), it can be written in prefix bar notation as

$$pref_{bar}(t_1) = b b ] a a ] a a ] a ] ] ] ] \quad \square$$

#### A. Factor oracle automata in stringology

Given a string  $x$ , the deterministic factor automaton is defined as the minimal deterministic finite automaton accepting all factors of  $x$ . A factor oracle automaton can be constructed from the factor automaton. This construction is based on merging so-called *corresponding states* in a factor automaton together [5].

**Definition 3.** Let  $M$  be a factor automaton for string  $x$  and  $q_1, q_2$  be different states of  $M$ . Let there exist two sequences of transitions in  $M$ :  $(q_0, x_1) \vdash^* (q_1, \varepsilon)$ , and  $(q_0, x_2) \vdash^* (q_2, \varepsilon)$ .

If  $x_1$  is a suffix of  $x_2$  and  $x_2$  is a prefix of  $x$  then  $q_1$  and  $q_2$  are corresponding states.

The factor oracle automaton can be constructed by merging the corresponding states.

**Example 2.** We construct factor oracle automaton for the prefix notation  $pref(t_1) = b_2 b_0 a_2 a_0 a_2 a_0 a_0$  of tree  $t_1$  from Example 1. After merging all pairs of corresponding states, the resulting automaton will be  $FM_{ora}(pref(t_1))$  and its transition diagram is depicted in Fig. 2. For more information on its construction, see [5].

The constructed string factor oracle automaton now accepts string  $x = b_2 b_0 a_2 a_0 a_0$ , which is not a factor of  $pref(t_1) = b_2 b_0 a_2 a_0 a_2 a_0 a_0$ . String  $x$  was created from  $pref(t_1)$  by omitting one repeat of string  $a_2 a_0$ .  $\square$

### III. PROPERTIES OF SUBTREES IN PREFIX AND PREFIX BAR NOTATION

In this section we present some general properties of the prefix and the prefix bar notation of a tree.

**Theorem 1.** Given a tree  $t$  and its notation  $pref(t)$  and  $pref_{bar}(t)$ , all subtrees of  $t$  in prefix and prefix bar notation are substrings of  $pref(t)$  and  $pref_{bar}(t)$ , respectively.

*Proof:* See [1], [6], [7].  $\blacksquare$

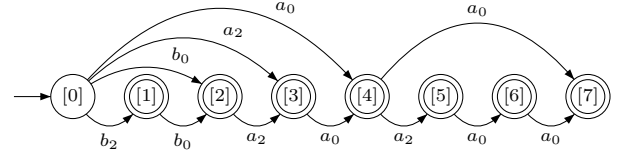


Fig. 2. Transition diagram of the deterministic string factor oracle automaton  $FM_{ora}(pref(t_1))$  for prefix notation  $pref(t_1) = b_2 b_0 a_2 a_0 a_2 a_0 a_0$  of tree  $t_1$  from Example 2

However, not every substring of  $pref(t)$  or  $pref_{bar}(t)$  is a prefix notation of a subtree. This property is formalised by the following definitions and theorems.

**Definition 4.** Let  $w = a_1 a_2 \dots a_m$ ,  $m \geq 1$ , be a string over a ranked alphabet  $\mathcal{A}$ . Then, the arity checksum  $ac(w) = \text{arity}(a_1) + \text{arity}(a_2) + \dots + \text{arity}(a_m) - m + 1 = \sum_{i=1}^m \text{arity}(a_i) - m + 1$ .

**Theorem 2.** Let  $pref(t)$  and  $w$  be a tree  $t$  in prefix notation and a substring of  $pref(t)$ , respectively. Then,  $w$  is the prefix notation of a subtree of  $t$ , if and only if  $ac(w) = 0$ , and  $ac(w_1) \geq 1$  for each  $w_1$ , where  $w = w_1 x$ ,  $x \neq \varepsilon$ .

*Proof:* See [1], [6], [7].  $\blacksquare$

Similar properties can be seen for unranked trees and their prefix bar notation.

**Definition 5.** Let  $w = a_1 a_2 \dots a_m$ ,  $m \geq 1$ , be a string over  $\mathcal{A} \cup \{\}\}$ . Then, the bar checksum is defined as follows:

- 1)  $bc(a) = 1$  and  $bc(\}) = -1$ .
- 2)  $bc(wa) = bc(w) + 1$  and  $bc(w]) = bc(w) - 1$ .

**Lemma 1.** Let  $w, w = b_1 b_2 \dots b_m$ ,  $m \geq 2$  be a string over alphabet  $\mathcal{A} \cup \{\}$  such that  $bc(w) = 0$ , and  $bc(w_1) \geq 1$  for each  $w_1$ , where  $w = w_1 x$ ,  $x \neq \varepsilon$ . Then  $b_1 \in \mathcal{A}$  and  $b_m = \}$ .

*Proof:* See [7].  $\blacksquare$

**Theorem 3.** Let  $pref_{bar}(t)$  and  $w$  be a tree  $t$  in bar notation and a substring of  $pref_{bar}(t)$ , respectively. Then,  $w$  is the bar notation of a subtree of  $t$ , if and only if  $bc(w) = 0$ , and  $bc(w_1) \geq 1$  for each  $w_1$ , where  $w = w_1 x$ ,  $x \neq \varepsilon$ .

*Proof:* See [7].  $\blacksquare$

**Theorem 4.** Let  $M = (\{Q, \mathcal{A}, \{S\}, \delta, 0, S, \emptyset)$  be an input-driven PDA of which each transition from  $\delta$  is of the form  $\delta(q_1, a, S) = (q_2, S^i)$ , where  $i = \text{arity}(a)$ .

Then, if  $(q_3, w, S) \vdash_M^+ (q_4, \varepsilon, S^j)$ , then  $j = ac(w)$ .

*Proof:* See [7].  $\blacksquare$

### IV. SUBTREE PUSHDOWN AUTOMATA

#### A. Subtree PDA for trees in prefix notation

A subtree pushdown automaton for ranked ordered trees has been introduced in [1]. Nondeterministic subtree PDA for trees in prefix notation is an input-driven PDA constructed by Alg. 1.

**Algorithm 1.** Construction of a nondeterministic subtree PDA for a tree  $t$  in prefix notation  $pref(t)$ .

**Input:** A tree  $t$ ; prefix notation  $pref(t) = a_1 a_2 \dots a_n$ ,  $n \geq 1$ .

**Output:** Nondeterministic subtree PDA  $M_{nps}(t) = (\{0, 1, 2, \dots, n\}, \mathcal{A}, \{S\}, \delta, 0, S, \emptyset)$ .

**Method:**

- 1) For each state  $i$ , where  $1 \leq i \leq n$ , create a new transition  $\delta(i-1, a_i, S) = (i, S^{Arity(a_i)})$ , where  $S^0 = \varepsilon$ .
- 2) For each state  $i$ , where  $2 \leq i \leq n$ , create a new transition  $\delta(0, a_i, S) = (i, S^{Arity(a_i)})$ , where  $S^0 = \varepsilon$ .  $\square$

Each nondeterministic input-driven PDA can be transformed to an equivalent deterministic input-driven PDA.

**Algorithm 2.** Transformation of an input-driven nondeterministic PDA to an equivalent deterministic PDA.

**Input:** Acyclic input-driven nondeterministic PDA  $M_{nx}(t) = (\{0, 1, 2, \dots, n\}, \mathcal{A}, \{S\}, \delta, 0, S, \emptyset)$ , where the ordering of its states is such that if  $\delta(p, a, \alpha) = (q, \beta)$ , then  $p < q$ .

**Output:** Equivalent deterministic PDA  $M_{dx}(t) = (Q', \mathcal{A}, \{S\}, \delta', q_I, S, \emptyset)$ .

**Method:**

- 1) Initially,  $Q' = \{[0]\}$ ,  $q_I = [0]$ ,  $cpds([0]) = \{S\}$  and  $[0]$  is an unmarked state.
- 2) a) Select an unmarked state  $q'$  from  $Q'$  such that  $q'$  contains the smallest possible state  $q \in Q$ , where  $0 \leq q \leq n$ .  
b) If there is  $S^r \in cpds(q')$ ,  $r \geq 1$ , then for each input symbol  $a \in \mathcal{A}$ :  
i) Add transition  $\delta'(q', a, \alpha) = (q'', \beta)$ , where  $q'' = \{q : \delta(p, a, \alpha) = (q, \beta) \text{ for all } p \in q'\}$ . If  $q''$  is not in  $Q'$  then add  $q''$  to  $Q'$  and create  $cpds(q'') = \emptyset$ . Add  $\omega$ , where  $\delta(q', a, \gamma) \vdash_{M_{dx}(t)} (q'', \varepsilon, \omega)$  and  $\gamma \in cpds(q')$ , to  $cpds(q'')$ .  
c) Set the state  $q'$  as marked.
- 3) Repeat step 2 until all states in  $Q'$  are marked.  $\square$

The deterministic subtree PDA for a tree in prefix notation is demonstrated by the following example.

**Example 3.** The deterministic subtree PDA for tree  $t_1$  in prefix notation from Example 1, which has been constructed by Alg. 2 and then determinised is PDA  $M_{dps}(t_1)$ . Its transition diagram is illustrated in Fig. 3.

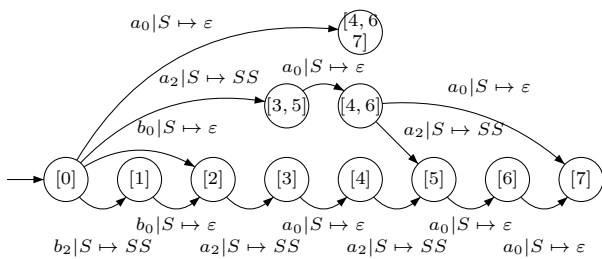


Fig. 3. Transition diagram of deterministic subtree PDA  $M_{dps}(t_1)$  for tree in prefix notation  $pref(t_1) = b_2 b_0 a_2 a_0 a_2 a_0 a_0$  from Example 3

During the processing of an input subtree  $st$  in prefix notation  $pref(st) = a_2 a_0 a_0$ , the automaton changes states

in sequence  $[0], [3], [5], [4], [6]$  and accepts the input in state  $[5]$  by an empty pushdown store.  $\square$

**Theorem 5.** Given a tree  $t$  and its prefix notation  $pref(t)$ , the PDA  $M_{nps}(t)$  constructed by Alg. 1 is a subtree PDA for  $pref(t)$ .

*Proof:* See [1].  $\blacksquare$

**Theorem 6.** Given an acyclic input-driven nondeterministic PDA  $M_{nx}(t) = (Q, \mathcal{A}, \{S\}, \delta, q_0, S, \emptyset)$ , the deterministic PDA  $M_{dx}(t) = (Q', \mathcal{A}, \{S\}, \delta', \{q_0\}, S, \emptyset)$  constructed by Alg. 2 is equivalent to PDA  $M_{nx}(t)$ .

*Proof:* See [7].  $\blacksquare$

### B. Subtree PDA for Prefix Bar Notation

Similarly, we can also construct subtree PDAs for unranked trees in the prefix bar notation.

**Algorithm 3.** Construction of a nondeterministic subtree PDA for a tree  $t$  in prefix bar notation  $pref_{bar}(t)$ .

**Input:** A tree  $t$ ; prefix bar notation  $pref_{bar}(t) = a_1 a_2 \dots a_n$ ,  $n \geq 2$ .

**Output:** Nondeterministic subtree PDA  $M_{npbs}(t) = (\{0, 1, 2, \dots, n\}, \mathcal{A} \cup \{\bar{\cdot}\}, \{S\}, \delta, 0, S, \emptyset)$ .

**Method:**

- 1) For each state  $i$ , where  $2 \leq i \leq n$ , create a new transition 
$$\delta(i-1, a_i, S) = \begin{cases} (i, \varepsilon) & \text{for } a_i = \bar{\cdot} \\ (i, SS) & \text{for } a_i \in \mathcal{A}. \end{cases}$$
- 2) For each state  $i$ , where  $1 \leq i \leq n$ , create a new transition  $\delta(0, a_i, S) = (i, S)$ .  $\square$

Again, the nondeterministic input-driven PDA can be transformed to an equivalent deterministic PDA by Algorithm 2.

**Theorem 7.** Given a tree  $t$  and its prefix bar notation  $pref_{bar}(t)$ , the PDA  $M_{npbs}(t)$  constructed by Alg. 3 is a subtree PDA for  $pref_{bar}(t)$ .

## V. SUBTREE ORACLE PDA

This section deals with subtree oracle pushdown automata. Properties of these pushdown automata will be shown on an example. Similarly to stringology, we construct *subtree oracle automaton* with the use of the definition of corresponding states (see Definition 3). For this purpose, we use a string representing the tree in prefix notation as a part of the definition of corresponding states.

**Definition 6.** Let  $M$  be a subtree automaton for tree  $t$ . Let  $q_1, q_2$  be different states of  $M$ . Let there exist two sequences of transitions in  $M$ :  $(q_0, w_1) \vdash^* (q_1, \varepsilon)$ , and  $(q_0, w_2) \vdash^* (q_2, \varepsilon)$ .

If  $w_1$  is a suffix of  $w_2$  and  $w_2$  is a prefix of  $pref(t)$ , then  $q_1$  and  $q_2$  are corresponding states.

**Definition 7.** Let  $M_{dps}(t)$  be deterministic subtree pushdown automaton (PDA) accepting all subtrees of tree  $t$ . Subtree oracle PDA  $M_{ops}(t)$  is a pushdown automaton created from  $M_{dps}(t)$  by merging all corresponding states.

Using our style of node numbering from 0 to  $n$ , it holds that labels of every two mutually corresponding states  $d_{q_1}, d_{q_2}$  have

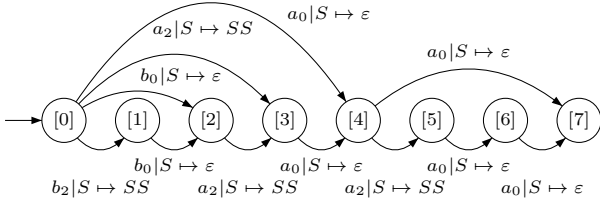


Fig. 4. Transition diagram of deterministic subtree oracle PDA  $M_{ops}(t_1)$  for tree  $t_1$  from Example 4

the same minimal value of their d-subsets, ie.  $\min(d_{q_1}) = \min(d_{q_2})$ .

**Example 4.** We construct the subtree oracle automaton for tree  $t_1$  from Example 1 defined in prefix notation as  $pref(t_1) = b_2 b_0 a_2 a_0 a_2 a_0 a_0$ .

The transition diagram of the deterministic subtree PDA for tree  $t_1$  is illustrated on Fig. 3. We can see three pairs of corresponding states:  $([3], [3, 5])$ ,  $([4], [4, 6])$ ,  $([4], [4, 6, 7])$

To construct subtree oracle PDA, we will merge all pairs of corresponding states. The resulting automaton is deterministic PDA  $M_{ops}(t_1)$ , depicted in Fig. 4.

We see that PDA now accepts  $Pref(t_2) = b_2 b_0 a_2 a_0 a_0$ , which is a subsequence of  $pref(t_1)$ . This property refers to the property of string factor oracle automata (see Example 2).  $\square$

## VI. PROPERTIES OF ORACLE PUSHDOWN AUTOMATA

Using PDAs instead of finite automata may lead to the elimination of some negative properties which are present in string factor oracle automata:

First, during the process of determinisation, the cancellation condition  $cpds(q) = \{\varepsilon\}$  in Algorithm 2 indicates all states from which all outgoing transitions can be omitted because of invalid pushdown operations. This means that some states become inaccessible and can be removed. Omitted transitions would have been responsible for accepting more inputs.

Second, we can define conditions for so-called *safe merging* of corresponding states. This safe merge does not affect the accepted language.

**Definition 8.** We define the minimal input arity checksum of state  $q \in Q$  as  $AC_{min}^-(q) = \min\{i : (q_0, xy, S) \vdash^* (q, y, S^i), x, y \in \mathcal{A}^*, i \geq 0\}$ .

**Definition 9.** We define the minimal input arity checksum of state  $q \in Q$  for input symbol  $a \in \mathcal{A}$  as  $ac_{min}^-(q, a) = \min\{i : (q_0, xay, S) \vdash^* (q, y, S^i), x, y \in \mathcal{A}^*, i \geq 0\}$ .

$AC_{min}^-$  specifies the minimal number of pushdown symbols that can appear in the pushdown store in the state  $q$  after reading any input  $x$ .  $ac_{min}^-$  specifies the last read symbol before reaching the state  $q$  (ie the label of the last transition is used). By replacing  $\min$  function with  $\max$ , we could define  $AC_{max}^-$  and  $ac_{max}^-$ .

**Definition 10.** We define maximal output arity checksum of state  $q \in Q$  as  $AC_{max}^+(q) = \max\{i : (q, x, S^i) \vdash^* (r, \varepsilon, \varepsilon), x \in \mathcal{A}^*, r \in Q, i \geq 0\}$ .

**Definition 11.** We define maximal output arity checksum of state  $q \in Q$  for input symbol  $a \in \mathcal{A}$  as  $ac_{max}^+(q, a) = \max\{i : (q, ax, S^i) \vdash^* (r, \varepsilon, \varepsilon), x \in \mathcal{A}^*, r \in Q, i \geq 0\}$ .

$AC_{max}^+$  specifies the maximal number of pushdown symbols that can be removed starting from the state  $q$  by reading arbitrary input.  $ac_{max}^+$  specifies first symbol of that input. By replacing  $\max$  function with  $\min$ , we could define  $AC_{min}^+$  and  $ac_{min}^+$ .

**Lemma 2.** Two distinct corresponding states  $q$  and  $r$  of deterministic subtree pushdown automata can be safely merged if at least one of following conditions is fulfilled:

- 1) For every input symbol  $a \in \mathcal{A}$  such that  $\delta(r, a, S) \neq \delta(q, a, S)$ , it holds that  $ac_{max}^+(r, a) < AC_{min}^-(q)$  and  $ac_{max}^+(q, a) < AC_{min}^-(r)$
- 2) Either  $AC_{max}^-(q) = AC_{max}^+(q) = 0$  or  $AC_{max}^-(r) = AC_{max}^+(r) = 0$ .

## VII. CONCLUSION

We have described oracle modification of subtree pushdown automata for ordered ranked and unranked trees in prefix and prefix bar notation, respectively. These pushdown automata are analogous in their properties to string factor oracle automata, which are widely used in stringology. The presented pushdown automata allow quick rejection of input patterns that are not subtrees of the subject tree or can be used for efficient backward oracle subtree matching. There are open questions for future research. First, the tree language accepted by the subtree oracle pushdown automaton should be investigated in details. Second, an algorithm for minimising the deterministic subtree PDA using only the safe merging of states, which is introduced in this paper, should also be researched.

For more information on tree algorithms using PDAs, see [8].

## REFERENCES

- [1] J. Janoušek, "String suffix automata and subtree pushdown automata," in *Proceedings of the Prague Stringology Conference 2009*, J. Holub and J. Žďárek, Eds., Czech Technical University in Prague, Czech Republic, 2009, pp. 160–172, available on: <http://www.stringology.org/event/2009>.
- [2] C. Allauzen, M. Crochemore, and M. Raffinot, "Factor oracle: A new structure for pattern matching," in *SOFSEM*, ser. Lecture Notes in Computer Science, J. Pavelka, G. Tel, and M. Bartosek, Eds., vol. 1725. Springer, 1999, pp. 295–310.
- [3] M. Crochemore and C. Hancart, "Automata for matching patterns," in *Handbook of Formal Languages*, G. Rozenberg and A. Salomaa, Eds. Springer-Verlag, Berlin, 1997, vol. 2 Linear Modeling: Background and Application, ch. 9, pp. 399–462.
- [4] M. Crochemore and W. Rytter, *Jewels of Stringology*. New Jersey: World Scientific, 1994.
- [5] B. Melichar, J. Holub, and J. Polcar, "Text searching algorithms," Available on: <http://stringology.org/athens/>, 2005, release November 2005.
- [6] B. Melichar, "Arbology: Trees and pushdown automata," in *LATA*, ser. Lecture Notes in Computer Science, A. H. Dediu, H. Fernau, and C. Martín-Vide, Eds., vol. 6031. Springer, 2010, pp. 32–49, invited paper.
- [7] J. Janoušek, *Arbology: Algorithms on Trees and Pushdown Automata*. Brno: Habilitation thesis, TU FIT, 2010.
- [8] "Arbology www pages," Available on: <http://www.arbology.org/>, July 2011.



# Semi-Automatic Component Upgrade with RefactoringNG

Zdeněk Troníček

Faculty of Information Technology  
 Czech Technical University in  
 Prague  
 Czech Republic  
 Email: tronicek@fit.cvut.cz

**Abstract**—Software components evolve and this evolution often leads to changes in their interfaces. Upgrade to a new version of component then involves changes in client code that are nowadays usually done manually. We deal with the problem of automatic update of client code when the client upgrades to a new version of component. We describe a new flexible refactoring tool for the Java programming language that performs refactorings described by refactoring rules. Each refactoring rule consists of two abstract syntax trees: pattern and rewrite. The tool searches for the pattern tree in client-source-code abstract syntax trees and replaces each occurrence with the rewrite tree. The client-source-code abstract syntax trees are built and fully attributed by the Java compiler. Thus, the tool has complete syntactic and semantic information. Semantic analysis and flexibility in refactoring definitions make the tool superior to most competitors.

## I. INTRODUCTION

EVOLUTION of components often leads to changes in their interfaces (API – application programming interface). If a new version of API is not source compatible with the old version, usually both old and new versions of API are maintained in parallel, so that clients that compiled against the old version can compile against the new version as well. For example, if method `enable()` in version 1.0 evolves to method `setEnabled(boolean b)` in version 2.0, both methods will probably be present in version 2.0 and method `enable()` will be marked deprecated. Upgrade from version 1.0 to version 2.0 then involves changes in client code. These changes are nowadays usually done manually which is tedious and error-prone.

In this paper, we describe a new refactoring tool for the Java programming language that enables automatic update of client code so that it compiles against a new API. Upgrade to a new component is not fully automatic because the tool expects refactoring rules that are assumed to be written by component author. Although authoring rules is not easy, the time spent with them may pay off because once we have the rules, we can upgrade thousands of clients in very straightforward way.

<sup>□</sup>This work has been supported by research program MSM6840770014.

The rest of the paper is structured as follows: section II introduces informally the rule language, section III describes API changes and corresponding refactoring rules, section IV discusses shortcomings of the tool, section V compares the tool with competitors, and section VI concludes.

## II. RULE LANGUAGE

In this section, we introduce the rule language in which we describe source code transformation. Rather than stating the exact syntax, we introduce the language informally on examples.

A refactoring rule defines transformation of one abstract syntax tree (AST) to another AST. Each has the following form: **Pattern**  $\rightarrow$  **Rewrite**. **Pattern** is an AST in original source code and **Rewrite** is an AST which the original AST will be rewritten to. For example, the rule that rewrites `p = null` to `p = 0` is as follows:

```
Assignment {
  Identifier [name: "p"],
  Literal [kind: NULL_LITERAL]
} ->
Assignment {
  Identifier [name: "p"],
  Literal [kind: INT_LITERAL, value: 0]
}
```

**Pattern** and **Rewrite** have the following structure:

**Tree Attributes Content** (attributes and content are optional). Trees are named as ASTs in Oracle Java Compiler [2,3] and attributes are named as their properties. Attributes are enclosed in [ and ] and are comma-separated. They specify additional information about the tree. For example, in `Literal [kind: NULL_LITERAL]` the `kind` attribute says that literal is the null literal. In **Pattern**, the attributes that are not specified match to any value in source code. For example, `Identifier` means any identifier and `Literal [kind: INT_LITERAL]` means any int literal. In **Rewrite**, the tree must be described completely so that a new tree can be built. For example, each `Identifier` in **Rewrite** must have the `name` attribute.

**Content** is a comma-separated list of children of the given tree node enclosed in { and }. For example,

```
Binary [kind: PLUS] {
```

```

    Literal [kind: INT_LITERAL],
    Literal [kind: INT_LITERAL]
  }

```

is addition of two int literals. Children of a given tree must be of appropriate types and all of them must be specified if the tree has any content. For example, `Binary` always must have two children (operands) if it has any content and either of them must be expression. If content of `Binary` is missing, the operands may have any value in source code. For example, `Binary [kind: MINUS]` means any subtraction.

In place where a tree is expected, any subclass of that tree may be used. For example, operands of `Binary` may be any subclasses of `Expression`:

```

Binary [kind: MULTIPLY] {
  Identifier,
  Literal [kind: INT_LITERAL, value: 0]
}

```

The tree hierarchy is the same as in Oracle Java compiler [2]. Tree attributes may have one or more values. If an attribute has more values, they are separated by `|`. For example,

```
Binary [kind: PLUS | MINUS]
```

is either addition or subtraction.

Each tree in *Pattern* may have the `id` attribute. The value of this attribute must be unique in a given rule and is instrumental to referring to the tree from *Rewrite*. For example,

```

Assignment {
  Identifier [id: p],
  Literal [kind: NULL_LITERAL]
} ->
Assignment {
  Identifier [ref: p],
  Literal [kind: INT_LITERAL, value: 0]
}

```

rewrites `p = null` to `p = 0` where `p` stands for any identifier.

References to attributes are written using `#`. For example, `b#kind` refers to the `kind` attribute of `b`. The attribute reference can be used in *Rewrite* as attribute value.

Lists use the same syntax as generic lists in Java. `List<T>` is a list of elements that are assignable to the `T` type. A list can be used either at the highest level or as part of another tree.

### III. API CHANGES

In this section, we show how RefactoringNG can help with adapting client code to a new API. We identified API changes caused by refactoring that are suitable for RefactoringNG. These changes are: Rename field, Rename method, Move field, Move method, Add method argument, Delete method argument, Reorder method arguments, Add type argument, Reorder type arguments, Delete type argument, Change instance method to static, Change static method to instance, Add annotation element, Delete annotation element, Rename annotation element, Rename annotation type, Delete annotation type, Nest top level type, and Unnest nes-

ted type. For all of them we found refactoring rules that update client code so that it compiles against the updated API. Below we discuss three of these rules.

#### A. Rename field

Let's rename the `x` field in the `Position` class to `dx`. To update the client code, we have to replace each occurrence of `x` with `dx`. The appropriate refactoring rule is as follows:

```

MemberSelect [identifier: "x"] {
  Identifier [id: p,
    instanceof: "component.Position"]
} ->
MemberSelect [identifier: "dx"] {
  Identifier [ref: p]
}

```

#### B. Rename method

Let's rename the `read` method in the `Input` class to `readInt`. The rule that replaces invocations of the `read` method with invocations of the `readInt` method is as follows:

```

MethodInvocation {
  List<Tree> { },
  MemberSelect [identifier: "read"] {
    Identifier [id: p,
      instanceof: "component.Input"]
  },
  List<Expression> { }
} ->
MethodInvocation {
  List<Tree> { },
  MemberSelect [
    identifier: "readInt"] {
    Identifier [ref: p]
  },
  List<Expression> { }
}

```

The argument list (`List<Expression>`) enables us to select a method in case of overloading. For example, to address a method invocation with a single string-literal argument, we use the following argument list:

```

List<Expression> {
  Literal [kind: STRING_LITERAL]
}

```

#### C. Add method argument

Let's add an argument to the one-argument `sail` method in the `Ship` class. The following rule replaces invocations of the one-argument method with the two-argument one:

```

MethodInvocation {
  List<Tree> { },
  MemberSelect [identifier: "sail"] {
    Identifier [id: s,
      instanceof: "component.Ship"]
  },
  List<Expression> [id: args]
} ->
MethodInvocation {
  List<Tree> { },
  MemberSelect [identifier: "sail"] {

```

```

    Identifier [ref: s]
  },
  List<Expression> {
    ListItems [ref: args],
    Literal [kind: INT_LITERAL,
            value: 42]
  }
}

```

When adding a method argument, we are not restricted to the last position in argument list. Using the `begin` and `end` attributes at `ListItems` we can insert a new argument at arbitrary position.

#### IV. EVALUATION

In this section, we discuss each rule stated in previous section and describe its limits and shortcomings.

##### A. Rename field

Although we expect the rule is sufficient in many situations, it has two serious shortcomings. First, it renames references to method `x` as well and even though it is a bad practice to have a field and method with the same name, it is allowed in the Java programming language and thus we must count with it. Second, this rule renames only references of form `p.x` where `p` is a variable and there is no simple way how to address references of form `e.x` where `e` is an expression of the `Position` type or any subtype.

##### B. Rename method

There is no simple way how to address method invocations that have either a string literal or a variable as argument in a single rule. We can address either a string literal by

```

List<Expression> {
  Literal [kind: STRING_LITERAL]
}

```

or a variable by

```

List<Expression> {
  Identifier [
    instanceof: "java.lang.String"]
}

```

but we cannot address both in one rule. So, if we want to update code in both cases, we need two rules.

Even worse situation is when the argument is an expression. There is no simple way how to address a method invocation with argument of specific type. This may cause problems if the method is overloaded:

```

public class Input {
  public int read(int i) { ... }
  public int read(String s) { ... }
}

```

If we want to rename `read(int i)` to `readInt`, there is no simple way how to address all invocations of this `read`. In addition, if the `read` method is declared with `int` and `Integer` arguments as follows:

```

public class Input {
  public int read(int i) { ... }
  public int read(Integer i) { ... }
}

```

there is no way how to distinguish between these two methods. Note that the following list of arguments matches either of them:

```

List<Expression> {
  Identifier [
    instanceof: "java.lang.Integer"]
}

```

Overriding causes problems too. For example, given classes `Input` and `ExtInput`:

```

public class Input {
  public int read() { ... }
}
public class ExtInput extends Input {
  public int readInt() { ... }
}

```

and the rule that renames `read()` to `readInt()`, we may end up with a program that is semantically wrong because we unintentionally redirect a method call to `readInt()` in `ExtInput`.

Another problematic situation is when we call a method only by name:

```

public class ExtInput extends Input {
  public int m() {
    return read();
  }
}

```

There is no simple way how to address such method call.

##### C. Add method argument

The shortcoming here is that the tool does not check existence of a method with the same signature as the resulting method. If such method exists, it may happen that we unintentionally redirect the method call to this method. For example, if client declares the `ExtShip` class as follows:

```

public class ExtShip extends Ship {
  public void sail(int direction,
                  int speed) { ... }
}

```

we may end up with a result that is semantically wrong.

Many of these shortcomings have a common cause: missing check whether the refactoring is valid in client context. This is why we strongly recommend inspecting the code before applying changes. For that purpose, the proposed changes are displayed in the standard NetBeans refactoring window and user may decide which of them they confirm.

Concerning a tool support to facilitate rule definition, RefactoringNG contains generator that for a given source code generates AST in RefactoringNG syntax. The output can be used as a base for a rule definition. The rules can be prepared in context-aware editor.

#### V. RELATED WORKS

Code refactoring is an area that is described extensively in literature. For example, Fowler, Beck, Brant, Opdyke, and

Roberts [1] describe many refactorings in detail. Refactoring is used in many programming languages and programmers usually spend some time doing refactoring every day, either manually or by refactoring tools. For that purpose, every Java IDE offers some kind of refactoring, such as rename or encapsulate, in their menu.

The API evolution has already been investigated too. Dig and Johnson [5] conducted a study of API changes of five frameworks. In all the cases, more than 80% of the API breaking changes were identified as refactorings.

Concerning the projects similar to RefactoringNG, we found the Jackpot project [8] that is part of NetBeans IDE [4]. Jackpot has a simple language for description of code transformation. The language is more intuitive but less expressive than the language in RefactoringNG. For example, in Jackpot, you cannot distinguish between a local variable and a field. In RefactoringNG, we can use the `elementKind` attribute for this.

IntelliJ [10] offers 'Structural search and replace'. You declare here two code fragments: one for searching and one as replacement. In some sense it is similar to RefactoringNG. As for differences, use of Structural search and replace is easier because the code fragments are described in almost pure Java. It also has a few features (e.g. regular expressions) that are not implemented in RefactoringNG. On the other hand, it lacks some RefactoringNG's features (e.g. attributes `elementKind` and `nestingKind`) and does not have batch processing.

As far as we know, no tool for automatic component upgrade is commonly used in Eclipse [11].

The problem of automatic upgrade to a new version of API has been investigated by several researchers. Chow and Notkin [6] describe an approach in which a library author annotates changed library functions with rules. These rules are then used to generate tools that can update client code automatically.

Henkel and Divan [7] describe the CatchUp! tool. The main idea behind the tool is to record and replay refactoring actions. The tool captures refactoring actions when developer evolves API and enables to replay them later. Replaying is used for updating client code. The tool supports only a few low-level refactorings that all can be done in RefactoringNG.

Balaban, Tip, and Fuhrer [12] present a framework for automatic migration between library classes. The framework is implemented as Eclipse plugin and uses a special language for specifying migrations. As for functionality, the framework provides only a subset of RefactoringNG.

Tansey and Tilewich [13] present a tool that infers transformation rules from two versions of a class, one before and one after upgrading. These rules are then used by transformation engine to refactor the application source code. It provides automatic inference of transformation rules and the rule language is more intuitive and readable but less powerful than the language in RefactoringNG.

Nguyen *et al.* [14] present a tool that learns how to adapt the client code to a new API. The tool identifies the API library changes and compares client codes before and after library migration. The comparison serves to identify adaptation patterns that are subsequently applied to other clients. The main limitation of this tool is that it requires a set of source codes that already migrated to a new API.

## VI. CONCLUSION

Evolution of software components often leads to changes in their API. When a new version of API is released, usually both old and new APIs are maintained in parallel so that clients that compile against the old API are not broken when they replace the old component with the new one. This approach to API evolution has two shortcomings: (i) maintaining several versions of API is tedious and (ii) it inhibits API evolution because API designers are restricted by the requirement for backward compatibility. Since the old API must be maintained until all clients migrate to the new API, it is desirable to speed the migration up. In this paper, we described the tool that facilitate code migration to a new API. Although several similar tools exist, none of them is widely used and code changes are usually done manually. This is quite surprising because several researchers already showed that many of these adaptations can be done automatically.

## ACKNOWLEDGMENT

Denis Stepanov deserves thanks for his contribution to RefactoringNG. He attached the project to NetBeans infrastructure and implemented the rule editor.

## REFERENCES

- [1] M. Fowler, K. Beck, J. Brant, W. Opdyke, and D. Roberts. *Refactoring: Improving the design of existing code*. Addison-Wesley, 1999.
- [2] *JSR 199: Java Compiler API*. <http://www.jcp.org/en/jsr/detail?id=199>.
- [3] *JSR 269: Pluggable annotation processing API*. <http://jcp.org/en/jsr/detail?id=269>.
- [4] *NetBeans IDE*. <http://www.netbeans.org>.
- [5] D. Dig and R. Johnson. How do APIs evolve? A story of refactoring. *Journal of Software Maintenance and Evolution: Research and Practice*, Volume 18, Issue 2, pp. 83–107, 2006.
- [6] K. Chow and D. Notkin. Semi-automatic update of applications in response to library changes. *International Conference on Software Maintenance*, pp. 359–368, 1996.
- [7] J. Henkel and A. Diwan. CatchUp!: capturing and replaying refactorings to support API evolution. *International Conference on Software Engineering*, pp. 274–283, 2005.
- [8] *Jackpot project*. <http://wiki.netbeans.org/Jackpot>.
- [9] *RefactoringNG project*. <http://kenai.com/projects/refactoringng>.
- [10] IntelliJ IDE. <http://www.jetbrains.com/idea>.
- [11] Eclipse IDE. <http://www.eclipse.org>.
- [12] I. Balaban, F. Tip, and R. Fuhrer. Refactoring support for class library migration. *OOPSLA*, pp. 265–279, 2005.
- [13] W. Tansey and E. Tilevich. Annotation refactoring: inferring upgrade transformations for legacy applications. *OOPSLA*, pp. 295–312, 2008.
- [14] H. A. Nguyen, T. T. Nguyen, G. Wilson, Jr., A. T. Nguyen, M. Kim, and T. N. Nguyen. A graph-based approach to API usage adaptation. *OOPSLA*, pp. 302–321, 2010.

# Extension of Iterator Traits in the C++ Standard Template Library

Norbert Pataki, Zoltán Porkoláb

Department of Programming Languages and Compilers,  
Eötvös Loránd University  
Pázmány Péter sétány 1/C H-1117 Budapest, Hungary  
Email: {patakino, gsd}@elte.hu

**Abstract**—The C++ Standard Template Library is the flagship example for libraries based on the generic programming paradigm. The usage of this library is intended to minimize classical C/C++ error, but does not warrant bug-free programs. Furthermore, many new kinds of errors may arise from the inaccurate use of the generic programming paradigm, like dereferencing invalid iterators or misunderstanding remove-like algorithms.

In this paper we present typical scenarios, that can cause runtime problems. We emit warnings while these constructs are used without any modification in the compiler. We argue for an extension of the STL's iterator traits in order to emit these warnings. We also present a general approach to emit “customized” warnings. We support the so-called believe-me marks to disable warnings.

## I. INTRODUCTION

THE C++ *Standard Template Library* (STL) was developed by *generic programming* approach [2]. In this way containers are defined as class templates and many algorithms can be implemented as function templates. Furthermore, algorithms are implemented in a container-independent way, so one can use them with different containers [15]. C++ STL is widely-used because it is a very handy, standard library that contains beneficial containers (like list, vector, map, etc.), a lot of algorithms (like sort, find, count, etc.) among other utilities.

The STL was designed to be extensible. We can add new containers that can work together with the existing algorithms. On the other hand, we can extend the set of algorithms with a new one that can work together with the existing containers. Iterators bridge the gap between containers and algorithms [3]. The expression problem [16] is solved with this approach. STL also includes adaptor types which transform standard elements of the library for a different functionality [1].

However, the usage of C++ STL does not guarantee bugfree or error-free code [5]. Contrarily, incorrect application of the library may introduce new types of problems [14].

One of the problems is that the error diagnostics are usually complex, and very hard to figure out the root cause of a program error [17], [18]. Violating requirement of special preconditions (e.g. sorted ranges) is not checked, but results in runtime bugs [6]. A different kind of stickler is that if we have an iterator object that pointed to an element in a container, but the element is erased or the container's memory allocation

has been changed, then the iterator becomes *invalid*. Further reference of invalid iterators causes undefined behaviour [13].

Another common mistake is related to removing algorithms. The algorithms are container-independent, hence they do not know how to erase elements from a container, just relocate them to a specific part of the container, and we need to invoke a specific erase member function to remove the elements physically. Therefore, for example the `remove` algorithm does not actually remove any element from a container [10].

Some of the properties are checked at compilation time. For example, the code does not compile if one uses the `sort` algorithm on a standard list container, because the list's iterators do not offer random accessibility [8]. Other properties are checked at runtime. For example, the standard vector container offers an `at` method which tests if the index is valid and it raises an exception otherwise [12].

Unfortunately, there are still a large number of properties are tested neither at compilation-time nor at run-time. Observance of these properties is in the charge of the programmers. On the other hand, type systems can provide a high degree of safety at low operational costs. As part of the compiler, they discover many semantic errors very efficiently.

Certain containers have member functions with the same names as STL algorithms. This phenomenon has many different reasons, for instance, efficiency, safety, or avoidance of compilation errors. For example, as mentioned, list's iterators cannot be passed to `sort` algorithm, hence code cannot be compiled. To overcome this problem list has a member function called `sort`. List also provides `unique` method. In these cases, although the code compiles, the calls of member functions are preferred to the usage of generic algorithms.

In this paper we argue for an approach that generates warning when the STL is used in an improper way or a better approach is available in certain cases. For example we want to warn the programmer if the `copy` or `transform` algorithm is used without inserter iterators. We argue for an extension of STL's trait type of iterators instead of compiler modification. Algorithms can be overloaded based on this extension and warnings can be triggered in the overloaded versions.

This paper is organized as follows. In section II we present some motivating examples, that can be compiled, but at runtime they can cause problems. Then, in section III we present new properties that should be added to the STL's

iterator traits. In section IV we present an approach to generate “customized” warnings at compilation time. In section V we argue for overloading algorithms on the new traits. In section VI the so-called *believe-me marks* are introduced to disable our specific warnings. Finally, this paper concludes in section VII.

## II. MOTIVATION

STL’s `copy` and `transform` algorithm can be used to copy an input range of objects into a target range. These algorithms neither allocate memory space nor call any specific inserter method while coping elements. They assume that the target has enough, properly allocated elements where they can copy elements with `operator=`. Inserter iterators can enforce to use `push_back`, `push_front` or `insert` method of containers. But these algorithms cannot copy elements into an empty list, for instance. They do not know how to insert elements into the empty container. The following code snippet can be compiled, but it results in an undefined behaviour:

```
std::list<int> li;
std::vector<int> vi;
v.push_back( 3 );

std::copy( vi.begin(),
           vi.end(),
           li.begin() );
```

In our opinion in this case a warning message should be emitted to the programmer, that this construct can be problematic.

## III. NEW TRAITS

Iterators are fundamental elements of the STL. They make connection between containers and algorithms. Iterators iterates through the containers or streams. They are the generalization of pointers, thus pointers also can be used in place of iterators.

Iterators have associated types. An iterator type, for instance, has an associated value type: the type of object that the iterator points to. It also has an associated type to describe the type of difference-based values. Generic algorithms often need to have access to these associated types; an algorithm that takes a pair of iterators, for example, might need to declare a temporary variable whose type is the iterators’ value type. The class `iterator_traits` is a mechanism that allows such declarations. For every iterator type, a corresponding specialization of `iterator_traits` class template shall exist or default implementation works (see below). Another reason also can be mentioned for the usage of traits. It can be used for implementing generic functions as efficient as possible. For example, `distance` or `advance` can fully take advantage of the iterator capabilities, and can run at constant time when random access iterators the taken and run at linear time otherwise.

At this point we extend `iterator_traits` in order to overload STL algorithms on new traits and generate warning

in some of them. First, we write two new types according to copying strategy.

```
class __inserting_iterator_tag {};
class __non_inserting_iterator_tag {};
```

The default `iterator_traits` is extended in the following way:

```
template <class T>
struct iterator_traits
{
    typedef typename T::iterator_category
        iterator_category;
    typedef typename T::value_type
        value_type;
    typedef typename T::difference_type
        difference_type;
    typedef typename T::pointer
        pointer;
    typedef typename T::reference
        reference;
    typedef
        __non_inserting_iterator_tag
        inserter;
};
```

We added one more attribute to default `iterator_traits` which is the copying strategy attribute called `inserter`. The `inserter` is a type alias to either `__inserting_iterator_tag` or `__non_inserting_iterator_tag`.

More traits can be mentioned, too. For instance, `find` and `count` algorithms are suboptimal if it is called on an *associative* container, because the algorithms cannot take advantage of sortedness. Hence, an attribute can be described if a container supports `find` or `count` method. Another attribute can define if a container supports a unique member function, such as `list`.

In this paper we do not deal with *safe* iterators [13]. However, safety can be an orthogonal attribute of iterator types which should be defined as a trait. Thus, STL algorithms can be overloaded on safe iterators, too.

In the specializations one have to set the new trait, too. In the different inserter iterator types and `ostream_iterator` types the `inserter` tag has to be set to `inserting_iterator_tag`. This can be easily done if the `iterator` base type is extended with the new trait.

## IV. GENERATION OF WARNINGS

Compilers cannot emit warnings based on the semantical erroneous usage of the library. `STLlint` is the flagship example for external software that is able to emit warning when the STL is used in an incorrect way [7]. We do not want to modify the compilers, so we have to enforce the compiler to indicate these kind of potential problems [11]. However, `static_assert` as a new keyword is introduced in C++0x to emit compilation

errors based on conditions, no similar construct is designed for warnings.

```
template <class T>
inline void warning( T t )
{
}

struct
COPY_ALGORITHM_WITHOUT_INSERTER_ITERATOR
{
};

// ...

warning(
    COPY_ALGORITHM_WITHOUT_\  

    INSERTER_ITERATOR()
);
```

When the warning function is called, a dummy object is passed. This dummy object is not used inside the function template, hence this is an unused parameter. Compilers emit warning to indicate unused parameters. Compilation of warning function template results in warning messages, when it is referred and instantiated. No warning message is shown, if it is not referred. In the warning message the template argument is printed. New dummy types have to be written for every new kind of warning.

Different compilers emit this warning in different ways. For instance, Visual Studio emits the following message:

```
warning C4100: 't' : unreferenced formal
parameter
...
see reference to function template
instantiation 'void
warning<
COPY_ALGORITHM_WITHOUT_INSERTER_ITERATOR
>(T)'  
being compiled

with
[
    T=
COPY_ALGORITHM_WITHOUT_INSERTER_ITERATOR
]
```

And g++ emits the following message:

```
In instantiation of 'void warning(T)
[with T =
COPY_ALGORITHM_WITHOUT_INSERTER_ITERATOR
]':
... instantiated from here
... warning: unused parameter 't'
```

Unfortunately, implementation details of warnings may differ, thus no universal solution is available to generate custom warnings. However, everyone can find a handy, custom solution for own compiler.

This approach of warning generation has no runtime overhead because the compiler optimizes the empty function body. On the other hand – as the previous examples show – the message refers to the warning of unused parameter, incidentally the identifier of the template argument type is appeared in the message.

## V. MODIFICATION OF ALGORITHMS

As an example, now we can overload `copy` algorithm. However, transform algorithm can be written likewise.

```
template <class InputIt,
         class OutputIt>
inline OutputIt copy(
    InputIt first,
    InputIt last,
    OutputIt result )
{
    return copy(
        first,
        last,
        result,
        typename
        iterator_traits<OutputIt>::
        inserter() );
}
```

Now, we write the “usual” version of the algorithm. In this case, no warning is emitted:

```
template <class InputIterator,
         class OutputIterator>
OutputIterator copy(
    InputIterator first,
    InputIterator last,
    OutputIterator result,
    __inserting_iterator_tag )
{
    while( first != last )
    {
        *result++ = *first++;
    }
    return result;
}
```

Finally, we create the new version of the algorithm to indicate warnings:

```
template <class InputIterator,
         class OutputIterator>
OutputIterator copy(
    InputIterator first,
    InputIterator last,
    OutputIterator result,
    __non_inserting_iterator_tag )
{
    warning(
```



```

    COPY_ALGORITHM_WITHOUT_ \
    INSERTER_ITERATOR()
);
return copy( first,
             last,
             result,
             __inserting_iterator_tag() );
}

```

## VI. BELIEVE-ME MARKS

Generally, warnings should be eliminated. On the other hand, the call of `copy` or `transform` without `inserter` iterators does not mean problem necessarily.

If the proposed extensions are in use, the following code snippet results in a warning message, but it works perfectly:

```

std::vector<int> vi;
// ...
std::list<int> li( vi.size() );
std::copy( vi.begin(),
           vi.end(),
           li.begin() );

```

Many similar patterns can be shown. We use `copy` or `transform` algorithm to a target, where enough allocated space is available. Moreover, we cannot disable these specific generated warnings by a compiler flag or a preprocessor pragma.

Believe-me marks [9] are used to identify the points in the programtext where the type system cannot obtain if the used construct is risky. For instance, in the hereinafter example, the user of the library asks the type system to “believe” that the target is already allocated in the proper way. This way we enforce the user to reason about the parameters of these algorithms.

```

std::vector<int> v;
// ...
std::list<int> li( v.size() );
std::copy( v.begin(),
           v.end(),
           li.begin(),
           transmogrify,
           ALREADY_ALLOCATED );

```

This can be created by a preprocessor macro:

```

#define ALREADY_ALLOCATED \
    __inserting_iterator_tag()

```

## VII. CONCLUSION

STL is the most widely-used library based on the generic programming paradigm. It is efficient and convenient, but the incorrect usage of the library results in weird or undefined behaviour.

In this paper we present some examples that can be compiled, but at runtime their usage is defective. We argue for an extension of the iterator traits in the library, and based on this extension we generate warning messages during compilation.

The effect of our approach is similar to the `STLlint` software. `STLlint` analyzes the programtext and emits warning messages when the STL is used in an erroneous way. `STLlint` is based on a modified compiler and this way it can emit better messages. On the other hand, it is not extensible. Our approach can be used for non-standard containers, iterators, algorithms, too. Compilers cannot know all the generic libraries.

We present an effective approach to generate custom warnings. Believe-me marks are also written to disable warning messages. We overload some algorithms of the STL based on the new traits in order to make the usage of the library safer.

## ACKNOWLEDGMENT

The Project is supported by the European Union and co-financed by the European Social Fund (grant agreement no. TÁMOP 4.2.1./B-09/1/KMR-2010-0003).

## REFERENCES

- [1] A. Alexandrescu, *Modern C++ Design*, Addison-Wesley, 2001.
- [2] M. H. Austern, *Generic Programming and the STL: Using and Extending the C++ Standard Template Library*, Addison-Wesley, 1998.
- [3] T. Becker, *STL & generic programming: writing your own iterators*, *C/C++ Users Journal* 2001 **19**(8), pp. 51–57.
- [4] G. Dévai, N. Pataki, *Towards verified usage of the C++ Standard Template Library*, In Proc. of the 10th Symposium on Programming Languages and Software Tools (SPLST) 2007, pp. 360–371.
- [5] G. Dévai, N. Pataki, *A tool for formally specifying the C++ Standard Template Library*, In *Annales Universitatis Scientiarum Budapestinensis de Rolando Eötvös Nominatae, Sectio Computatorica* **31**, pp. 147–166.
- [6] D. Gregor, J. Järvi, J. Siek, B. Stroustrup, G. Dos Reis, A. Lumsdaine, *Concepts: linguistic support for generic programming in C++*, in Proc. of the 21st annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications (OOPSLA 2006), pp. 291–310.
- [7] D. Gregor, S. Schupp, *Stllint: lifting static checking from languages to libraries*, *Software - Practice & Experience*, 2006 **36**(3), pp. 225–254.
- [8] J. Järvi, D. Gregor, J. Willcock, A. Lumsdaine, J. Siek, *Algorithm specialization in generic programming: challenges of constrained generics in C++*, in Proc. of the 2006 ACM SIGPLAN conference on Programming language design and implementation (PLDI 2006), pp. 272–282.
- [9] T. Kozsik, T., *Tutorial on Subtype Marks*, in Proc. of the Central European Functional Programming School (CEFP 2006), LNCS **4164**, pp. 191–222.
- [10] S. Meyers, *Effective STL - 50 Specific Ways to Improve Your Use of the Standard Template Library*, Addison-Wesley, 2001.
- [11] N. Pataki, *Advanced Functor Framework for C++ Standard Template Library* Studia Universitatis Babeş-Bolyai, Informatica, Vol. LVI(1), pp. 99–113.
- [12] N. Pataki, Z. Porkoláb, Z. Istenes, *Towards Soundness Examination of the C++ Standard Template Library*, In Proc. of Electronic Computers and Informatics, ECI 2006, pp. 186–191.
- [13] N. Pataki, Z. Szűgyi, G. Dévai, *Measuring the Overhead of C++ Standard Template Library Safe Variants*, In *Electronic Notes in Theoretical Computer Science (ENTCS)* **264**(5), pp. 71–83.
- [14] Z. Porkoláb, Á. Sipos, N. Pataki, *Inconsistencies of Metrics in C++ Standard Template Library*, In Proc. of 11th ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering QAOOSE Workshop, ECOOP 2007, Berlin, pp. 2–6.
- [15] B. Stroustrup, *The C++ Programming Language (Special Edition)*, Addison-Wesley, 2000.
- [16] M. Torgersen, *The Expression Problem Revisited – Four New Solutions Using Generics*, in Proc. of European Conference on Object-Oriented Programming (ECOOP) 2004, LNCS **3086**, pp. 123–143.
- [17] L. Zolman, *An STL message decryptor for visual C++*, In *C/C++ Users Journal*, 2001 **19**(7), pp. 24–30.
- [18] I. Zólyomi, Z. Porkoláb, *Towards a General Template Introspection Library*, in Proc. of Generative Programming and Component Engineering: Third International Conference (GPCE 2004), LNCS **3286**, pp. 266–282.

# 3<sup>rd</sup> Workshop on Software Services: Semantic-based Software Services

**T**HE GOAL of the series of Workshops on Software Services (WoSS) is to present and discuss the recent significant developments in the field of software services.

WoSS intends to be an open forum for academics, practitioners, and vendors, allowing them to discuss the current trends and scientific and technological challenges, such as service quality assurance, adaptability, liability, interoperability, and automating service-oriented application construction and management.

The first three editions (September 2010, June 2011, and this one, upcoming in September 2011) are organized with the support of the FP7-ICT SPRERS project, with the aim to highlight the achievements of the on-going European collaborative projects subscribing to the FP7-ICT programme and the activities of the teams from new member states in software services.

The third edition is dedicated to Semantic-based Software Services and is supported also by the FP7-ICT mOSAIC project.

## TOPICS

- semantic Web technologies and standards
- semantic representations and ontologies
- semantic aware application tools
- semantic enabled middleware
- semantic collaborative research environments
- semantic interoperability of software services
- interoperability of semantic artifacts
- semantic web for clouds
- scalable reasoning for semantic web
- web scale querying and searching
- management of semantic streams
- semantic analysis of text streams
- information and service economy
- enterprise architectures and services
- context-aware computing
- service-based autonomic computing
- dynamic discovery, matching and composition of services
- quality of services and non-functional aspects
- negotiation and service-level agreements
- software engineering techniques for building semantic middleware

- software service business models
- techniques for accessing linked data

## WORKSHOP CHAIRS

**Dana Petcu**, West University of Timisoara, Romania  
petcu@info.uvt.ro

**Beniamino Di Martino**, Second University of Napoli, Italy  
beniamino.dimartino@unina.it

## PROGRAM COMMITTEE

**Alfred Ioan Letia**, Technical University of Cluj Napoca, Romania

**András Micsik**, MTA SZTAKI, Hungary

**Carlos Pedrinaci**, Open University, United Kingdom

**Claudia Raibulet**, University of Milano-Bicocca, Italy

**Dave De Roure**, Oxford e-Research Centre, United Kingdom

**Dieter Fensel**, University of Innsbruck, Austria

**Florin Fortis**, West University of Timisoara, Romania

**Giovanni Tummarello**, National University of Ireland, Galway, Ireland

**Ioan Salomie**, Technical University of Cluj Napoca, Romania

**Ivan Jelínek**, Czech Technical University in Prague, Czech Republic

**Karol Furdik**, Centre for Information Technologies, Technical University of Košice, Slovakia

**Kuo-Ming Chao**, Coventry University, UK

**Lyndon Nixon**, Free University of Berlin, Germany

**Olegas Vasilecas**, Faculty of Fundamental Sciences, Vilnius Gediminas Technical University, Lithuania

**Mikhail Simonov**, Politecnico di Milano and ISMB, Italy

**Mircea Trifu**, Forschungszentrum Informatik Karlsruhe, Germany

**Tomáš Pitner**, Faculty of Informatics, Masaryk University, Brno, Czech Republic

**Václav Snášel**, Technical University of Ostrava, Czech Republic

**Vlado Stankovski**, University of Ljubljana, Slovenia

## ORGANIZING COMMITTEE

**Viorel Negru**, **Victoria Iordan**, **Daniela Zaharie**, **Cosmin Bonchis**, West University of Timisoara, Romania



# Search-Based Testing, the Underlying Engine of Future Internet Testing

Arthur I. Baars\*, Kiran Lakhotia<sup>‡</sup>, Tanja E.J. Vos\* and Joachim Wegener<sup>†</sup>

\* Centro de Métodos de Producción de Software (ProS)  
Universidad Politecnica de Valencia, Valencia, Spain  
{abaars, tvos}@pros.upv.es

<sup>†</sup>Berner & Mattner, Berlin, Germany  
joachim.wegener@berner-mattner.de

<sup>‡</sup>CREST, University College London, London, United Kingdom  
k.lakhotia@cs.ucl.ac.uk

**Abstract**—The Future Internet will be a complex interconnection of services, applications, content and media, on which our society will become increasingly dependent. Time to market is crucial in Internet applications and hence release cycles grow ever shorter. This, coupled with the highly dynamic nature of the Future Internet will place new demands on software testing.

Search-Based Testing is ideally placed to address these emerging challenges. Its techniques are highly flexible and robust to only partially observable systems. This paper presents an overview of Search-Based Testing and discusses some of the open challenges remaining to make search-based techniques applicable to the Future Internet.

**Index Terms**—evolutionary testing; search-based testing; re-search topics.

## I. INTRODUCTION

**F**UTURE Internet (FI) applications testing will need to be continuous, post-release testing since the application under test does not remain fixed after its initial release. Services and components could be dynamically added by customers and the intended use of an application could change. Therefore, testing has to be performed continuously, even after an application has been deployed to the customer. The overall aim of the European Funded FITTEST project<sup>1</sup> (ICT-257574) is to develop and evaluate an integrated environment for continuous automated testing, which can monitor a FI application and adapt itself to the dynamic changes observed.

The underlying engine of the FITTEST environment, which will enable automated testing and cope with FI testing challenges like dynamism, self-adaptation and partial observability, will be based on Search-Based Testing (SBT).

The impossibility of anticipating all possible behaviours of FI applications suggests a prominent role for SBT techniques, because they rely on very few assumptions about the underlying problem they are attempting to solve. In addition, stochastic optimisation and search techniques are adaptive and therefore able to modify their behaviour when faced with new unforeseen situations. These two properties - freedom from limiting assumptions and inherent adaptiveness - make SBT approaches ideal for handling FI applications testing.

<sup>1</sup><http://www.facebook.com/FITTESTproject>

Since SBT is unfettered by human bias, misguided assumptions and misconceptions about possible ways in which components of a system may combine are avoided. SBT also avoids the pitfalls that are found with a humans' innate inability to predict what lies beyond their conceivable expectations and imagination. However, FI applications give users increasingly more power and flexibility in shaping an application, thus placing exactly these requirements on a human tester.

Past research has shown that SBT is suitable for several types of testing, including functional [55] and non-functional [42] testing, mutation testing [10], regression testing [60], test case prioritization [52], interaction testing [12] and structural testing [20], [25], [28], [32], [41], [43], [54]. The relative maturity of SBT means it will provide a robust foundation upon which to build FI testing techniques. However, despite the vast body of previous and ongoing work in SBT, many research topics remain unsolved. Most of them are outside the scope of the FITTEST project and so the aim of this paper is to draw attention to open challenges in SBT in order to inspire researchers and raise interest in this field.

We have divided the research topics into four categories: Theoretical Foundations, Search Technique Improvements, New Testing Objectives and Tool Environment/Testing Infrastructure.

The rest of the paper is organized as follows. Section II provides background information about SBT. Section III highlights the need for research into the theoretical foundations of SBT, before going on to list areas in which SBT may also be improved in the future in section IV. Testing objectives that remain as yet unsolved in the literature are listed in section V. Section VI presents an overview of the demands placed on tools implementing SBT techniques, before section VII concludes.

## II. SEARCH-BASED TESTING

Search-Based Testing uses meta-heuristic algorithms to automate the generation of test inputs that meet a test adequacy criterion. Many algorithms have been considered in the past, including Parallel Evolutionary Algorithms [2], Evolution

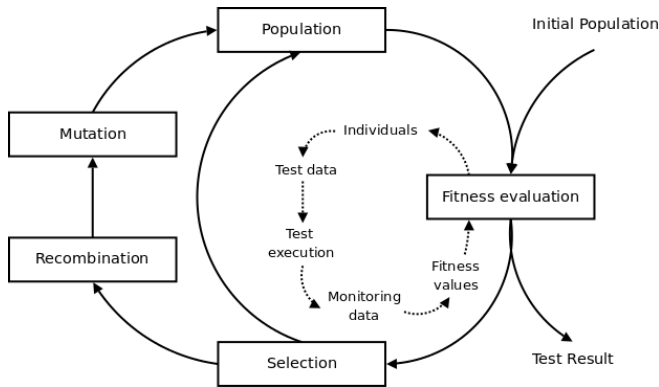


Fig. 1. A typical Evolutionary Algorithm cycle for testing.

Strategies [1], Estimation of Distribution Algorithms [48], Scatter Search [11], Particle Swarm Optimization [57], Tabu Search [13] and the Alternating Variable Method [29]. However, by far the most popular search techniques used in SBT belong to the family of Evolutionary Algorithms in what is known as Evolutionary Testing [22], [56], [55], [39].

Evolutionary Algorithms represent a class of adaptive search techniques based on natural genetics and Darwin's theory of evolution [17], [26]. They are characterized by an iterative procedure that works in parallel on a number of potential solutions to a problem. Figure 1 shows the cycle of an Evolutionary Algorithm when used in the context of Evolutionary Testing.

First, a population of possible solutions to a problem is initialized, usually at random. Starting with randomly generated individuals results in a spread of solutions ranging in fitness because they are scattered around the search-space. This is equal to sampling different regions of the search-space and provides the optimization process with a diverse set of 'building blocks'. However, for the purpose of testing one may want to seed the initial population instead, for example with existing test cases. Seeding allows the optimization process to benefit from existing knowledge about the System Under Test (SUT).

Next, each individual in the population is evaluated by calculating its fitness via a fitness function. The principle idea of an Evolutionary Algorithm is that *fit* individuals survive over time and form even fitter individuals in future generations. This is an analogy to the 'survival of the fittest' concept in natural evolution. A selection strategy is responsible for implementing this behaviour. It selects pairs of individuals from the population for reproduction such that fitter individuals have a higher probability of being selected than less fit ones. Selected individuals are then recombined via a crossover operator. The aim of the crossover operator is to combine good parts from each parent individual to form even better offspring individuals (again analogous to biological reproduction).

After crossover, the resulting offspring individuals may be subjected to a mutation operator. Mutation aims to introduce new information into the gene pool of a population by making random changes to an individual. It is an important opera-

tor to prevent the optimization process from stagnation (i.e. crossover operations are not able to produce fitter individuals).

Once offspring individuals have been evaluated, the population is updated according to a re-insertion strategy. For example, one may choose to replace the entire population with the new offspring. More commonly however, only the worst members of a generation are replaced, ensuring fit individuals will always be carried across to the next generation.

An Evolutionary Algorithm iterates until a global optimum is reached (e.g. a test criterion has been satisfied), or another stopping condition is fulfilled. Evolutionary algorithms are generic and can be applied to a wide variety of optimization problems. In order to specialize an evolutionary algorithm for a specific problem, one only needs to define a problem-specific fitness function. For Evolutionary Testing, the fitness function must capture the properties of a test adequacy criterion.

### III. THEORETICAL FOUNDATIONS

An advantage of meta-heuristic algorithms is that they are widely applicable to problems that are infeasible for analytic approaches. All one has to do is come up with a representation for candidate solutions and an objective function to evaluate those solutions. Despite this flexibility, meta-heuristic algorithms are not a 'golden hammer' that can be applied to any (testing) problem. Finding a good representation for individuals and designing suitable fitness functions is a hard task and may prove impossible for some problems.

To help a tester overcome these challenges, there is a need for guidelines on what test techniques to use for different testing objectives, different levels of testing or different phases of system development. These guidelines need to extend to how these techniques contribute to the overall reliability and dependability of the SUT, and how efficient and usable their application is. Such guidelines barely exist for traditional testing techniques, and even less is known about SBT. A *good theoretical foundation is missing to tell us which problems can be solved using search-based testing, and for which it is unsuitable*. This problem is not unique to SBT but is experienced throughout the field of Search-Based Software Engineering.

Many search algorithms are available and it is not clear which is the best for a certain problem or fitness landscape. The choice is often made somewhat ad hoc, based on experience or by trying an arbitrary selection of algorithms. To tackle this problem Harman [19] called for a more concerted effort to characterise the difficulty of Software Engineering problems for which search already produced good results. Such characterisations will aid in selecting the most suitable search technique for a particular class of problems. Since Harman's publication, researchers, leading amongst them Arcuri *et al.*, have taken up this call and started to look at theoretical properties of SBT [3], [25], [4], [5].

However, because of the diversity and complexity of the field, empirical studies remain the main means of evaluating a SBT technique. Empirical studies, if well evaluated [6], thus play an essential part in laying the foundations for guidelines

and hence need to be integrated in a general Test & Quality Assurance strategy. The result of these studies should be used to establish a central repository of examples that can act as a *benchmarking suite* for evaluation of new techniques (for functional as well as non-functional properties). A central benchmark would not only contribute to filling the knowledge gap identified by Harman [19], but it would also allow for much better development of experiments, by enabling a more thorough comparison of different testing techniques, search-based as well as others. It can further be used by the research community to gain insights into the strengths and weaknesses of each technique. This insight is invaluable for industry and enables them to make well-founded decisions on which tool or technique to apply.

Finally, theoretical foundations of SBT need to provide an *assessment of the quality of the output produced by SBT*. How good are the generated tests compared to tests derived using other techniques or developed manually by a tester? Most commonly random testing forms the baseline against which any new SBT technique is evaluated. This hardly seems sufficient in an industrial context. Furthermore, figures are needed to assess the reliability of the test results, given that SBT is based on stochastic algorithms. Such assessments are necessary to determine to which extent SBT could be used as a substitute for manual testing and to which extent as an addition to manual tests.

#### IV. SEARCH TECHNIQUE IMPROVEMENTS

Many approaches that look promising in the lab are inapplicable in the field, because they do not scale up. However, making a solution scalable is easier said than done. This section highlights some areas that may enable search-based techniques to scale up in practice.

##### A. Parallel computing

A great advantage of evolutionary computing is that it is naturally parallelizable. Fitness evaluations for individuals can easily be performed in parallel, with hardly any overhead. Search algorithms in general and SBT in particular therefore offer a ‘killer application’ for the emergent paradigm of ubiquitous user-level parallel computing. Grid-computing for example is the subject of a great number of EU-projects, so there is an opportunity to team up with these projects and apply the technologies developed in that area in SBT.

Another active research area is about achieving parallelization of evolutionary computation through General Purpose Graphics Processing Unit (GPGPU) cards. In the context of evolutionary computation, GPGPU cards are most commonly used to evolve (parts of) programs through Genetic Programming [47], [46], [18], [8], [38], [34]. Genetic Programming uses trees to represent a program (or part of it) and applies genetic operators such as crossover, mutation and selection to a population of trees. Since trees can grow large in size, and possibly have to be executed (interpreted) a number of times (to account for non-determinism) in order to obtain a fitness value, GP can be very computationally expensive.

However, GPGPU cards have also been used to parallelize Particle Swarm Algorithms [15], Evolution Strategies [62], Genetic Algorithms [58], [51], [45], [37], [44] and multi-objective problems [59].

Despite the large body of work on evolutionary computation on GPGPU cards, more research is required to utilize GPGPU cards for SBT. Langdon *et al.* [35] have used GPGPU cards to optimize the search for higher order mutants in mutation testing. The goal of mutation testing is to evaluate the quality of a test suite by generating a set of mutant programs, where each mutant represents a possible fault in the program. A mutant is said to be killed if one or more test cases that pass on the original program, either fail for the mutant program, or produce a different output.

The intuition behind mutation testing is that the more mutants a test suite is able to kill, the better it is at finding real faults. Typically, a mutant program only contains a single change. The concept of higher order mutation testing is to introduce multiple changes into one mutant program, because it is argued that higher order mutants more closely represent real faults in software [23].

##### B. Combining search techniques

Another way of increasing efficiency is to combine different search techniques in the form of Memetic Algorithms. A study by Harman and McMinn [25] shows that a Memetic Algorithm, combining a global and local search, is better suited to generate branch adequate test data than either a global or local search on its own. Again, more research is needed to find out what combination of search techniques is best for which category of test objective.

Sometimes, a search technique may also be combined with a different testing technique. The work of Lakhota *et al.* [33] combines different search techniques with Dynamic Symbolic Execution (DSE) [16], [49] to improve DSE in the presence of floating point computations. Inkumsah and Xie combined a Genetic Algorithm (GA) and DSE in a framework called Evacon [27] to improve branch adequate testing of object oriented code.

In both studies, the combination of different test data generation techniques outperforms a pure search-based or dynamic symbolic execution-based approach.

##### C. Multi-objective approaches

In many scenarios, a single-objective formulation of a test objective is unrealistic; testers will want to find test sets that meet several objectives simultaneously in order to maximize the value obtained from the inherently expensive process of running test cases and examining their output. An added benefit of multi-objective optimization is the insight a Pareto front can yield into the trade-offs between different objectives.

By targeting multiple test objectives at the same time, the value obtained from the expensive process of executing the SUT can be maximized. A case study by Harman *et al.* [24] shows promising results in this direction. The study investigates the performance of a multi-objective GA for the

twin objectives of achieving branch coverage and maximizing dynamic memory allocation.

Besides the work by Harman *et al.* [24], the field of multi-objective testing has remained relatively unexplored. Regression testing, and in particular test suite minimization [61], is one of the few areas where multi-objective algorithms have been applied. The work mentioned in this sub-section on multi-objective testing suggests that search-based techniques are well suited for multi-objective testing tasks, but more research is needed to maximise their potential.

#### D. Static parameter tuning

Evolutionary Algorithms are a very powerful tool for many problems. However, to obtain the best performance it is crucial that their parameters are well-tuned in order to perform a particular task. This requires a level of expertise in the area of evolutionary computation. Unfortunately testers usually have very little knowledge in this field.

A solution would be that the testing tool automatically tunes its parameters. One approach is to tune parameters a priori, based on the characteristics of the SUT. These could be obtained, for example, from the tester, as a tester has a lot of knowledge about the SUT. In the case of white box testing this information can also stem from (static) analysis of the SUT.

#### E. Dynamic parameter tuning

A better approach is to let a search algorithm tune itself, based on how well it is proceeding. In this way the search can automatically adapt to a (possibly changing) fitness landscape. This approach seems very promising for search problems that have many different sub-goals or are very dynamic as in FI applications. Every sub-goal represents a new optimisation problem, thus, parameter settings that work well for one sub-goal, might not work well for others.

#### F. Testability transformations

A testability transformation [21] is a source-to-source program transformation that seeks to improve the performance of a previously chosen test data generation technique. For structural testing, it is possible to remove certain code constructs that cause problems for SBT by applying transformations. This approach is taken, for example, when removing flag variables. A flag variable is a boolean variable, and the flag problem deals with the situation where relatively few input values exist that make the flag adopt one of its two possible values. As a consequence, flag variables introduce large plateaus in the search space, effectively deteriorating a guided search into a random search.

A possible solution to the flag problem is to apply a testability transformation. Many different transformations have been suggested in the literature to deal with different types of flag variables; simple flags [22], function assigned flags [53] and loop-assigned flags [9].

Testability transformations have also been used to remove nesting of conditional statements from source code for the

purpose of test data generation [40]. Nested predicates can have a similar effect on SBT to flag variables. If nested conditional statements are linked through a data dependency, the search is missing crucial information on how to satisfy the nested predicates. Eventually a search will be able to obtain this information, but at that point the required test data has been found. The lack of information available during the optimization process can again lead to plateaus in the fitness landscape.

#### G. Search space size reduction

Another way to improve efficiency of SBT is to use knowledge about the SUT to restrict the size of the search space. For example, knowledge on value ranges could be used to set parameters of the search, such as step size for variation of integers, doubles, etc. Another example is the seeding of test data with literals extracted from the program code. Such strategies could result in a very significant search space reduction and thus potential speed up of the testing process.

There are many ways to uncover information about a SUT. The models and specifications (on system, software, design or component level) could be analysed for information that can be used to improve the test or the search. Static analysis may be used to determine which input-variables are relevant to the search. Irrelevant variables can be left out, reducing the complexity of the input domain. The effect of input domain reduction via irrelevant variable removal has been investigated by Harman *et al.* [20] on two commonly used search algorithms; the Alternating Variable Method, a form of hill climbing, and a Genetic Algorithm. A theoretical and empirical analysis shows that both test data generation methods benefit from a reduced search space.

The bounds of variables or the control flow are other examples of knowledge that can be used by a fitness function to guide the search. Abstract interpretation may be employed to provide equivalence partitions. Such a partition is a range of values for which the SUT behaves the same. During the search one needs to sample only a single element of the partition to cover the whole range, greatly reducing the search space.

#### H. Minimizing generated individuals

It is often assumed that a fitness evaluation is the most time consuming task of SBT. However, for time consuming functional testing of complex industrial systems, minimizing the number of generated individuals may also be highly desirable. This might be done using an assumption about the 'potential' of individuals in order to predict which individuals are likely to contribute to any future improvement. This prediction could be achieved by using information about similar individuals that have been executed in earlier generations.

#### I. Seeding of test data

Instead of starting with a completely random population, the search may be initialised using results from previous testing activities. This way the search can benefit from prior knowledge. Different strategies for seeding of test data are investigated by Arcuri *et al.* [7].



### J. Other interesting questions

What can we learn about the system under test from the execution of a huge number of test data? Is testing the only thing or could we achieve results for other software engineering activities from that?

## V. NEW TESTING OBJECTIVES

The bulk of previous work on SBT focuses on structural test objectives, such as branch coverage [20], [25], [28], [32], [41], [43], [54]. Although the topic of branch coverage is extensively researched, there are still many points for improvements:

- dealing with internal states
- dealing with predicates containing complex types, such as strings, dates, data structures and arrays.
- dealing with loops, especially data dependencies between values calculated in loops and used outside the loop.
- how to improve the calculation of the fitness function for combined conditions (logical and, logical or, etc.)

Besides these points, SBT may also be used to address new testing objectives, both structural as well as functional ones. Research is needed to develop an appropriate representation and fitness function for each new testing objective.

Below we describe a number of possible testing objectives. For some it is clear how to implement them, for others the required representations and fitness functions have not yet been designed and are thus open research topics.

### A. Run-time error testing

Some examples of run-time errors are: integer overflow, division by zero, memory leaks. For testing run-time errors the objective is to find inputs that trigger such an error. It should be possible to tackle this area by extending the existing approaches for structural testing. For example to test for memory leaks the fitness function should favour test-inputs on which the subject under test uses more memory. The work by Harman *et al.* [24] provides a starting point in this direction. Equally, the work by Tracey *et al.* [50] on exception testing provides a base on which to build.

### B. Testing interactive systems

The test input for interactive systems is a sequence of user actions, such as keystrokes and mouse clicks. This is similar to GUI testing. A possible test criterion is the responsiveness of an application. In this case a fitness function should favour combinations of user actions that take a long time to complete. Another objective could be the coverage of different user actions in various combinations.

### C. Integration testing

A system usually consists of a number of modules that are more or less independent from each other. These modules should of course be tested in isolation. However, there are also problems that only occur when integrating the different modules. What should the test goals and corresponding fitness functions be for applying SBT to integration tests?

### D. Testing parallel, multi-threaded systems

Testing parallel, multi-threaded systems is hard, especially finding bugs that only occur with a certain interleaving of the processes or threads of the SUT. The need for testing becomes ever larger, systems get more and more complex and multi-processor computers are getting more common. An objective for testing such systems is trying to find deadlock or race conditions. The fitness function should somehow favour executions that are close to a deadlock or race condition. Open questions for this type of testing include the representation of individuals (corresponding to interleaving executions of the SUT) and how to measure the ‘closeness’ to a deadlock.

Again, a combination of different techniques might prove to be desirable. For example, Bohuslav *et al.* [30] use SBT to optimize the configuration of ConTest [14], a concurrency testing tool for JAVA. Their work shows that SBT can help increase synchronisation coverage of code, and thus increase the chance of finding bugs that appear in commercial software due to errors in synchronization of its concurrent threads.

## VI. TOOL ENVIRONMENT/TESTING INFRASTRUCTURE

Despite a growing momentum of SBT research in academia, SBT is struggling to find any up-take in industry. One of the reasons is a lack of tooling available. The limited tools that are available are often research prototypes, focused on a subset of a particular programming language. Furthermore, problematic language constructs, such as variable length argument functions, pointers and complex data types (variable size arrays, recursive types such as lists, trees, and graphs) remain largely unsupported. Lakhota *et al.* [31] made some progress towards this, but many more problems remain, some of which are listed in [32].

Central to any SBT technique is a well designed fitness function. Current tools require a tester to manually write code for a fitness function and integrate that function into the tool. In the research prototypes available, this is often not a straightforward task. Lindlar [36] introduces an approach that aims at simplifying the process of designing fitness functions for evolutionary functional testing in order to further increase acceptability by practitioners. The difficult task of designing a suitable fitness function could further be supported by using a wizard based approach.

Finally, non of the currently available SBT tools provide any visualization that might aid a tester. However, visualisation can provide a user with important insights. There are several aspects of visualisation:

- 1) Visualisation of testing progress, for example how much was tested, testing effort, test coverage, reliability figures.
- 2) Visualisation of search progress, e.g. how does the search perform, potential for better results when continuing, identify potentials for improving the search and fitness landscape.

Important questions include, which data is useful to a practitioner, and how to display this data in a concise manner?

Displaying the amount of coverage for a small piece of code is easy, one can simply colour the covered code in the editor, or display a coloured control flow graph of the code. However, for a large system consisting of many lines of code, different techniques need to be developed. Further, visualising the fitness landscape is a challenge. It usually has many dimensions, making it hard to display concisely in 2-D.

## VII. CONCLUSION

Search-Based Testing provides the basis on which the European Funded FITTEST project builds. The goal of the FITTEST project is to perform automated, continuous testing of Future Internet applications. Such testing places new demands on any automated testing technique. This paper has presented an overview of Search-Based Testing and discussed some of the open challenges remaining to make search-based techniques applicable to industry as well as the Future Internet. The aim is to encourage further research into these topics such that users of SBT (like the FITTEST project) may benefit from the results.

## ACKNOWLEDGEMENT

Many people have contributed to the contents of this paper through personal communications and discussions. We would like to thank Mark Harman from University College London; Youssef Hassoun from King's College London; Marc Schoenauer from INRIA; Jochen Hänsel from Fraunhofer FIRST; Dimitar Dimitrov and Ivaylo Spasov from RILA; Dimitris Togias from European Dynamics; Phil McMinn from University of Sheffield; John Clark from the University of York.

## REFERENCES

- [1] Enrique Alba and Francisco Chicano. Software Testing with Evolutionary Strategies. In *RISE*, pages 50–65, 8-9 September 2005.
- [2] Enrique Alba and Francisco Chicano. Observations in using Parallel and Sequential Evolutionary Algorithms for Automatic Software Testing. *Computers & Operations Research*, 35(10):3161–3183, October 2008.
- [3] A. Arcuri. Full theoretical runtime analysis of alternating variable method on the triangle classification problem. In *Search Based Software Engineering, 2009 1st International Symposium on*, pages 113–121, may 2009.
- [4] Andrea Arcuri. Insight knowledge in search based software testing. In *GECCO*, pages 1649–1656. ACM, 2009.
- [5] Andrea Arcuri. Theoretical analysis of local search in software testing. In *Stochastic Algorithms: Foundations and Applications*, volume 5792 of *Lecture Notes in Computer Science*, pages 156–168. Springer Berlin / Heidelberg, 2009.
- [6] Andrea Arcuri and Lionel Briand. A practical guide for using statistical tests to assess randomized algorithms in software engineering. In *ICSE*, pages 1–10. ACM, 2011.
- [7] Andrea Arcuri, David Robert White, John Clark, and Xin Yao. Multi-objective improvement of software using co-evolution and smart seeding. In *Proc of the 7th Int Conf on Simulated Evolution And Learning (SEAL '08)*, volume 5361 of *LNCS*, pages 61–70. Springer, December 7-10 2008.
- [8] Wolfgang Banzhaf, Simon Harding, William B. Langdon, and Garnett Wilson. Accelerating genetic programming through graphics processing units. In *Genetic Programming Theory and Practice VI*, pages 1–19. 2009.
- [9] André Baresel, David Binkley, Mark Harman, and Bogdan Korel. Evolutionary testing in the presence of loop-assigned flags: a testability transformation approach. In *ISSA*, pages 108–118. ACM, 2004.
- [10] Benoit Baudry, Franck Fleurey, Jean-Marc Jézéquel, and Yves Le Traon. From genetic to bacteriological algorithms for mutation-based testing. *Softw. Test, Verif. Reliab*, 15(2):73–96, 2005.
- [11] Raquel Blanco, Javier Tuya, Eugenia Daz, and B. Adenso Daz. A Scatter Search Approach for Automated Branch Coverage in Software Testing. *International Journal of Engineering Intelligent Systems (EIS)*, 15(3):135–142, September 2007.
- [12] Myra B. Cohen, Peter B. Gibbons, Warwick B. Mugridge, and Charles J. Colbourn. Constructing test suites for interaction testing. In *ICSE*, pages 38–48. IEEE Computer Society, May 3–10 2003.
- [13] Eugenia Díaz, Javier Tuya, Raquel Blanco, and José Javier Dolado. A Tabu Search Algorithm for Structural Software Testing. *Computers & Operations Research*, 35(10):3052–3072, October 2008.
- [14] Orit Edelstein, Eitan Farchi, Evgeny Goldin, Yarden Nir, Gil Ratsaby, and Shmuel Ur. Framework for testing multi-threaded java programs. *Concurrency and Computation: Practice and Experience*, 15(3-5):485–499, 2003.
- [15] S. Genovesi, R. Mitra, A. Monorchio, and G. Manara. Particle swarm optimization of frequency selective surfaces for the design of artificial magnetic conductors. Technical report, 2006.
- [16] Patrice Godefroid, Nils Klarlund, and Koushik Sen. DART: Directed Automated Random Testing. *ACM SIGPLAN Notices*, 40(6):213–223, June 2005.
- [17] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, 1989.
- [18] Simon Harding and Wolfgang Banzhaf. Distributed genetic programming on gpus using cuda. In *WPABA'09: Proceedings of the Second International Workshop on Parallel Architectures and Bioinspired Algorithms (WPABA 2009)*, pages 1–10. Universidad Complutense de Madrid, September 12-16 2009.
- [19] Mark Harman. The current state and future of search based software engineering. In *2007 Future of Software Engineering, FOSE '07*, pages 342–357. IEEE Computer Society, 2007.
- [20] Mark Harman, Youssef Hassoun, Kiran Lakhota, Phil McMinn, and Joachim Wegener. The impact of input domain reduction on search-based test data generation. In *ESEC/SIGSOFT FSE*, pages 155–164. ACM, 2007.
- [21] Mark Harman, Lin Hu, Rob Hierons, Joachim Wegener, Harmen Sthamer, André Baresel, and Marc Roper. Testability transformation. *IEEE Trans. Softw. Eng.*, 30:3–16, January 2004.
- [22] Mark Harman, Lin Hu, Robert Hierons, André Baresel, and Harmen Sthamer. Improving evolutionary testing by flag removal. In *GECCO*, pages 1359–1366. Morgan Kaufmann Publishers, 9-13 July 2002.
- [23] Mark Harman, Yue Jia, and William B. Langdon. A manifesto for higher order mutation testing. In *Mutation 2010*, pages 80–89. IEEE Computer Society, 6 April 2010. Keynote.
- [24] Mark Harman, Kiran Lakhota, and Phil McMinn. A multi-objective approach to search-based test data generation. In *GECCO*, pages 1098–1105. ACM, 2007.
- [25] Mark Harman and Phil McMinn. A theoretical and empirical study of search-based testing: Local, global, and hybrid search. *IEEE Trans. Software Eng.*, 36(2):226–247, 2010.
- [26] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975.
- [27] Kobi Inkumsah and Tao Xie. Evacon: A framework for integrating evolutionary and concolic testing for object-oriented programs. In *Proc. 22nd IEEE/ACM International Conference on Automated Software Engineering (ASE 2007)*, pages 425–428, November 2007.
- [28] Bogdan Korel. Automated software test data generation. *IEEE Transactions on Software Engineering*, 16(8):870–879, August 1990.
- [29] Bogdan Korel. Automated software test data generation. *IEEE Transactions on Software Engineering*, 16(8):870–879, 1990.
- [30] Bohuslav Krena, Zdenek Letko, Tomás Vojnar, and Shmuel Ur. A platform for search-based testing of concurrent software. In *PADTAD*, pages 48–58. ACM, 2010.
- [31] Kiran Lakhota, Mark Harman, and Phil McMinn. Handling dynamic data structures in search based testing. In *GECCO*, pages 1759–1766. ACM, 2008.
- [32] Kiran Lakhota, Phil McMinn, and Mark Harman. An empirical investigation into branch coverage for C programs using CUTE and AUSTIN. *The Journal of Systems and Software*, 83(12):2379–2391, December 2010.
- [33] Kiran Lakhota, Nikolai Tillmann, Mark Harman, and Jonathan De Halleux. Flopsy: search-based floating point constraint solving

- for symbolic execution. In *Proceedings of the 22nd IFIP WG 6.1 international conference on Testing software and systems, ICTSS*, pages 142–157. Springer-Verlag, 2010.
- [34] William B. Langdon and Mark Harman. Evolving a CUDA kernel from an nvidia template. In *IEEE Congress on Evolutionary Computation*, pages 1–8. IEEE, 2010.
- [35] William B. Langdon, Mark Harman, and Yue Jia. Efficient multi-objective higher order mutation testing with genetic programming. *The Journal of Systems and Software*, 83(12):2416–2430, December 2010.
- [36] Felix Lindlar. Search-based functional testing of embedded software systems. In *Doctoral Symposium in conjunction with ICST*, 2009.
- [37] Ogier Maitre, Laurent A. Baumes, Nicolas Lachiche, Avelino Corma, and Pierre Collet. Coarse grain parallelization of evolutionary algorithms on gpgpu cards with easea. In *GECCO*, pages 1403–1410. ACM, 2009.
- [38] Ogier Maitre, Pierre Collet, and Nicolas Lachiche. Fast evaluation of GP trees on GPGPU by optimizing hardware scheduling. In *Proceedings of the 13th European Conference on Genetic Programming, EuroGP 2010*, volume 6021 of *LNCS*, pages 301–312. Springer, 7-9 April 2010.
- [39] Phil McMinn. Search-based software test data generation: A survey. *Software Testing, Verification and Reliability*, 14(2):105–156, 2004.
- [40] Phil McMinn, David Binkley, and Mark Harman. Empirical evaluation of a nesting testability transformation for evolutionary testing. *ACM Trans. Softw. Eng. Methodol.*, 18:11:1–11:27, June 2009.
- [41] Christoph C. Michael, Gary McGraw, and Michael Schatz. Generating software test data by evolution. *IEEE Trans. Software Eng.*, 27(12):1085–1110, 2001.
- [42] F. Mueller and J. Wegener. A comparison of static analysis and evolutionary testing for the verification of timing constraints. In *RTAS '98: Proc of the Fourth IEEE Real-Time Technology and Applications Symposium*, page 144. IEEE, 1998.
- [43] R. P. Pargas, M. J. Harrold, and R. R. Peck. Test-data generation using genetic algorithms. *Journal of Software Testing, Verification and Reliability*, 9(4):263–282, 1999.
- [44] Petr Pospchal, Ji Jaro, and Josef Schwarz. Parallel genetic algorithm on the cuda architecture. In *Applications of Evolutionary Computation*, LNCS 6024, pages 442–451. Springer Verlag, 2010.
- [45] Jos L. Risco-Martn, Jos M. Colmenar, and Rubn Gonzalo. A parallel evolutionary algorithm to optimize dynamic memory managers in embedded systems. In *WPABA'09: Proceedings of the Second International Workshop on Parallel Architectures and Bioinspired Algorithms (WPABA 2009)*, pages 21–30. Universidad Complutense de Madrid, September 12-16 2009.
- [46] Denis Robilliard, Virginie Marion, and Cyril Fonlupt. High performance genetic programming on GPU. In *Proceedings of the 2009 workshop on Bio-inspired algorithms for distributed systems*, pages 85–94. ACM, 2009.
- [47] Denis Robilliard, Virginie Marion-Poty, and Cyril Fonlupt. Genetic programming on graphics processing units. *Genetic Programming and Evolvable Machines*, 10(4):447–471, December 2009. Special issue on parallel and distributed evolutionary algorithms, part I.
- [48] Ramón Sagarna, Andrea Arcuri, and Xin Yao. Estimation of Distribution Algorithms for Testing Object Oriented Software. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, pages 438–444. IEEE, 25-28 September 2007.
- [49] Nikolai Tillmann and Jonathan de Halleux. Pex-white box test generation for .NET. In *Tests and Proofs, Second International Conference, TAP 2008, Prato, Italy, April 9-11, 2008. Proceedings*, volume 4966 of *Lecture Notes in Computer Science*, pages 134–153. Springer, 2008.
- [50] Nigel Tracey, John A. Clark, Keith Mander, and John McDermid. Automated test-data generation for exception conditions. *Software Practice and Experience*, 30(1):61–79, 2000.
- [51] Shigeyoshi Tsutsui and Noriyuki Fujimoto. Solving quadratic assignment problems by genetic algorithms with gpu computation: a case study. In *GECCO*, pages 2523–2530. ACM, 2009.
- [52] Kristen R. Walcott, Mary Lou Soffa, Gregory M. Kapfhammer, and Robert S. Roos. Time aware test suite prioritization. In *ISSTA*, pages 1 – 12. ACM Press, 2006.
- [53] Stefan Wappler, Joachim Wegener, and André Baresel. Evolutionary testing of software with function-assigned flags. *The Journal of Systems and Software*, 82(11):1767–1779, November 2009.
- [54] J. Wegener, A. Baresel, and H. Sthamer. Evolutionary test environment for automatic structural testing. *Information and Software Technology*, 43(1):841–854, 2001.
- [55] Joachim Wegener and Oliver Bühler. Evaluation of different fitness functions for the evolutionary testing of an autonomous parking system. In *Proc of the Genetic and Evolutionary Computation Conf*, pages 1400–1412, 2004.
- [56] Joachim Wegener, Kerstin Buhr, and Hartmut Pohlheim. Automatic test data generation for structural testing of embedded software systems by evolutionary testing. In *GECCO*, pages 1233–1240. Morgan Kaufmann, 2002.
- [57] Andreas Windisch, Stefan Wappler, and Joachim Wegener. Applying particle swarm optimization to software testing. In *GECCO*, pages 1121–1128. ACM, 2007.
- [58] Man Wong and Tien Wong. Implementation of parallel genetic algorithms on graphics processing units. In *Intelligent and Evolutionary Systems*, pages 197–216. 2009.
- [59] Man Leung Wong. Parallel multi-objective evolutionary algorithms on graphics processing units. In *GECCO*, pages 2515–2522. ACM, 2009.
- [60] Shin Yoo and Mark Harman. Pareto efficient multi-objective test case selection. In *ISSTA*, pages 140–150. ACM, 2007.
- [61] Shin Yoo and Mark Harman. Using hybrid algorithm for pareto efficient multi-objective test suite minimisation. *Journal of Systems Software*, 83(4):689–701, April 2010.
- [62] Weihang Zhu. A study of parallel evolution strategy: pattern search on a gpu computing platform. In *GECCO*, pages 765–772. ACM, 2009.



## Testing and Remote Maintenance of Real Future Internet Scenarios

Towards FITTEST and FastFix Advanced Software Engineering

Alessandra Bagnato  
Corporate Research Divisions, TXT e-solutions  
alessandra.bagnato@txt.it

Anna I Esparcia-Alcázar  
S2 Grupo, 46022 Valencia  
aesparcia@s2grupo.es

Tanja E.J. Vos  
Centro de Métodos de Producción de Software (ProS)  
Universitat Politècnica de València  
tvos@pros.upv.es

Beatriz Marín  
Centro de Métodos de Producción de Software (ProS)  
Universitat Politècnica de València  
bmarin@pros.upv.es

José Oliver Murillo  
Infoport Valencia  
joliver@infoportvalencia.es

Salvador I. Folgado  
BULL Spain  
salvador.folgado@bull.es

Auxiliadora Carlos Alberola  
INDRA  
acarlosa@indra.es

**Abstract**—In recent years, software testing and maintenance services are key factors of customers' perception of software quality. Nowadays, customers are more demanding about these services, while contribution of maintenance and testing services to products total cost of ownership should be reduced. Reducing these costs is even more crucial for SME's. To do this, new methods and techniques that will be aligned with the needs of companies are required. This paper presents the preliminary results of an interactive workshop celebrated by researchers and three companies. In the workshop, researchers present the advanced software engineering methods proposed by FastFix<sup>1</sup> and FITTEST<sup>2</sup> European projects. After that, discussions about their potential use in three application scenarios at Infoport Valencia, BULL Spain, and INDRA were performed and some lessons were learned.

**Index terms**—testing; maintenance; practical experience.

### I. INTRODUCTION

IN RECENT years, software testing and maintenance services are key factors of customers' perception of software quality. Therefore, companies are more demanding about these services, while contribution of maintenance and testing services to products total cost of ownership should be reduced. Reducing these costs is even more crucial for SME's, which has limited resources to spend in testing and maintenance of their products.

The Future Internet (FI) aims at reducing developing, testing and maintenance services through a common space

where different services can be combined to produce software reducing their total cost. Thus, it can be anticipated that the Future Internet will be a complex interconnection of services, applications, content, and media; all of which will increase with semantic information. Also, it can be foreseen that future web applications will offer a rich user experience, extending and improving current hyperlink-based navigation.

Thus, future internet applications are expected to be complex applications, which present the following characteristics: self-modifiability, autonomic behavior, low observability, asynchronous information, time- and load-dependent behavior; a huge feature configuration space, and ultra-large scale. Since current maintenance and testing techniques are not suitable to future internet applications, FastFix and FITTEST projects are developed to face the challenges of these applications.

The overall purpose of FastFix is to provide software applications with a maintenance environment featuring the highest time efficiency at the lowest cost and the strongest accuracy. To this effect, FastFix will develop a platform and a set of tools that will monitor on-line customer environments, collecting information on program execution and user interaction, with the objective of identifying symptoms of execution errors, performance degradation or changes in user behaviour. By use of correlation techniques, the platform will also allow failure replication in order to identify incorrect execution patterns, patch generation, and patch deployment.

The overall aim of the FITTEST project is to address the testing challenges that arise in Future Internet Systems due to the complexity of the technologies involved. To this end, FITTEST will develop an integrated environment for automated testing, which can monitor the Future Internet applications under test and adapt the testing to the dynamic

<sup>1</sup> FastFix (Monitoring Control for Remote Software Maintenance) (FP7-25810) is an FP7 project.

<sup>2</sup> FITTEST (Future Internet Testing) (FP7-257574) is an FP7 project.

changes observed. The environment will implement continuous post-release testing since a Future Internet application does not remain fixed after its release. Services and components could be dynamically added by customers and the intended use could change significantly. The environment will integrate, adapt and automate various techniques for continuous Future Internet testing (e.g. dynamic model inference, model-based testing, log-based diagnosis, oracle learning, combinatorial testing, concurrent testing, regression testing, etc.).

In order to make sure that both projects are aligned with what is needed in industry with regard to testing and maintenance, a joint FITTEST & FastFIX workshop was organized in June 2011 during which the following research questions were posed:

**Question 1:** Can Future Internet Maintenance and Testing support real scenarios?

**Question 2:** Are the approaches proposed by FastFix and FITTEST interesting to the real end-user needs?

**Question 3:** How much effort has to be put into creating new tools to support the expressed needs?

The rest of the paper is organized as follows: Section II presents the FastFix Project and Section III presents the FITTEST Project. Sections IV, V and VI describe the IN-DRA, BULL and Infoport Valencia Scenarios, respectively. Lessons learned are addressed in Section VII. Finally, Section VIII presents some conclusions and outlines future work.

## II. FASTFIX PROJECT

Maintenance and support services that are time and cost efficient is the driving goal of the FastFix project [2], which started in June 2010. This is to be achieved by monitoring software applications, replicating execution failures, and automatically generating patches.

Among the results of the project will be a platform and a set of tools that will monitor online customer environments. This will gather information on program execution and user interaction, aiming to identify symptoms of execution errors, performance degradation or changes in user behaviour. The platform will also allow the replication of failures by means of correlation techniques; the purpose of this feature is to identify incorrect execution patterns and facilitate patch generation and deployment.

In order to achieve this, mechanisms will be developed and set up to gather the required information on application execution, errors, context, and user behaviour. These mechanisms will be applicable to both new and existing applications; they should also be non-intrusive and pose a minor, acceptable burden on performance.

Information thus gathered will be sent in real time to a support centre, via the Internet. Hence, special care will be required with regard to security and privacy.

At the support centre, information will be used to replicate errors, by means of correlation techniques and error ontologies which will allow the identification of behavioral patterns and possible causes of error. At its best, the FastFix platform will be able to generate patches for the errors in an automated way. These patches will consist on application

modifications, changes in the system configuration or parameterization or even a limitation in functionality in order to avoid system or application crashes. Patches will be sent back to the application's runtime environment and will be deployed automatically, resulting in a self-healing software application.

Figure 1 illustrates the components of the FastFix Architecture.

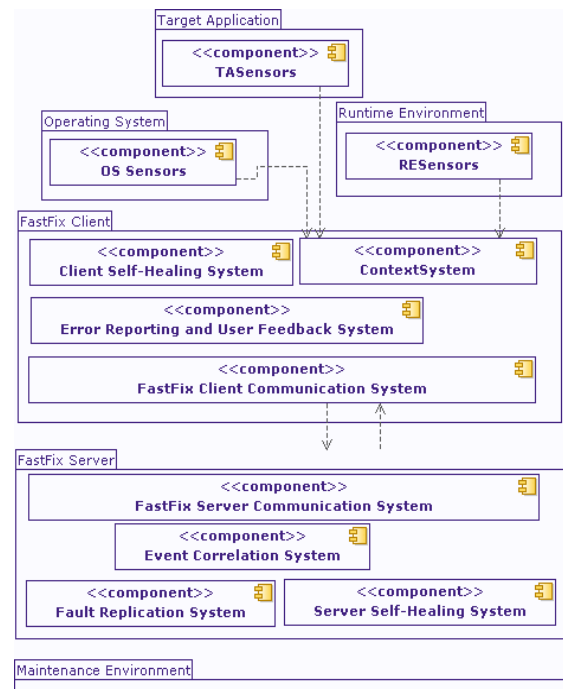


Figure 1. FastFix architecture for software maintenance

Four main lines of research converge in the FastFix project and constitute the core of innovation activities:

- **Context elicitation and user modeling:** determines which information on execution and interaction is going to be gathered independently from the application and its environment, and how this is to be done.
- **Event correlation:** allows drawing conclusions about the kind of problems the application is undergoing and their possible causes, based on the information gathered.
- **Fault replication:** provides the platform that allows replicating faults, which mimic the real circumstances as much as possible.
- **Patch generation and self-healing:** determines which patches are going to be generated and how this is to be done, plus the way they will be deployed to the application at the runtime environment. An approach based on control theory that allows for fast and reliable maintenance fixes is mentioned in [3], this approach aims to disable faulty or vulnerable system functionalities and requires to instrument the system before deployment so that it can

later be monitored and interact with a supervisor at runtime.

FastFix will provide innovation in any of its four main research areas, namely event correlation, fault replication, patch generation, and context elicitation.

Event correlation techniques will be used within FastFix in the field of software defect detection and cause identification. Developments in this area have mostly focused on system software [9]. Thanks to the event correlation FastFix will be also able to determine the type and level of monitoring that must be exercised in each execution instance.

Failure information will consider privacy concerns associated with the release of failure information providing novel data obfuscation techniques which will preserve the program re-execution's accuracy. Analysis techniques operating at source code [12][13] and binary level [10][11] are taken as a starting point for this research.

In FastFix, autonomous system principles and methods focusing on assessing the viability of auto-generating patches [14][15][16] will be applied.

As opposed to current approaches [17] [18] [19], in which collected data related to context describes only the usage of a particular tool, context elicitation in FastFix will be carried out independently from the monitored application and the usage domain,

### III. FITTEST PROJECT

The FITTEST project (September 2010-2013) is being carried out by eight partners: Universitat Politècnica de València, University College London, University of Utrecht, Fondazione Bruno Kessler, Berner&Mattner System technic, Sulake, Soft-Team, and IBM Israel.

Existing literature on web testing (such as [20][21][22]) is focused on client-server applications which implement a strictly serialized model of interaction, based on <form submission, server response> sequences. Testing of Ajax and rich client web applications has been considered only more recently [23][24]. The differences between Ajax and more traditional web testing are discussed in [25]. Even though there are some recent works that consider testing of dynamic web applications, they are not addressing the testing challenges of future web applications mentioned in [5]. For this reason, we consider the development and evaluation of an integrated environment for continuous evolutionary automated testing, which can monitor the FI application and adapt itself to the dynamic changes observed.

FITTEST testing will be continuous post-release testing since the application under test does not remain fixed after its release. Services and components could be dynamically added by customers and the intended use could change. Therefore, testing has to be performed continuously after deployment to the customer.

The FITTEST testing environment will integrate, adapt and automate various techniques investigated in the project for continuous FI testing, providing a user friendly way to activate and parameterize them. See Figure 2 for a global picture of the testing environment.

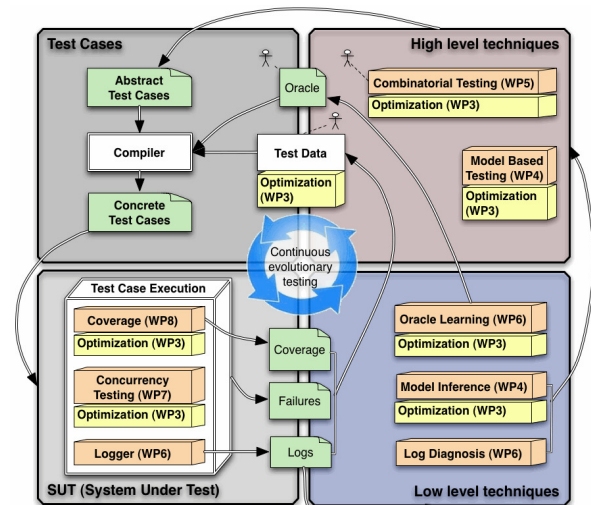


Figure 2. Global Picture of the FITTEST Testing environment

The underlying engine of the FITTEST environment, that will make it possible to automate undecidable problems and cope with the testing challenges like dynamism, self-adaptation and partial observability, will be based on search-based testing [4].

The impossibility of anticipating all possible behaviours of FI applications suggests a prominent role for evolutionary testing techniques, because it relies on very few assumptions about the underlying problem that is attempting to solve. In addition, stochastic optimisation and search techniques are adaptive and, therefore, able to modify their behaviour when faced with new unforeseen situations. These two properties – their freedom from limiting assumptions and their inherent adaptiveness – make evolutionary testing approaches ideal for handling FI applications testing, with their dynamism, self-adapting, autonomous and unpredictable behaviour. Since evolutionary testing is unfettered by human bias, misguided assumptions and misconceptions about possible ways in which the components of the system may combine, FITTEST avoids the pitfalls that are found with humans-in-nate inability to predict that which lies beyond their conceivable expectations and imagination. Moreover, evolutionary techniques are well understood techniques for solving general undecidable problems and will constitute a robust and stable foundation upon which to build FITTEST.

To achieve this overall aim, FITTEST will address a set of objectives that directly map to the identified challenges.

**Objective 1: Search based testing approach.** To cope with dynamism, self-adaptation and partial observability that characterize FI applications, we will use search-based software testing. Evolutionary algorithms themselves exhibit dynamic and adaptive behaviour and, as such, are ideally suited to the nature of the problem. Moreover, evolutionary algorithms have proved to be very efficient for solving general undecidable problems and provide a robust framework.

**Objective 2: Continuous, automated testing approach.** Since the range of behaviours is not known in advance, testing will be done continuously; feedback from post-release



executions will be used to co-evolve the test cases for the self-adaptive FI application; humans alone cannot achieve the desired levels of dependability, so automation is required.

**Objective 3: Dynamic model inference.** Self-adapting applications with low observability demand for dynamic analysis; models will be inferred continuously rather than being fixed upfront.

**Objective 4: Model based test case derivation.** Behavioural models inferred from monitored executions will be the basis for automated test case generation. Paths in the model associated with semantic interactions will be regarded as interesting execution sequences. To support continuous, extensive testing of FI applications, test case generation will proceed fully unattended, including the generation of input data and the verification of feasibility for the test adequacy criteria of choice.

**Objective 5: Log-based diagnosis and oracle learning.** Since correct behaviour cannot be fixed upfront, executions will be analysed to identify atypical ones, indicating likely faults or potential vulnerabilities.

**Objective 6: Dynamic classification tree generation.** The huge configuration space will be dealt with by testing combinatorially, using dynamically and continuously automated generated classification trees.

**Objective 7: Test for concurrency bugs.** Toward successful concurrency testing and debugging of FI applications, we will develop a mechanism to control and record factors like communication noise, delays, message timings, load conditions, etc, in a concurrent setup.

**Objective 8: Testing the unexpected.** Due to the high dynamism, it is impossible to define the expected interactions upfront; we will use genetic programming to simulate unpredicted, odd, or even malicious interactions.

**Objective 9: Coverage and regression testing.** Novel coverage and regression testing criteria and analytical methods will be defined for ultra-large scale FI applications, for which the standard criteria and analysis techniques are not applicable since they just do not scale.

**Objective 10: General methodological evaluation framework for FI testing.** Large scale case studies will be performed using realistic systems and software testing practitioners. The studies will be executed using an instantiation and/or refinement of the general methodological evaluation framework to fit specific software testing techniques and tools and evaluation situations.

#### IV. INDRA SCENARIO

INDRA [6] is a global company of technology, innovation, and talent, leader in solutions and services for Transport and Traffic, Energy and Industry, Public Administration, Healthcare, Finance, Insurance, Security and Defense, Telecom and Media sectors.

At INDRA Valencia, one thousand of professionals are working in different markets. Indra main activity (65%) is on health care, followed by other activities in Public administration and energy industry, representing the 21% and 6% of INDRA total work, respectively.

Health care software is then the most relevant sector at Valencia. In particular it consists of the following products:

- Abucasis, a web application that connects together the entire primary assistance system, including electronic prescriptions.
- INDRAHealth Solution, an integrated hospital and primary assistance web application.
- Medas, different modules that ingrate different specific applications as for example: blood transfusion center, or emergency management

INDRAHealth solution is a project that involves a huge amount of health professionals and developers, so it was necessary to define a clear development methodology to have success in the project.

The methodology defined at INDRA has been adapted to perform testing. In INDRA methodology, testing activity starts after the working tasks are assigned to the developers, meaning that testing is done in parallel with the development team. With this methodology, we have obtained quite an efficient distribution of work load. Thus, testing is not only performed by testers. Developers are also assigned a set of test activities that are performed before the source code is released.

INDRA's testing methodology is organized in three different stages:

1. The first stage include test definition and planning, and is performed in parallel with code development
2. The second stage is performed after developers have finished, and include test execution and fault registration.
3. Once the faults are solved, they are tested again and, finally a regression test is performed.

The automated tests are launched when nobody is working on the application with a clean database. Finally, when the application is stable, INDRA starts the performance tests. The purpose of performance testing is to simulate the normal use of the system before it passes to production, in order to find potential problems and correct them.

In the simulation, it is possible to predict how INDRA Helath Solution will behave with a specific load. In a broad sense: the system capacity is verified to be adequate for the demands of work supports, and the potential bottlenecks and inefficiencies are identified by providing the necessary information for correct them.

Every semester INDRA calculates the following measures in order to improve their testing process:

- Test coverage
- Effectiveness of tests
- Number of faults detected per KLOCs
- Number of faults resolved per KLOCs
- Percent of faults resolved

The principal testing problems at INDRA are:

- No economic resources allocated to testing
- Little understanding of the test work from the developers' part.
- Changes in functionality are implemented at the final stages of projects.
- There is low stability in modules, but is not possible to automate everything. Thus, more automation is needed.
- Sometimes, the communication between teams is slow.

Thus, in the future, INDRA would like to show the managers that test is a necessary part of the software process development. Also INDRA has to make developers understand that test can be a useful tool to help them in the software development. As web applications may present large changes in a short time, INDRA needs an easy and fast way to test, which means less manual test.

## V. BULL SCENARIO

As the only European player that covers the whole of the IT value chain, BULL [7] is ideally placed to help its customers build value-creating information systems. In an open and fast-changing world, Bull helps companies and administrations to liberate themselves from technological shackles, enabling them to radically gear up their innovative capabilities. In its relentless quest for safe, cost-effective, sustainable, open-critical infrastructures, BULL cultivates cutting-edge know-how, teams up with top-tier partners, and develops a broad, powerful, and modular range of products and services.

By consistently offering customers the fruit of innovative, purpose-designed, high-performance ideas, BULL has earned worldwide recognition and enjoys a leading position in Europe.

At the Software Quality department at BULL Spain S.A. (BULL QA), assessing the quality of Service-Oriented Architectures (SOA) through a specialization of the testing environment is particularly interesting.

SOA, by its nature has the following characteristics: self-contained; highly modular and independently deployed; consists of distributed components that are available over the network; has a published interface; only needs to see the interface; stresses interoperability; different implementation languages and platforms are involved; is discoverable; needs a directory of services that are registered and located; is dynamically bound; and can locate services and bind them at runtime.

An Enterprise Service Bus (ESB) is a logical architectural component that provides an integration infrastructure consistent with the principles of SOA. The most effective way to test SOA environments is through a systematic approach following the V testing model. All components that are part of the architecture first need to be tested independently and then in integration with other components and systems involved.

SOA test design should follow a top down approach, and the test execution should follow a bottom up approach, starting at individual service (or component) level. The following characteristics must be considered when testing:

**Functionality.** Since there are no user interfaces, a “formal contract” to reach adequate testing quality must be used. Key business stakeholders and users should be more actively involved in all project lifecycle. Regression testing must be made more efficient and should be automated.

**Interoperability.** It is necessary to focus on interfaces, and assure that interface behavior and information sharing between services are working as specified. Integration testing should include communication, network protocols, transformations, etc.

**Compliance.** It is necessary to satisfy standard (law) more formal and more accurately.

**Backward Compatibility.** BULL needs to assure that SOA architecture continuously work successfully even when any modification is done in any component/service, BULL needs to assure “Loose Coupling”.

**Security.** SOA provide access and potential modification of services (data) from different physical locations, over the WAN. Thus, How safe is the data? Business requirements must include security requirements. BULL must perform a security risk analysis during design and will need formal reviews to assure that the organization security standard is satisfied. Penetration security test must be planned and executed.

**Performance.** Is the most degraded quality characteristic the designer must take care of when doing SOA design. The following specific characteristics that would impact the performance at SOA must be considered in order to perform a correct performance testing:

- *Distributed computing.* Services are normally located in different containers, most often in different machines.
- *Heterogeneous message and protocols.* Transformation must be managed and transform inside ESB.
- *Different platforms.* Involves different technologies, different behaviors, BULL must be care with the “The Weakest Link” effect.
- *XML intensive.* XML message can be 10 or 20 times larger than equivalent binary representation, so transmission over a network takes longer. XML uses text format, so it must be processes before any operation is performed (Parsing, Validation and Transformation) are CPU and MEMORY intensive.
- *Scalability (vertical, horizontal).* Because service users know only about service interface and not its implementation, providing scalable solutions requires little overhead.

- *Latency.* Distributions means interconnections, between networks and its service quality, conventional TCP is a guaranty delivered protocol. TCP has a direct inverse relationship between latency and throughput, then, massive message at ESB implementation could increase latency.
- *Not data oriented.* ESB is not a final data oriented solution, however governance produces a significant data quantity to be managed, as accounting, auditing, service location, etc.

In this context, for SOA testing at BULL the following conclusions and challenges are listed:

- SOA testing is different from other applications testing.
- SOA implementations are a combination of any kind of components.
- SOA testing is more complex than traditional testing (granularity, systems, messages, etc.)
- Functionality has less risk than others quality characteristics.
- Early testing (design, unit testing) is highly recommended to assure a proper SOA architecture.
- Testing approach strategy depend on SOA implementation (test design top-down approach, test execution bottom-up approach)
- The SOA design and implementation will provide the critical main quality characteristics to satisfy
- Performance characteristics could be impacted in a SOA Approach.

## VI. INFOPORTVALENCIA SCENARIO

Infoport Valencia [8] is an information technology and communications service provider. The company has been working on development and software maintenance projects since its creation, more than 10 years ago.

One of the projects Infoport is working on is a set of IT operations and testing service for Valencia Port Community System (PCS), which is an information system that offers services for managing port, sea, and land logistic operations, tracking, and other operations with more than 400 organizations as users, including private enterprises and public institutions as Valencia Port Authority.

PCS platform allows electronic data interchange over the Internet between organizations involved in port logistics activity. The system is fully developed with Microsoft .NET technology. More than 30 IT professionals belonging 4 different companies work on development and operations tasks

of the platform. Technical infrastructure includes more than 20 servers between physical and virtual ones.

PCS project started more than 7 years ago and goes on launching new services and functionalities every year. This means a continuous flow of software development and maintenance packages that require a high level coordination and well-defined processes to guarantee an agile evolution of the system.

The software development, testing and deployment processes for new functionalities and maintenance packages make use of four different environments (development, testing, pre-production, and production). Obviously, a package that does not pass the tests and validation process in one environment will not be deployed in the next one.

Software maintenance requires an environment maintenance and synchronization too, at data and code levels, and a common planning including all software packages dependencies, timing, and resources. A common problem is when a package does not pass the acceptance (or any other) tests and other packages planned to be deployed after that package with dependencies on that one need to be replanned.

Once a package is deployed in production environment two kinds of maintenance operations are needed: one adaptive (or preventive) for dealing with changes in the software environment or user requirements, and a corrective maintenance to deal with incidents found and fix them.

Infoport has defined an incident lifecycle with a 3-tier support: first, a user reports an incident to the user's support center, where operators may attend and solve functional requests. If an incident requires a technical analysis, it is assigned to the second support level, where a technician may decide to fix an incident in production environment with a hotfix, depending on its impact and priority. If the incident is not critical or requires a major modification to be fixed, then, it is assigned to the third support level, where developers will fix and prepare a new package to be delivered.

The testing process at Infoport requires a planning prior to deliver a release. At this stage, information about dependencies with other packages, priority, and resources are defined. Next step is to deliver the required documentation to testing and operations teams (as functional and technical design and requirements). These documents are reviewed in order to check its completeness and used them as an input to prepare the test cases (scenarios, data set and expected results). Once the release is delivered, it is built, and software verification and validation is done. If the package is accepted, it is deployed in the testing environment, and test cases are executed. A report containing the results of the executed test cases is created. If faults have been found, these are sent to development teams to fix them in a new package. This process is repeated as many times as needed until the package passes all test cases. Sometimes, this means several iterations and delays in planning.

In order to improve this, Infoport is evaluating now some changes in the process to anticipate functional testing to an early stage and reduce the number of iterations that a package requires to pass tests, and therefore, be deployed in production environment.

A common problem in the testing process is the time required to prepare and execute test cases. Data sets need to be prepared with so many combinations as exist. If this is a

manual process, it is highly time-costly and usually data sets do not cover all possible cases. If testing tools are used, time may be reduced in some parts of the process but on the other hand it is increased in others, as testing tool needs to be prepared too. Infoport combines the use of manual and automated testing. Usually, Infoport testing process is a manual process, but in some cases software simulators are developed to test specific parts.

In this context, Infoport's main challenges to improve their testing and maintenance processes are:

- **Automation.** Vulnerability to failure because most of the testing process is manual. Currently Infoport is evaluating the use of testing tools as VS2010 and Lab Management 2010 to automate part of the process.
- **Number of test-error-fix iterations.** Impact of replanning testing tasks and delays in production environment deployments. To mitigate this, Infoport has created testing teams to execute functional tests in early stages of the process, before delivery to the operations team.
- **Environments synchronization**
- **Requirements detail level.** Sometimes requirements definition and functional design is not enough for testing teams to create the test cases. Infoport is evaluating the use of Microsoft Team Foundation Server for quality assurance and requirements management.

## VII. LESSONS LEARNED

Question 1, "Future Internet Maintenance and Testing can support real scenarios?" and Question 2 "Are the approaches proposed by FastFix and FITTEST interesting to the real end-user needs?" have been positively answered by the companies during the workshop. As researchers and experts agreed, it has been acknowledged that the relative cost for software testing, maintenance, and management of its evolution represents around the 70% of the total cost.

**FastFix** could help significantly reduce time used in failure cause identification, patch generation, and deployment meeting end-user needs and expectances.

Maintenance encompasses all the costs incurred to fix faults ("corrective maintenance"), maintain the engineering integrity of the application ("adaptive maintenance"), change the structure of the application to meet changing business needs ("perfective maintenance"), and stop predictable faults in the future ("preventive maintenance"). FastFix could help in corrective maintenance by identifying the failure and its context, and partially in adaptive and preventive maintenance by identifying failures and patching the system, at least, providing temporary patches.

Even in cases where FastFix will not be able to automatically identify causes or generate patches, it has been consid-

ered as very valuable the fact that FastFix could provide valuable context information, both on execution environment and in user interaction, which will facilitate the task for a software engineer.

The outcomes of the **FITTEST** project will support companies with their test automation needs. Moreover, the FITTEST continuous testing approach will help them to cope with changing requirements and dynamic nature of the Internet applications.

In particular:

- **INDRA** will see how advances in FITTEST and FastFix projects can help in the automation of test cases and in the improvement of the testing work in place.
- The objectives of the FITTEST and FastFix projects were considered as extremely important for **Infoport**, as their lines of research and results may help Infoport to incorporate some improvements to their testing and maintenances processes to make them more efficient, particularly to automate part of the process and reduce the number of test-error-fix iterations.
- During the workshop it became clear that the FITTEST project's objectives tackle many of **BULL**'s SOA testing challenges. **BULL** will see how the advances in the FITTEST and FastFix projects could help in providing an integrated framework capable of analyzing and evaluating the quality of SOA.

Question 3 "How much effort has to be put into creating new tools to support the expressed needs?" had to be postponed to the next conference event that will be organized in a year time by FITTEST and FastFix projects. The two projects are in the early phases of their research efforts, and currently they are working in building their first prototypes and could not provide a precise estimation.

A very positive aspect that was remarked by the involved researchers was how close to the business needs of the involved SMEs the two projects are. This is an incentive to pursue further the collaboration among FITTEST and FastFix projects, and to the organization of the second "FITTEST & FastFIX joint workshop - Conference Testing and Remote Maintenance of the Future Internet workshops" 2012.

## VIII. CONCLUSION AND FUTURE WORK

In this article, we have summarized the needs and the feedbacks gained by introducing the FastFix and FITTEST advanced research ideas to three different industrial environment and scenarios context. The results of the potential of the tools developed within the two projects are promising and it was acknowledged that their adoption would allow improving the current practice in real industrial scenarios. Using FastFix and FITTEST tools, software can be tested and maintained with improved quality and in a faster way.

It has been considered as very important for the acceptance of the produced tools the fact that testers and maintenance engineers could be able to easily use the tools in the testing phase and in the maintenance phase of the software development lifecycle.

It was evident from the discussions carried out that SMEs needs to take full advantage of an adequate on-site customer support for maintenance as the one proposed within FastFix. Software vendors need a system to remotely provide a high quality support service to their customers, improve user experience and facilitate corrective, adaptive and preventive maintenance – of both new and existing software products.

It was also evident from the discussions carried out that SMEs need testing techniques and tools that perform automatically the testing process of their applications. This is precisely the main result of FITTEST project, which consist in a set of testing techniques and tools that allows the automatic generation and execution of test cases for future internet applications. At this stage, the difficulties related to the selection, usage and adaptation of different available testing tools in the different contexts remain an issue to be treated case by case.

As future work, the two projects (i.e., FastFix and FITTEST) plan to host another conference in a year time when more results will be available.

#### ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under the grant agreement FP7-258109 FastFix and FP7-257574 FITTEST.

#### REFERENCES

- [1] FITTEST Project Consortium. Web Site. <http://www.facebook.com/FITTESTproject>
- [2] FastFix Project Consortium. Web Site <http://www.fastfix-project.eu/>
- [3] Gaudin B., Bagnato A. (2011). Software Maintenance through Supervisory Control. SEW-34, the 34th Annual Ieee Software Engineering Workshop Proceedings Limerick, Ireland, 21 June 2011
- [4] Mark Harman. The current state and future of search based software engineering. In 2007 Future of Software Engineering, FOSE '07, pages 342–357. IEEE Computer Society, 2007.
- [5] Tanja E. J. Vos, Paolo Tonella, Joachim Wegener, Mark Harman, Wishnu Prasetya, Elisa Puoskari, and Yarden Nir Buchbinder. Future internet testing with fittest. Software Maintenance and Reengineering, European Conference on, 0:355–358, 2011.
- [6] INDRA Web site <http://www.indracompany.com/>
- [7] BULL Web site <http://www.bull.es>
- [8] InfoPort Valencia Web site <http://www.infoportvalencia.es>
- [9] Viliam Holub, LERO et alter. "Run-Time Correlation Engine for System Monitoring and Testing". ICAC'09. June 2009
- [10] Newsome, J., Brumley, D., Franklin, J., and Song, D. 2006. Replayer: automatic protocol replay by binary analysis. In Proceedings of the 13th ACM Conference on Computer and Communications Security (USA, October 30 - November 03, 2006). CCS '06. ACM, New York, NY, 311-321.
- [11] David Brumley, Juan Caballero, Zhenkai Liang, James Newsome, and Dawn Song. Towards Automatic Discovery of Deviations in Binary Implementations with Applications to Error Detection and Fingerprint Generation. Proceedings of USENIX Security Symposium, Aug 2007.
- [12] S. Elbaum, H. N. Chin, M. B. Dwyer, and J. Dokulil. Carving differential unit test cases from system test cases. In Symp. Foundations of Software Engineering, pages 253–264, 2006.
- [13] Xu, G., Rountev, A., Tang, Y., and Qin, F. 2007. Efficient checkpointing of java software using context-sensitive capture and replay. In Proceedings of the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering ESEC-FSE '07. ACM, New York, NY, 85-94.
- [14] G. Williamson, D. Cellai, S. Dobson, and P. Nixon, "Self-management of routing on human proximity networks," in In Proceedings of International Workshop on Self-Organising Systems, Springer Verlag Lecture Notes in Computer, 2009.
- [15] S. Dobson, S. Denazis, A. Fernández, D. Gaiti, E. Gelenbe, F. Massacci, P. Nixon, F. Saffre, N. Schmidt, and F. Zambonelli, "A survey of autonomic communications," ACM Transactions on Autonomous and Adaptive Systems, vol. 1, pp. 223–259, December 2006.
- [16] B. Gaudin, P. Nixon, K. Bines, F. Busacca, and Casey, "Model bootstrapping for auto-diagnosis of enterprise systems," in Proceedings of the International Conference on Computational Intelligence and Software Engineering (CiSE), p. to appear, IEEE Press, December 2009.
- [17] W. Maalej, Task-First or Context-First? Tool Integration Revisited, In 24th ACM/IEEE International Conference On Automated Software Engineering, 2009
- [18] W. Maalej and H.J. Happel, From Work to Words: How do Software Developers Describe Their Work, in Proceedings of the 6th IEEE Conference On Mining Software Repositories, IEEE CS, 2009
- [19] W. Maalej, H.J. Happel, A. Rashid, When Users Become Collaborators: Towards Continuous and Context-Aware User Input, In companion of ACM OOPSLA 2009.
- [20] F. Ricca, P. Tonella, Analysis and Testing of Web Applications, in: International Conference on Software Engineering (ICSE), 2001, pp. 25-34.
- [21] S.G. Elbaum, G. Rothermel, S. Karre, M. Fisher, Leveraging User-Session Data to Support Web Application Testing, IEEE Trans. Software Eng., vol. 31 n° 3 (2005), pp. 187-202.
- [22] S. Sampath, S. Sprenkle, E. Gibson, L.L. Pollock, A.S. Greenwald, Applying Concept Analysis to User-Session-Based Testing of Web Applications, IEEE Trans. Softw Eng., vol. 33 n° 10 (2007), pp. 643-658.
- [23] A. Mesbah, A. van Deursen, Invariant-based automatic testing of AJAX user interfaces, in: International Conference on Software Engineering, 2009, pp. 210-220.
- [24] A. Marchetto, P. Tonella, F. Ricca, State-Based Testing of Ajax Web Applications, in: ICST, 2008, pp. 121-130.
- [25] A. Marchetto, F. Ricca, P. Tonella, A case study-based comparison of web testing techniques applied to AJAX web applications, STTT, vol. 10 n° 6 (2008), pp. 477-492.

# A Neural Model for Ontology Matching

Emil Șt. Chifu and Ioan Alfred Leția

Department of Computer Science, Technical University of Cluj-Napoca, Barițiu 28, RO-400027 Cluj-Napoca,  
Romania

Email: Emil.Chifu@cs.utcluj.ro, letia@cs-gw.utcluj.ro

**Abstract**—Ontology matching is a key issue in the Semantic Web. The paper describes an unsupervised neural model for matching pairs of ontologies. The result of matching two ontologies is a class alignment, where each concept in one ontology is put into correspondence with a semantically related concept in the other one. The framework is based on a model of hierarchical self-organizing maps. Every concept of the two ontologies that are matched is encoded in a bag-of-words style, by counting the words that occur in their OWL concept definition. We evaluated this ontology matching model with the OAEI benchmark data set for the bibliography domain. For our experiments we chose pairs of ontologies from the dataset as candidates for matching.

## I. INTRODUCTION

THE USAGE of software services raises the problem of discovering the relevant ones for a given purpose, and still manual effort is needed to find and compose services. To solve this problem, the researchers in Semantic Web propose to make the service descriptions more meaningful by annotating them with a semantic description of their functionality. And the meaning of these semantic descriptions is specified in a domain ontology [12].

The semantic heterogeneity problem is encountered classically in the information integration area as well as in the new domain of Semantic Web. Ontology matching allows the knowledge and data expressed in different ontologies to interoperate. To name only two situations, the interoperability is important when two agents communicate, as well as for merging two ontologies into a result, combined and still consistent single ontology. As a consequence, ontology matching is a key issue in the Semantic Web [5].

In this paper we propose an unsupervised neural model for matching pairs of ontologies. The result of matching two ontologies is a class alignment: each concept in one ontology is put into correspondence with the most related concept in the other ontology from the semantic point of view. We disregard here the problem of also matching the properties and the instances of the ontology concepts.

In order to establish a mapping of the concepts of one ontology to the concepts of the other ontology, we classify the concepts of the first ontology against the taxonomic structure of the second ontology. The classification starts from a representation of the concepts built as a result of analyzing their OWL concept definitions. We collect the

words used in the semistructured textual descriptions extant in OWL ontology definition files. Based on this text mining process, we represent the ontology nodes (the concepts) as bag-of-words vectors. These vectors are used as input data for an unsupervised neural network. More specifically, our framework is based on an unsupervised training of an extended model of hierarchical self-organizing maps.

In our approach, we consider the OWL semistructured textual definition of each concept as a small text document. As such, we represent the ontology classes as text documents, like in a document categorization setting. We cast the concept classification problem as a text document classification in a vector space. The semantic content features of a concept are the frequencies of occurrence of different words in the document representation of the concept. The classification of the concepts of the first ontology into the taxonomy of the second ontology proceeds by associating every ontology 1 concept to one node of the taxonomy of ontology 2, based on a similarity in the vector space.

The taxonomy of the second ontology is given as the initial state of the neural network. The training of the unsupervised neural network takes place by exposing the initialized hierarchical self-organizing map to the vector representations of the document-like concepts of the first ontology, as extracted from their OWL definitions.

In the rest of the paper, after a review of related work, section III presents the neural network learning solution chosen and adapted in our framework. Then section IV details the architecture and implementation of our ontology matching model, and section V describes the experimental results. The conclusions and future directions are presented in section VI.

## II. RELATED WORK

A classification process algorithmically similar to our ontology matching scenario takes place in the named entity classification and taxonomy enrichment methods, where the task is to associate every named entity to one taxonomic concept or to add new concepts as subclasses attached under existent nodes (concepts) of a given initial taxonomy. From this point of view, our approach is similar to [10, 1, 13, 2]. All the methods mentioned in [10, 1, 13, 2] use a similarity based top-down classification process like in our model. The

main difference from our work is that their classification is based on decision trees, whereas our classification is driven by a neural network.

The book [5] includes a comprehensive survey of ontology matching approaches and tools. Taking into account the kind of input used for computing the similarity of a concept in the first ontology to an appropriate concept in the second ontology, the authors classify the approaches as name-based (terminological) techniques, structure-based, extensional, and semantic approaches. In their turn, the terminological techniques are divided into string based and linguistic approaches.

The method described in this paper belongs to the string based approaches. In order to compute the similarity between two matched concepts we actually compute a *token-based distance* between them. The computation of token-based distances is inspired from information retrieval, and consequently a concept definition is taken as a string consisting of a multi-set of words (also called bag of words). The bag of words for each concept is a vector in a vector space, and the token based similarity is a similarity metric specific to the vector space.

This token string based category of methods is called linguistic matching in [9]. And the multi-set of words (the bag of words) is called a virtual document, which contains the definition of a concept. The approaches in this category extract the meaning of each concept from such virtual documents.

Many of the ontology matching approaches make use of external resources of common knowledge like lexicons, thesauri, and upper level or domain specific ontologies [5]. It is worth mentioning the method in [11], which dynamically select, exploit (reuse), and combine multiple and heterogeneous online ontologies in order to derive semantic mappings. The mappings are provided either by a single online ontology or by a reasoning process that uses several ontologies. By this process of harvesting the Semantic Web, the authors prove that the Semantic Web itself is a dynamic source of background knowledge that can be successfully used to solve real-world problems, including ontology matching.

### III. MACHINE LEARNING METHOD

In our framework for ontology matching, we classify the concepts of the first ontology against the taxonomic structure of the second ontology. Our extended model of hierarchical self-organizing maps – Enrich-GHSOM – represents the unsupervised neural network based learning solution adopted by our framework. This choice is suitable to the knowledge structure onto which the concepts of the first ontology are classified – a taxonomy, i.e. an is-a hierarchy of concepts.

#### A. Self-organizing Maps

The Self-Organizing Map (SOM, also known as Kohonen map) learning architecture [8] is one of the most popular unsupervised neural network models. SOM can be seen as a projection method which maps a high dimensional data space into a lower dimensional one. The resulting lower dimensional output space is a rectangular SOM map,

represented as a two-dimensional grid of neurons. Each input data item is mapped into one of the neurons in the map. SOM plays also the role of a clustering method, so that similar data items – represented as vectors of numerical attribute values – tend to be mapped into nearby neurons.

The SOM map learns by a self-organization process. The training proceeds with unlabeled input data like any unsupervised learning. Clusters (classes) are discovered and described by gradually detected characteristics during the training process. These gradually adjusted characteristics play the role of weights in the weight vector associated to each neuron.

#### B. Growing Hierarchical Self-organizing Maps

The growing hierarchical self-organizing map (GHSOM) model consists of a set of SOM maps arranged as nodes in a hierarchy and it is able to discover hierarchical clusters [4]. The SOM's in the nodes can grow horizontally during the training by inserting either one more row or one more column of neurons. This happens iteratively until the average data deviation (quantization error) over the neurons in the SOM map decreases under a specified threshold  $\tau_1$ .

The SOM's in the nodes can also grow vertically during the training, by giving rise to daughter nodes. Each neuron in the SOM map could be a candidate for expansion into a daughter node SOM map (see Figure 1). The expansion takes place whenever the data deviation on the current neuron is over a threshold  $\tau_2$ . The thresholds  $\tau_1$  and  $\tau_2$  control the granularity of the hierarchy learned by GHSOM in terms of depth and branching factor.

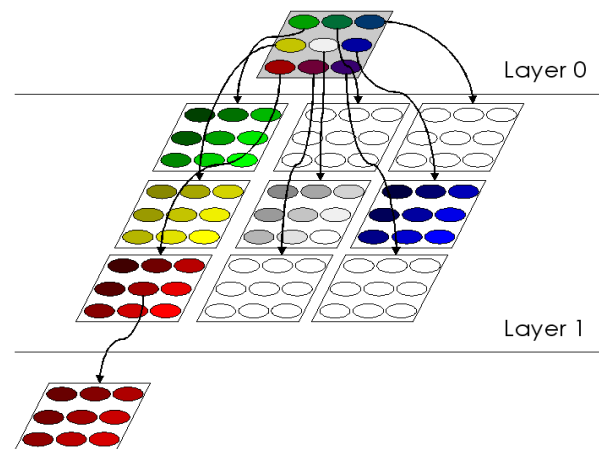


Figure 1. The GHSOM neural network model.

#### C. Enrich-GHSOM

The growth of a GHSOM neural network is a completely unsupervised process, being only driven by the unlabeled input data items themselves together with the two thresholds and some additional learning parameters. There is no way to suggest from outside any initial paths for the final learnt hierarchy. We have extended the GHSOM model with the possibility to force the growth of the hierarchy along with some predefined paths of a given hierarchy.

Our new extended model, Enrich-GHSOM, is doing a classification of the data items into an existing tree hierarchy structure. This initial tree plays the role of an initial state for



the tree-like neural network model. The classical GHSOM model grows during the training by only starting from a single node. The top-down growth in our extended model starts from a given initial tree and inserts new nodes attached as successors to any of its intermediate and leaf nodes.

In Enrich-GHSOM, the nodes of the predefined hierarchy are labeled with some data item labels from the input data space used for training. Algorithm 1 describes the learning algorithm of the Enrich-GHSOM neural network, where  $mqe_j$  is the mean quantization error of the data items mapped on neuron  $j$ , and  $mqe_0$  is the global quantization error of the entire training data set.

---

**Algorithm 1:** *Training the Enrich-GHSOM neural network*

---

**Inputs:** predefined initial tree;  
training data space;

**Output:** enriched tree;

**begin**

layer  $i = 0$

**do**

{

// training epoch associated to layer  $i$ :

**for all** (SOM maps on layer  $i$ )

{

// phase 1:

Train the SOM map

// The SOM training converges

// by satisfying threshold  $\tau_1$ .

// phase 2:

**for all** (neurons  $j$  of the current SOM map)

{

**if**(neuron  $j$  has been initialized as predefined)

{

Propagate the data set mapped in neuron  $j$

towards the predefined daughter map

of neuron  $j$ .

}

**else** // Neuron  $j$  has been initialized randomly.

**if**( $mqe_j > \tau_2 * mqe_0$ )

{

Born a new daughter map from neuron  $j$ ,

exactly like in the GHSOM model.

}

}

}

$i = i + 1$

}

**while**( there is at least one SOM map on layer  $i$  )

**end**

---

The training data items propagate top-down throughout the given tree hierarchy structure. When the propagation process hits a parent SOM of a tree node, then the weight vector of the corresponding parent neuron in that parent SOM is already initialized with the data item vector of that predefined daughter node label. The weight vectors of the SOM neurons with no predefined daughter are initialized with random values. Then the training of that SOM map proceeds by classifying the training data items against the

initialized neurons. Training data items that are similar (as vectors) to the predefined initialized neurons are propagated downwards to the associated predefined daughter SOM nodes to continue the training (recursively) on that predefined daughter SOM. Data items that are not similar to the initialized neurons are mapped to other, non-initialized (actually randomly initialized), neurons in the same SOM. They propagate downwards only if the threshold  $\tau_2$  is surpassed, giving rise to new daughter SOM nodes.

For instance, consider the parent neuron of a current SOM node is labeled *mammal*, and there are two predefined daughter nodes labeled *feline* and *bear*, which correspond to two predefined initialized neurons in the current SOM. Then the training data item vector *dog* is not similar to any of the two neuron initializer weight vectors associated to *feline* and *bear* (see Figure 2, where the neuron initializers are marked with bold). So *dog* will remain as classified into that SOM – mapped on another, non-initialized neuron – i.e. as daughter (direct hyponym) of *mammal* and twin of the existent nodes *feline* and *bear*. Also, a data item labeled *tiger* – similar with the weight vector of the “*feline*” neuron – will be propagated into the associated predefined daughter SOM map together with other terms that correspond to felines. They will all become direct or indirect hyponyms of the concept *feline*. The process continues top-down for all the SOM nodes in the predefined initial tree hierarchy, ending at the leaves. The data item vector representations of the node labels of the given initial tree play the role of *predefined initializer weight vectors* for our neural model.

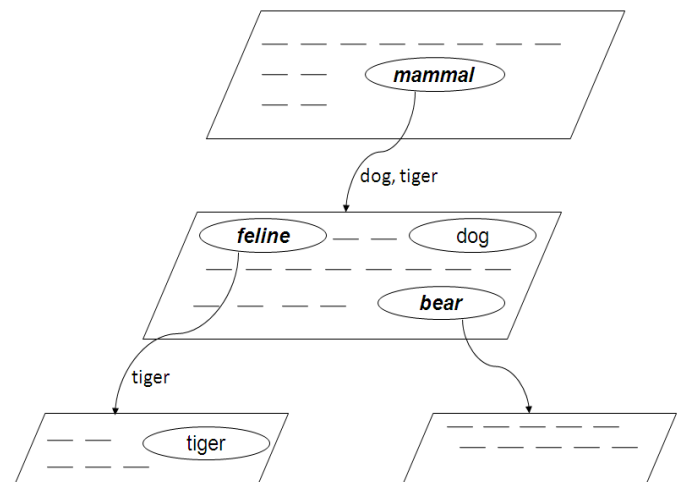


Figure 2. The Enrich-GHSOM neural network model.

For the evaluation of the classification accuracy of the Enrich-GHSOM neural network, we compare the given initial tree-like state of the network with the enriched tree-like state. The output tree is nothing else than the input tree enriched with the new nodes inserted as classified during the top-down training process. We consider a training data item as classified into, or associated to, one node of the given input tree when that SOM node is the last node of the given tree that has been traversed by the data item before leaving the tree. Obviously this is also the deepest of the input tree nodes traversed by the item during the training. After leaving the predefined input tree, the data item will only traverse

SOM nodes newly inserted during the Enrich-GHSOM training. In short, the nodes to which different data items get classified are the nodes where that data items leave the given input tree.

#### IV. A MODEL FOR UNSUPERVISED ONTOLOGY MATCHING

The architecture of our model is implemented as a pipeline with several processing stages. The whole processing can be divided in two main steps: the *acquisition of the vector representations of the concepts*, and then the *classification of the first ontology concepts into the taxonomy of the second ontology*.

The OWL class definition of each concept of the two ontologies being matched is considered as a small text document. For one concept, the document is the fragment broken out of the OWL ontology description, which defines only that concept. From such a semistructured document-like representation of the taxonomic node, we extract the words which compose the concept name, and also the words occurring in the name of its direct super-class. We also collect the words from the names of relations and properties for which the concept plays the role of domain, as well as the words in the concept names to which these relations point, i.e. the range concepts of the relations. Usually the words in the OWL descriptions are agglutinated, so we apply the rule of thumb that a new word begins with an uppercase letter (the camel case convention) or after an underscore or a dash.

The semantic content features of the vector representation of one concept document are the frequencies of occurrence of the different words extracted from the OWL semistructured text document representation of the concept. The words extracted in this way can be considered as a gloss definition of the concept: “An instance of the concept *Collection* is a *Book* (direct superclass of *Collection*) which has as *parts* (relation *parts*) instances of the concept *InCollection* (range concept for the relation *parts*)”. Always the gloss of a concept defines the restrictions imposed upon its direct superclass as being the ones specific to the current concept. In the example, the restriction is that the range of the property *parts* to be restricted to the concept *InCollection*. (In the ontology of the example, the property *parts* is defined as having *Part* as range concept, for which *InCollection* is only one of its subclasses.) The concepts in this example can be recognized in Figure 3, which illustrates the taxonomy of one of the ontologies actually used in our experiments reported in section V. Some of the OWL concept definitions, including the ones used in our experiments, also have a comment in natural language. This again plays the role of a gloss definition. For the example concept *Collection*, the gloss comment is “*A book that is collection of texts or articles*”.

The first ontology concepts treated as documents are mapped to the second ontology classes (concepts). The classification algorithm proceeds by “populating” the taxonomy of the second ontology with the first ontology concept documents. The *Enrich-GHSOM* neural network drives a top-down hierarchical classification of the first ontology concepts along with the taxonomy branches of the

second ontology. Every ontology 1 concept is associated to one node of ontology 2.

In order to use our Enrich-GHSOM neural network to induce such a classification behavior, a symbolic-neural translation is first done by parsing a textual representation of the second ontology taxonomy, which has the form of *is a(concept, superconcept)* assertions. The result of this parsing is the initial internal tree-like state of the neural network, which mirrors the taxonomy of the second ontology. In order for the initialized network to be able to classify ontology 1 concept documents into this (ontology 2) taxonomic structure, a representation as a numerical vector is needed for each node in this taxonomy. This node vector plays the role of predefined initial weight vector for the neural network (see section III-C). Actually, for each of the second ontology nodes, we define this vector as being the document vector representation of the concept associated to the node.

During the top-down Enrich-GHSOM classification process, vector similarity computations take place between the vector representations of the classified documents (which represent concepts of ontology 1) on one hand, and the vector representations of the documents which represent the taxonomic nodes (of ontology 2) traversed by the ontology 1 concepts on the other hand. The union of all the words used as features for the classified ontology 1 concepts and for the taxonomy 2 nodes constitutes a global vocabulary.

##### A. Vector Representation

Since Enrich-GHSOM is a neural network system, the ontology 1 concepts classified by Enrich-GHSOM and the concepts of taxonomy 2 have to be represented as vectors. In our framework, the features of the vector representation of a concept encode semantic content information in an  $\mathbf{R}^n$  vector space. Specifically, the features are the frequencies of occurrence of different words in the document representing the concept.

In such a setting, the meaning of semantically similar concepts is expressed by similar vectors in the vector space. The Euclidean distance is used in our current model to compute the dissimilarity between vectors.

The framework allows different ways to encode the frequencies of occurrence: simple *flat counts of occurrences*, the *TF-IDF weighting scheme*, and the *word category histograms (WCH)*. One of these representations, i.e. the TF-IDF (“term (or word) frequency times inverse document frequency”) weighing scheme is commonplace for text document classification settings, which is also the case for our model.

The third method from the enumeration above encodes the vector representation as a *word category histogram (WCH)*. In order to compute such histograms, first a SOM map [8] is trained having the global vocabulary of words as input data space to arrive at a *reduced set of semantic categories of words*. Words with similar meaning are clustered together by the unsupervised SOM neural network. In this SOM training, the words are represented as vectors of frequencies of their occurrence in the different concept documents. Equally like the document vectors, the word vectors are collected from

the same word/document occurrence matrix, but after transposing this matrix.

For a given document, by summing together the frequencies in document of all the words that belong to one and the same category, and merely keeping these summed frequencies of the words per categories as histogram vector features, we arrive at a reduced dimensionality for the document vector representation. In other words, the WCH histogram is obtained by counting the occurrences for words of distinct categories instead of counting the occurrences of distinct words. The categories of words are in fact semantic categories of words, as induced by training the SOM map on the entire global vocabulary of words used in the concept documents. By reducing the dimensionality of the feature vectors, the word category histograms also *reduce the data sparseness* of the vector representations in our setting.

The vector representations are sparse, since the concept documents are small, and consequently each document contains only a few of the global vocabulary of words. For the rest of the words, the document vector has value zero as the feature corresponding to a word that is absent from the document. Besides using the WCH histograms, two other ways of reducing the number of zeros in our vector representations are the *centroid vector* and the *category vector* [10, 2]. In our current experiments, they are only used for reducing the data sparseness of the ontology 2 concept documents, in order to improve the semantic discrimination power of taxonomy 2, which plays the role of a decision tree. We have used the idea of centroid in the following way: the average vector of the vector representations of all the concepts in the sub-tree rooted by the given concept, including the root itself. Likewise, the category vector of a sub-tree is the sum of all the concept vectors of the sub-tree, normalized to unit length (unit norm).

## V. EXPERIMENTAL RESULTS

We have evaluated our ontology matching approach with the Ontology Alignment Evaluation Initiative (OAEI) benchmark data set for the bibliography domain [5, 3]. The set consists of a suite of ontologies obtained by altering an initial ontology. The alterations have been made along with different aspects, such as the class and property names, removing the properties or the instances, distorting the specialization hierarchy (i.e. the taxonomy, the subclass relations) etc. The result is a set consisting of more than 50 pairs of ontologies to be matched, where ontology 1 is one of the altered ontologies and ontology 2 is the initial ontology.

### A. Experimental Setup

The initial ontology from the OAEI benchmark suite of ontologies consists of 33 concepts. 24 of them are established into the *Reference* taxonomy, having the concept *Reference* in the root. The remaining concepts are called “special classes” by the authors of the benchmark suite. They are arranged in very small taxonomies or alone, playing rather the role of range concepts for the relations which start (as domain) from the 24 concepts in the *Reference* taxonomy.

The concept *Reference* is the most generic concept in the bibliography topic, having the meaning of “any bibliographic reference”. Figure 3 illustrates this *Reference* taxonomy, which always plays the role of “ontology 2” (initial ontology) in the experiments described below. In all of our experiments we classify the document-style “ontology 1” concepts against the *Reference* taxonomy, where ontology 1 is one of the altered ontologies.

For the experimental evaluation, we assess the mapping from ontology 1 concepts to ontology 2 concepts. We evaluate the correctness of the pairs ( $o1\_concept$ ,  $o2\_concept$ ) by comparison to a *reference alignment*. As being a benchmark suite for evaluating the ontology matching algorithms, the OAEI bibliography dataset comes equipped with a reference alignment for each of the pairs of ontologies to be matched. A reference alignment is a list of concept pairs where the correctness of the pairs has been found consensually by people.

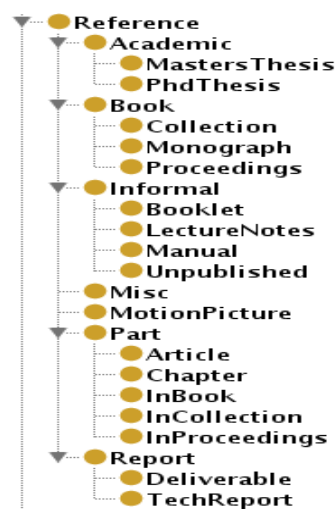


Figure 3. The *Reference* taxonomy - “ontology o2”.

### B. Evaluation Measures

The most important measure proposed in the literature for evaluating the ontology matching systems is the pair consisting of *precision* and *recall*. Their computation is based on counting the correct and the incorrect pairs of concepts from the matched ontologies. This is an “all-or-nothing” measure, and the alignments found by an ontology matching system can nevertheless contain *near misses*, i.e. pairs of concepts ( $o1\_concept$ ,  $o2\_concept$ ) that are semantically close to a correct pair. This observation led to the *relaxed precision* and *recall*, which are weighted by a measure of *overlap proximity* [5]. One of the overlap proximities proposed is the *symmetric* one [5], which weights the precision and recall by the similarity between the concept from the alignment found by the system  $S$  and the concept from the reference alignment  $R$ . The similarity between two concepts is understood here as a taxonomic similarity. It is inverse proportional with the taxonomic distance between the concepts in terms of the number of taxonomy edges which separate the two concepts in the ontology the concepts belong to. This ontology can be either of the two ontologies

<sup>1</sup> <http://oaei.ontologymatching.org/>

$o1$  and  $o2$  being matched, so, symmetrically, both elements  $o1\_concept$  and  $o2\_concept$  of the pairs are compared in the system alignment  $S$  versus the reference alignment  $R$ .

We propose here an evaluation measure inspired from the area of semantic classification (like ontology enrichment and named entity classification): the *learning accuracy* [6, 2, 10, 1, 13]. By choosing this measure, we consider correct semantic classifications with different levels of detail. For instance, the named entity *Tom* can be mapped to the concept *cat*, *feline*, *carnivore*, *mammal*, or *animal* with different levels of detail, as a consequence of different hypernym-hyponym taxonomic distances between the concept chosen by the system and the correct one. Consequently, being a “near miss” measure (as opposed to “all-or-nothing”), and also by weighting the semantic closeness by the same measure of taxonomic distance, the learning accuracy is in agreement with the relaxed precision and recall proposed in the ontology matching literature. The only difference is that, somehow non-symmetrically, we measure the taxonomic similarity between the system alignment  $S$  and the reference alignment  $R$  only in what concerns  $o2\_concept$ , so only for the second component of the pairs in the two compared alignments  $S$  and  $R$ . This is because our ontology matching system classifies semantically any given (fixed)  $o1\_concept$  against the ontology  $o2$ .

For a given concept  $o1\_concept$ , let  $s$  be the concept (from ontology  $o2$ ) assigned by the system, and  $r$  be the correct concept according to the reference alignment. Then the learning accuracy is the average over all the classified concepts  $o1\_concept$  (from ontology  $o1$ ) of the function  $LA(s, r)$ , where the function  $LA$  is defined as in [6]:

$$LA(s, r) = \frac{\delta(top, a) + 1}{\delta(top, a) + \delta(a, s) + \delta(a, r) + 1} \quad (1)$$

Where  $top$  is the root of the taxonomy, and  $a$  is the least common subsumer of the concepts  $s$  and  $r$  (i.e. the most specific common hypernym of  $s$  and  $r$ ).  $\delta(x, y)$  is the taxonomic distance between concepts  $x$  and  $y$ . The learning accuracy has a real value between 0 and 1, also interpreted as a percentage between 0 and 100%.

In the experimental results reported in this paper, the *symmetric learning accuracy* actually corresponds to the definition in formula (1), and the *learning accuracy* is a historically initial version of the learning accuracy measure introduced by [7] and also defined in [6]:

$$LA'(s, r) = \begin{cases} \frac{\delta(top, a) + 1}{\delta(top, r) + 1} & \text{if } s \text{ is ancestor of } r \\ & \text{(then also } a = s) \end{cases} \quad (2)$$

$$LA'(s, r) = \frac{\delta(top, a) + 1}{\delta(top, a) + 2 * \delta(a, s) + 1} \quad \text{otherwise}$$

We evaluated our experiments in terms of both variants of the *learning accuracy* and also a third variant of it, which is called *edge measure*. The *edge measure* counts literally the taxonomic distance between the system predicted concept  $s$  (i.e. according to the system alignment) and the one from the reference alignment  $r$ .

### C. Evaluation Results

In the experiments reported in tables I to IV, we use the following notations. *Flat* means document vectors represented as flat counts of occurrences. When *TFIDF* occurs together with *WCH*, then the TF-IDF weighting scheme is first applied to the flat count document vectors. Then the result TF-IDF vectors are converted into *WCH* histograms, thus reducing the concept vector dimensionality. *WHCM* means “big histograms”, i.e. the vector dimensionality is about 100 for the different experimental runs, whereas *WCHm* means “small histograms”, and the vector dimensions are only about 35. *Centr* and *categ* are the means to compute the vector of a taxonomic node of ontology  $o2$ , as a centroid or as a category of all the nodes in the subtree of the concept node.

Table I illustrates the results of matching the initial ontology (playing the role of ontology 2) with an ontology (acting as ontology 1) altered by substituting the concept names with synonyms, e.g. *Unpublished* became *Manuscript*, and *LectureNotes* became *CourseMaterial*. This pair of ontologies is the test nr. 205 in the OAEI benchmark suite and the test pair is equipped (built-in in the dataset) with the following reference alignment:

$$R_{205} = \{ (Manuscript, Unpublished), \\ (CourseMaterial, LectureNotes), \\ \dots \} \quad (3)$$

In the altered ontology of the test pair in Table II all the properties and relations have been removed from the initial ontology. In table III ontology 1 from the test pair has been altered by completely translating the initial ontology (except the comments) in French. *Unpublished* becomes *NonPublié* and *LectureNotes* becomes *Polycopié*. Table IV shows a test pair in which the altered ontology has an enriched taxonomic structure as compared to the initial ontology. The altered *Reference* taxonomy  $o1$  has 45 concepts compared to 24 of the initial *Reference* taxonomy  $o2$  (see Figure 4 versus Figure 3).

Across the three tests, there is a natural tendency that using the TF-IDF weighting measure improves the results. TF-IDF is a semantic-oriented scheme, which gives a higher weight to the words appearing in fewer documents. This leads to an increase in the semantic discrimination power between documents (actually between concepts in our setting, since the concepts are represented by documents). At the same time, there is also a slight improvement when representing ontology 2 concepts as category, as compared to representing them as centroid.

TABLE I. ALIGNMENT 205 – SYNONYMS

Experiment	Symmetric Learning Accuracy	Learning Accuracy	Edge Measure
flat, centr	0.329861	0.368055	2.458333
flat, WCHM, centr	0.315972	0.343055	2.333333
flat, WCHm, centr	0.430555	0.423611	1.583333
TFIDF, centr	0.427083	0.430555	1.625
TFIDF, WCHM, centr	0.40972	0.40278	1.66667
TFIDF, WCHm, centr	0.45833	0.46667	1.75
flat, categ	0.33681	0.35139	2.70833
flat, WCHM, categ	0.37847	0.39583	2.16667
flat, WCHm, categ	0.38125	0.39861	2.16667
TFIDF, categ	<b>0.5</b>	<b>0.5</b>	<b>1.375</b>
TFIDF, WCHM, categ	0.43403	0.4375	1.70833
TFIDF, WCHm, categ	0.48958	0.49444	1.54167

TABLE II. ALIGNMENT 228 – NO PROPERTY

Experiment	Symmetric Learning Accuracy	Learning Accuracy	Edge Measure
flat, centr	0.30903	0.35556	2.58333
flat, WCHM, centr	0.30903	0.35556	2.58333
flat, WCHm, centr	0.37847	0.38889	1.91667
TFIDF, centr	0.5625	0.5625	1.20833
TFIDF, WCHM, centr	0.40972	0.40972	1.625
TFIDF, WCHm, centr	0.38542	0.40972	2
flat, categ	0.39792	0.39444	2.54167
flat, WCHM, categ	0.39583	0.43056	2.25
flat, WCHm, categ	0.39722	0.4125	2.29167
TFIDF, categ	<b>0.76389</b>	<b>0.72917</b>	<b>0.79167</b>
TFIDF, WCHM, categ	0.59792	0.59444	1.33333
TFIDF, WCHm, categ	0.38681	0.40417	2.08333

The very best results attained (76.4%) are in the case when the properties are suppressed, whereas the second best accuracy corresponds to the expanded taxonomy (63.2%). This is not a surprise, since in both cases the altered ontology keeps the concept names the same as in the initial ontology. Finally, the worst accuracy (a maximum of 47.2%) is obtained when the alteration means a translation to French. In this case the semantic classifier, which is the heart of our ontology matcher, had to face the cross language problem.

TABLE III. ALIGNMENT 206 – FOREIGN NAMES

Experiment	Symmetric Learning Accuracy	Learning Accuracy	Edge Measure
flat, centr	0.35764	0.35556	1.91667
flat, WCHM, centr	0.35417	0.36806	2.08333
flat, WCHm, centr	0.375	0.36806	1.75
TFIDF, centr	0.38889	0.38194	1.66667
TFIDF, WCHM, centr	0.39931	0.39028	1.70833
TFIDF, WCHm, centr	0.4375	0.4375	1.54167
flat, categ	0.32431	0.33472	2.20833
flat, WCHM, categ	0.38542	0.41667	2.08333
flat, WCHm, categ	0.38889	0.40278	1.95833
TFIDF, categ	0.43056	0.43056	1.58333
TFIDF, WCHM, categ	0.42014	0.41806	1.66667
TFIDF, WCHm, categ	<b>0.46528</b>	<b>0.47222</b>	<b>1.54167</b>

TABLE IV. ALIGNMENT 238 – EXPANDED HIERARCHY

Experiment	Symmetric Learning Accuracy	Learning Accuracy	Edge Measure
flat, centr	0.37153	0.40278	2.33333
flat, WCHM, centr	0.43403	0.44444	1.875
flat, WCHm, centr	0.41667	0.43056	1.95833
TFIDF, centr	0.58333	0.56944	1.16667
TFIDF, WCHM, centr	0.51042	0.50694	1.41667
TFIDF, WCHm, centr	0.48056	0.48056	1.66667
flat, categ	0.38889	0.40556	2.45833
flat, WCHM, categ	0.46806	0.46944	2.04167
flat, WCHm, categ	0.39583	0.43056	2.25
TFIDF, categ	0.57639	0.5625	1.20833
TFIDF, WCHM, categ	<b>0.63194</b>	<b>0.62639</b>	<b>1.16667</b>
TFIDF, WCHm, categ	0.47292	0.49861	2.04167

## VI. CONCLUSIONS AND FURTHER WORK

We have presented an unsupervised top-down neural network based model for class based ontology matching. The matching is cast as a document classification problem, where the concepts of the two ontologies being matched are considered as text documents. The matcher provides good alignment results, which means a reduced effort required from a user assistant to fix the alignments found by the system.

As future work, we will use other data sets from the Ontology Alignment Evaluation Initiative, such as real life expressive ontologies in the anatomy domain, or large thesauri in the agriculture domain. For the sake of symmetry

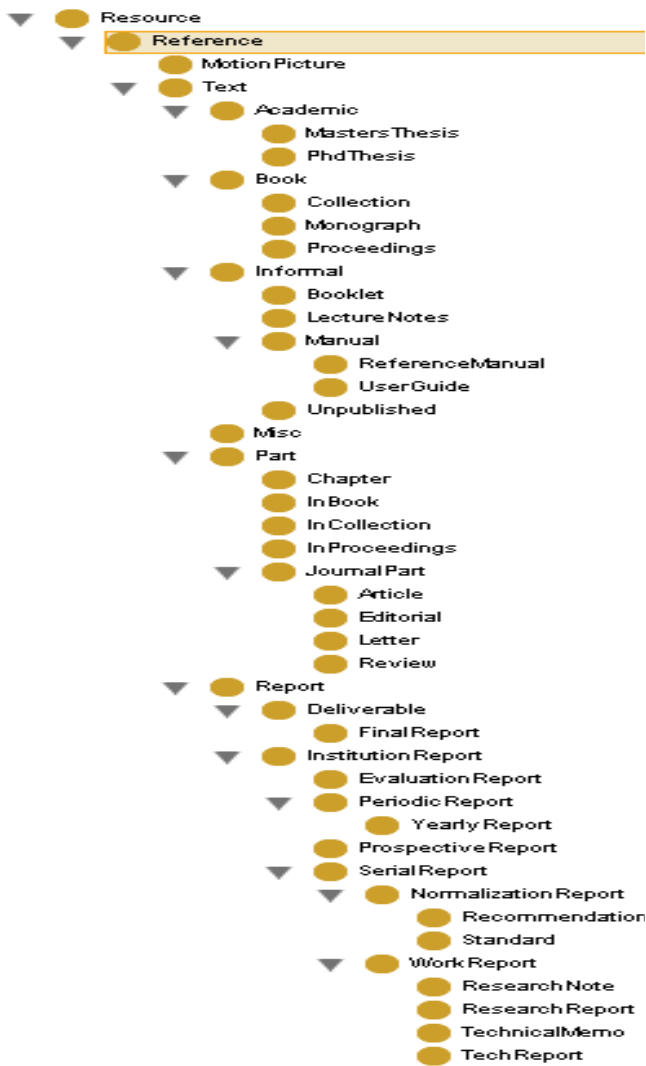


Figure 4. The enriched *Reference* taxonomy from OAEI benchmark test pair nr. 238.

and a perfect compliance with the relaxed precision and recall used in the literature [5], we will interchange the role of “ontology 1” and “ontology 2” as defined in our setting. We will compute the learning accuracy of classifying ontology 1 concepts against the taxonomy of ontology 2, then interchange the two ontologies and compute again the learning accuracy. The final measure for the quality of

matching will be the average of the learning accuracy of the two symmetric alignments found by the matcher.

#### ACKNOWLEDGMENTS

We are grateful to the anonymous reviewers for the very useful comments. Part of this work was supported by the grant ID\_170/672 from the National Research Council of the Romanian Ministry for Education and Research.

#### REFERENCES

- [1] E. Alfonseca and S. Manandhar, “Extending a lexical ontology by a combination of distributional semantics signatures”, in A. Gómez-Pérez, V.R. Benjamins, eds., *13th EKAW Conference, LNAI*, Springer, 2002, pp. 1-7.
- [2] P. Cimiano and J. Völker, 2005, “Towards large-scale, open-domain and ontology-based named entity classification”, *RANLP’05 Conference*, 2005, pp. 166-172.
- [3] J. David and J. Euzenat, “Comparison between ontology distances (preliminary results)”, in A. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. Finin, K. Thirunarayan, eds., *The semantic web, Lecture notes in computer science 5318*, 2008, pp. 245-260.
- [4] M. Dittenbach, D. Merkl, and A. Rauber, “Organizing and exploring high-dimensional data with the Growing Hierarchical Self-Organizing Map”, in L. Wang, et al., eds., *1st International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 2, 2002, pp. 626-630.
- [5] J. Euzenat and P. Shvaiko, *Ontology Matching*, Springer-Verlag, Heidelberg, 2007.
- [6] M. Grobelnik, P. Cimiano, E. Gaussier, P. Buitelaar, B. Novak, J. Brank, and M. Sintek, “Task description for PASCAL challenge. Evaluating ontology learning and population from text”, 2006.
- [7] U. Hahn and K. Schnattinger, “Towards text knowledge engineering”, *15th AAAI Conference and 10th IAAI Conference*, 1998, pp. 524-531.
- [8] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero, and A. Saarela, “Self-organization of a massive document collection”, *IEEE Transactions on Neural Networks*, **11**, 3, 2000, pp. 574-585.
- [9] R. Levy, J. Henriksson, M. Lyell, X. Liu, and M.J. Mayhew, “Ontology matching across domains”, *Workshop on Agent-based Technologies and applications for enterprise interOPerability, AAMAS Conference*, 2010.
- [10] V. Pekar and S. Staab, “Taxonomy learning – factoring the structure of a taxonomy into a semantic classification decision”, *COLING’02 Conference*, 2002, pp.786-792.
- [11] M. Sabou, J. Gracia, S. Angetou, M. d’Aquin, and E. Motta, “Evaluating the Semantic Web: a task-based approach”, The 6th International Semantic Web Conference and the 2nd Asian Semantic Web Conference, Busan, Korea, 2007, pp. 423-437.
- [12] K. Sivashanmugam, K. Verma, A. Sheth, J. Miller, “Adding semantics to Web services standards”, *ICWS03 Conference*, 2003.
- [13] H. F. Witschel, “Using decision trees and text mining techniques for extending taxonomies”, *Learning and Extending Lexical Ontologies by using Machine Learning Methods, Workshop at ICML-05*, 2005, pp. 61-68.



# An Adaptive Virtual Machine Replication Algorithm for Highly-Available Services

Adrian Coleșa

Computer Science Department  
Technical University of Cluj-Napoca, Romania  
Email: adrian.colesa@cs.utcluj.ro

Bica Mihai

Computer Science Department  
Technical University of Cluj-Napoca, Romania  
Email: bicamihai.m@gmail.com

**Abstract**—This paper presents an adaptive algorithm for the replication process of a primary virtual machine (VM) hosting a service that must be provided high-availability. Running the service in a VM and replicating the entire VM is a general strategy, totally transparent for the service itself and its clients. The replication takes place in phases, which are run asynchronous for efficiency reasons. The replication algorithm adapts to the running context, consisting of the behavior of the service and the available bandwidth between primary and backup nodes. The length of each replication phase is determined dynamically, in order to reduce as much as possible the latencies experienced by the clients of the service, especially in the case of a degraded connectivity between primary and backup nodes.

We implemented our replication algorithm as an extension of the Xen hypervisor's VM migration operation. It proved better than its non-adaptive variants.

**Index Terms**—high-availability, virtualization, replication, asynchronous, adaptive

## I. INTRODUCTION

THE AVAILABILITY of a service is given by the proportion of time that service is perceived by its clients as functioning according to its specifications, in both normal and abnormal conditions, the latter ones being determined for example by electric power outages, hardware dis-functionalities or software attacks [1], [2].

The high-availability requirement of a service emerges from the fact that its clients need a permanent access to the service. The unavailability of some services would have a negative impact for their clients, like in case of banking institutions, telecommunication companies, military applications or even hospitals. This is the main reason the research in this field has received a great attention in the latest two decades [1], [3], [4].

We will call in this paper the way a service is made highly-available *high-availability strategy*. The software infrastructure needed to provide high-availability for that service will be named *high-availability* or *protection system* and the service itself will be referred to as the *protected service*.

Most existing solutions require design changes to hardware or software components of the protection system, because they are based on special properties of the service they support [5], [6], [7]. Being integrated within the service, the main advantage of such solutions is their efficiency. Their main problem though is that any change in the service's properties requires the redesign and reimplementation of the whole

system in order to maintain initial requirements. Also, they cannot be applied to services that cannot be reimplemented.

Other solutions try to provide high-availability independently of the service. Some of them [8], [9] are capturing the state of the service at the application level and are based on the record-and-replay technique. They are dependent on the operating system the service is running on and in general does not support nondeterministic behavior of the service. Another strategy is used by [10], [11], [12], [13], [14], [15], [16]. They run the service in a virtual machine and replicate the entire virtual machine. Such a strategy can be applied to any service and provides transparency for both service's clients and the service itself. Yet, the generality advantage of the service virtualization results in the solution not being efficient for any type of service. Also, the existing solutions do not adapt to the service's specific properties or running behavior, nor even to other environment characteristics.

In this paper we describe an improved variant of the virtual machine replication algorithm proposed in [16]. The resulting algorithm is adaptive to different environment changes during the runtime of the protected service. It dynamically calculates each replication phase's duration based on the characteristic of locality of memory changes of the protected service and the number of output network packets generated during runtime. The algorithm also takes into account the available bandwidth between the primary and backup nodes. The resulted protection system aims to reduce the network traffic in case of degraded connectivity and still maintain the latencies experienced by the service's clients as close as possible to the required values. In cases of normal condition the algorithm tries to increase the efficiency of the CPU usage on the VM running the service.

We implemented the proposed algorithm over the migration mechanism of the Xen hypervisor [12]. We run the services we want to make highly-available on a Linux distribution. The tests we performed proved our adaptive algorithm's efficiency relative to its non-adaptive variants.

## II. HIGH-AVAILABILITY STRATEGY AND SYSTEM ARCHITECTURE

The high-availability system we use is the one proposed in [15], [16]. This paper contributes to the replication algorithm used by the system. This section briefly describes the general



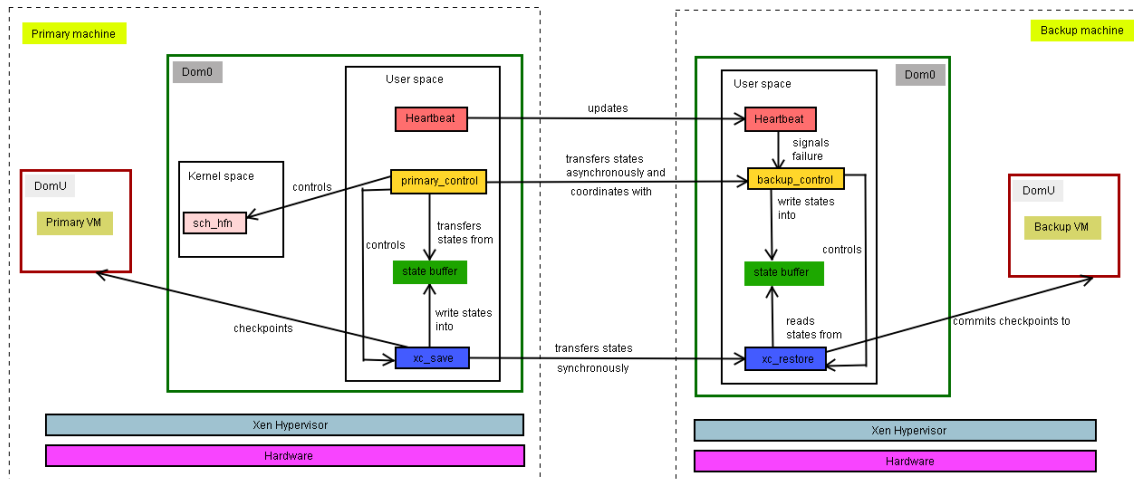


Fig. 1. Highly-Available System Architecture Built on the Xen Hypervisor

architecture and main functional characteristic of the protection system, as prerequisites for understanding the adaptive replication algorithm.

We considered centralized services that are already in use and cannot be modified in order to increase their availability level. High-availability implies tolerance to the service's components failure, which can be provided only based on some degree of redundancy. Though, the fact that the service is centralized means that it is implemented and provided by a single running server, which constitutes a single point-of-failure. In order to improve its high-availability, we must create some replicas of the service's server and run them altogether in a distributed manner. The replication strategy we consider is the passive one, with a primary and more backup replicas. It provides total transparency for the service's clients [1].

The only way of replicating a centralized service without modifying it is to place the service's server in a virtual machine (VM) and replicate the entire virtual machine [10], [14], [15]. This method is general because it can be applied to any service and operating system the service runs on.

Our solution is based on VM replication over the Xen hypervisor and extends the live VM migration operation supported by Xen. Xen allows several guest operating systems to execute on the same computer hardware in different VMs. The first VM which is booted, called *Dom0* (Domain 0), is a privileged one. It runs a Linux kernel and is used by the Xen hypervisor to interact with the hardware. This way Xen is independent of the hardware, letting this responsibility to the Linux kernel in *Dom0*. The other VMs are called *DomU* (Domain Unprivileged). User space Xen tools in *Dom0* allows the user to gain control rights over the other guest operating *DomU* VMs. To implement our high-availability system we modified some of the processes which are controlling *DomU* domains. We run the service we want to make highly-available in a *DomU* VM on a primary node and replicate that VM on a backup node in a corresponding backup *DomU* VM. The replication is controlled by processes placed in *Dom0* VM

on both nodes. The architecture of the resulting system is illustrated in Figure 1.

The main components of our system are the two user-space processes *primary\_controller* and *backup\_controller*, which controls the replication mechanism. They interact with the Xen's *xc\_save* and *xc\_restore* components, which normally implemented the Xen live migration operation, but which we modified to transform the one-time migration in a continuous replication of the VM machine running on the primary node to the backup one.

The replication takes place in phases, each one consisting in the following ordered stages: (1) *running*, lasting  $t_R$ , during which the primary VM is run, accepting inputs, but having its network outputs blocked, (2) *saving*, lasting  $t_S$ , when the VM is suspended and its state saved in the *state\_buffer*, (3) *transfer*, lasting  $t_T$ , when the previously saved VM state are transferred from the primary node on the backup one, (4) *output release*, lasting  $t_O$  and corresponding to the act of releasing the outputs of the VM corresponding to its already replicated state.

The running stage is controlled by our system and its length  $t_R$  is calculated dynamically for each stage in a way described in details in the next section. The VM state saved during a saving stage consists in the memory pages modified from the last saving point and current values of the VM's CPU registers. The corresponding  $t_S$  depends on the number of modified pages. Being small relative to the other times, we considered it constant. The transfer time  $t_T$  is dependent on the size of the VM's saved state and the available bandwidth between the primary and backup nodes. These two terms cannot be directly controlled by our system. Though, we will see below how we can indirectly reduce the transfer time of the overall replication process, trying to replicate as few pages as possible, when network bottlenecks occur.

Blocking the primary VM's outputs until its corresponding state is replicated, provides for its transparent replacement by

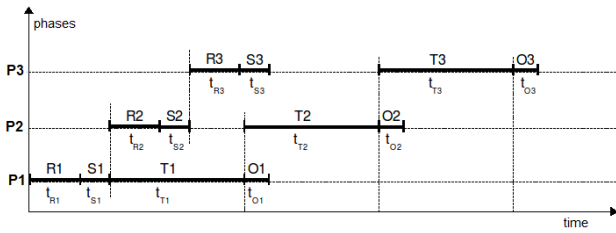


Fig. 2. Asynchronous Replication Strategy

a backup VM, in case it crashes. The kernel-space module `sch_hfn` controls the blocking and releasing of the primary VM's network outputs. The crashes of the primary VM are detected using a simple *heartbeat* mechanism.

The replication process is asynchronous, parallelizing as many replication stages as possible, in order to reduce the latencies experienced by the service's clients. Actually, only some of the stages can be performed in parallel. The way different stages of successive replication phases overlap is illustrated in Figure 2. We can note that during the transfer and output release stages of previous phases, running and saving stages of the following phases can take place.

### III. THE ADAPTIVE REPLICATION ALGORITHM

Experimental results in [16] have led us to the conclusion that an adaptive replication algorithm, which would calculate dynamically the running time  $t_R$  of each replication phase, based on the current runtime environmental conditions, could provide improvements in terms of decreasing the network load, increasing the CPU usage efficiency, and reducing the latencies perceived by the clients (response time).

We must note however that not all the mentioned parameters could be optimized simultaneously. For instance, the strategy of getting a better CPU usage could be in contradiction with the one trying to decrease the response time. We will see below the way our adaptive algorithm trades between these factors.

We denote the *CPU usage efficiency* by  $\eta$  and defined it by the relation:

$$\eta = \frac{t_R}{t_R + t_S} \quad (1)$$

The smaller the running time and, implicitly, more frequent the saving stages, the lower the efficiency of the CPU usage, because most of the time the CPU would be used by the saving process, which is time consumed additionally by the protection system and not by the service. Equation (1) can be thought as the fraction between the running times in cases the protection system is deactivated and respectively, activated.

One condition required to our replication algorithm is to try to keep  $\eta$  above a certain value  $\eta_{min}$ . For instance having a  $t_S = 50ms$ , in order to provide a  $\eta_{min} = 90\%$ , the algorithm must use a running time  $t_R$  of at least  $450ms$ . This could be too much for a response time of a client request. Just to have an idea of this problem, we will calculate the maximum delay our protection system could introduce in a client request's response

time in the particular favorable case, when we consider having enough network bandwidth available in each replication phase, such that the transfer stage of a phase could start immediately after the saving stage of that phase. This means that for any phase  $i \geq 1$ ,  $t_{T_i} \leq t_{R_{i+1}} + t_{S_{i+1}}$ . For simplicity we also consider the time of all similar stages to be identical in all phases and such we will have the following formula for the delay introduced by our system in the response time:

$$D_{max} = (n + 2)t_S + t_T + t_R \quad (2)$$

where  $n$  is the number of phases needed for the service to handle that client request and generate a response in terms of one or more output network packets.

Coming back to the above example, for  $t_R = 450ms$  and  $t_S = 50ms$ , considering a  $1Gbs$  network bandwidth, a VM state to be replicated of about  $10000pages$  of  $4KB$  each and  $n = 1$ , the maximum delay will be  $D_{max} \approx 855ms$ , which could be really too long for many of the client applications. So the running time should be decreased in such cases. In the case of not having enough bandwidth available, the transfer time becomes the dominant factor in the delay formula, and the result could be even worse.

From the example above we can understand intuitively the way our replication algorithm will act. Firstly, it will try to keep the response time as small as possible, because this is directly perceived by the service's clients. This depends on the size of the VM state that must be replicated and also on the available bandwidth. As long as the replication algorithm can provide

$$D_{max} \leq D_{max}^{req} \quad (C1)$$

where  $D_{max}^{req}$  is the maximum acceptable delay required by service's clients, it increases the running time of the current replication phase, in order to get a CPU usage efficiency as close as possible to the required  $\eta_{min}$ . This case is what we will consider to correspond to the *normal functionality* of the system. In case, the network bandwidth is not enough (e.g. network overloaded or a large VM state) to transfer in time the VM state, the replication algorithm will calculate the running time of the each replication phase such that to minimize the delay  $D_{max}$ . Such a situation we will call *abnormal* or *degraded functionality*.

The parameters taken into account to establish each  $t_{R_i}$  are: (1) the protected service's behavior regarding the locality of memory changes, measured as the number of new distinct pages that will be modified in the next future in current phase  $i$ , noted  $\Delta^{add}$ , (2) the available current network bandwidth between the primary and the backup nodes, noted  $B_{crt}$ , (3) the output network packets generated on the primary VM.

The first two parameters are predicted each  $\tau ms$  (a period established by the system administrator) using the exponential average. The third parameter consists in both the number of the output packets generated until the current moment, noted  $P_{out}$ , and that of the packets that will be generated in the next  $\tau ms$ , noted  $P_{out}^{add}$ .

The ideas the algorithm is based on are the followings: when there are output packets waiting to be released, it tries to reduce the running time, in order to reduce the response time of client requests. Reducing the  $t_R$  will result in a smaller VM state that must be replicated, so a smaller transfer time, so again a smaller delay in the response time. Nevertheless, the number of replicated states will be bigger, comparing with the case of longer replication phases. This means, firstly, a worse CPU usage and, secondly, a possible greater total network load. The latter especially occur when the replicated VM and implicitly the protected service manifests a greater locality of memory changes, i.e. modifies approximately the same set of pages in consecutive time intervals. This could result in the same set of pages being replicated more times consecutively, corresponding to more consecutive saved VM states. Such a situation must be avoided when the network is overloaded and the transfer of a saved state cannot be started immediately after the saving stage finishes.

Based on the above considerations, the first thing the algorithm does is to check whether there are output packets waiting to be released. If there is no one, then it makes no sense terminating the current phase, especially if there is no bandwidth available, because there will be no delay perceived by the service's clients. Furthermore, the shorter the running stages, the greater the number  $n$  of replication phases a client request handling lasts and, consequently, a larger response time.

The second thing the algorithm considers is the available network bandwidth. If there is enough available, then the current phase could be terminated and a new one started, because the replication of the current one could be started immediately. If not, maybe it could be more appropriate to enlarge the current phase (actually its running stage), especially if the service behaves locality of memory changes, just to avoid a greater amount of memory replicated and a greater overall delay.

The exact decision the algorithm takes at one moment regarding the continuation or finish of the current running stage (and implicitly current phase) depends on the number of output packets and the number of VM's memory pages already modified, i.e. the size of the VM state that must be replicated. It evaluates, based on the currently measured and predicted values, if the average delay of the output packets would be greater if the current phase is continued or terminated. The best case is always chosen. If the decision is to continue the current phase, then a new evaluation is made after a  $\tau$  period.

The detailed description of the algorithm will illustrate the above ideas. Algorithm 1 describes the strategy followed in case of normal functionality, actually when enough bandwidth is available. It returns *TRUE* if the current phase must be terminate and *FALSE* otherwise. Also, it establish the time after which a new estimation will be performed. Firstly, the algorithms checks if continuing the current phase could result in exceeding the  $D_{max}^{req}$ . Condition C2 take into account the currently introduced maximum delay  $D_{max}$  and also the additional time to transfer new distinct pages that will be

modified if continuing the current phase. Thus, we have the following relation for C2:

$$D_{max} + \frac{\Delta_i^{add}}{B_{crt}} > D_{max}^{req} \quad (C2)$$

In case the required delay is not exceeded, the algorithm checks if the state buffer will be filled or not taking into account the current size of the buffer, the page already modified, i.e. the current size of the VM state  $\Delta^i$  that must be replicated and the number of new distinct pages that will probably be modified in the next period  $\tau$ . Thus, the condition C3 can be express like:

$$size(state\_buffer) + \Delta_i + \Delta^{add} - B_{crt}\tau > MAX\_BUF \quad (C3)$$

The term  $\Delta^{add} - B_{crt}\tau$  represents the number of pages that will be actually accumulated to the current state in the next period. If the buffer is not to be filled, the next checking is whether the required CPU usage efficiency  $\eta_{min}$  is met or not. Condition C4 is expressed based on Equation (1):

$$t_{R_i} \geq \frac{\eta_{min} t_S}{1 - \eta_{min}} \quad (C4)$$

In case the  $\eta_{min}$  is not reached, the current phase is continued. Otherwise, the algorithm does not decide to terminate the current state. If the average delay of currently waiting output could be greater than the estimated average delay of the overall output packets, then the current state will be extended. Intuitively, this could happen if in the immediate future (at least next  $\tau$  ms) the number of new generated output packets will be great relative to the number of new distinct modified pages that will be added to the current VM state, i.e. the VM manifest a good locality of memory changes. The condition C5 could be express using the formula:

$$P_{out}(\frac{\Delta^{add}}{\tau} + B_{crt}) \leq P_{out}^{add} \frac{\Delta^{com}}{\tau} \quad (C5)$$

where  $\Delta^{com}$  is the number of pages that would belong to both current phase and a possible next one, if the current one would be terminated.

Algorithm 2 describes the steps taken by the protection system in case the required maximum delay in the client response time cannot be provided. Some decisions are similar with those in Algorithm 1, so we detail only the others. Firstly, the algorithm checks if there is available bandwidth. Actually, it checks if the transfer of the current state can start immediately or not. So, the condition C6 can be written:

$$t_S \geq size(state\_buffer)/B_{crt} \quad (C6)$$

In case the current state cannot be replicated immediately, the algorithm try to find the best solution to get a minimum overall delay. In essence, this depends on the locality of memory changes manifested by the replicated VM and the number of generated outputs. The state buffer is tested for three other distinct situations, different by that of empty buffer

```

task VM_replication;
function normal_functionality () : boolean
   $t_{chk} \leftarrow \tau$ ;
  if “it is possible to exceed  $D_{max}^{req}$ ” then
    | return TRUE;
  else
    if “the state buffer is full” then
      | return TRUE;
    else
      if “required  $\eta_{min}$  provided” then
        if “it is more efficient extending the
        current phase than starting the next one”
        then
          | return FALSE;
        else
          | return TRUE;
        end
      else
        | return FALSE;
      end
    end
  end

```

**Algorithm 1:** Replication algorithm in case of *normal functionality*

tested by condition C6: (1) buffer full, tested by condition C3, (2) available space above a safe threshold, tested by condition C8, and (3) available space below a safe threshold, i.e. the trend is to rapidly fill the state buffer, tested by condition C9.

We will not insist anymore on conditions C8 and C9, because they are very similar with C3. We only note that in case the buffer is almost full, the next evaluation moment will not be as usual after  $\tau$  ms, but after the time needed to transfer the saved VM state with the maximum size.

Condition C7 is very similar with C5, the difference consisting in a new factor taken into account: the time the algorithm has to wait for enough space in state buffer to save the current VM state, in case the current phase is terminated, before starting a new replication phase. It can be written like:

$$P_{out} t_T^{add} \leq P_{out}^{add} (t_T^{com} - t_{wait}) \quad (C7)$$

where  $t_T^{add}$  is the transfer time for the  $\Delta^{add}$  pages,  $t_T^{com}$  is the transfer time for the  $\Delta^{com}$  pages, and  $t_{wait}$  is the time the algorithm must wait for sufficient space in state buffer to save the current VM state.

#### IV. TESTS AND RESULTS

The following tests were made on two computers having the same configuration Intel Core 2 Duo 2.7GHZ processor, 2GB of RAM, 80GB of hard disk space and a 100Mbit Ethernet interface. Xen 3.3.1 was installed on each machine, with the Linux kernel 2.6.26-1-xen-686, running Ubuntu 9.04 as primary operating system for Dom0. In DomU was installed Linux Debian Lenny.

```

task VM_replication;
function degraded_functionality () : boolean
   $t_{chk} \leftarrow \tau$ ;
  if “there is available bandwidth” then
    | return TRUE;
  else
    if “state buffer is full” then
      if “it is more efficient extending the current
      phase than waiting for saving space in state
      buffer and starting the next phase” then
        |  $t_{wait} \leftarrow \frac{\max(0, \text{size}(\Delta_i) + \text{size}(\text{state\_buffer}) - \text{MAX\_BUF})}{B_{crt}}$ ;
        |  $t_{chk} \leftarrow \min(t_{wait}, \tau)$ ;
        | return FALSE;
      else
        | return TRUE;
      end
    else
      if “it is more efficient extending the current
      phase than starting the next one” then
        | return FALSE;
      else
        if “not enough space in state buffer” then
          if “state buffer is almost full” then
            |  $t_{chk} \leftarrow \max(t_{T_k} | \Delta_k \in$ 
            |  $\text{state\_buffer}) - t_S$ ;
            end
          end
          return TRUE;
        end
      end
    end
  end

```

**Algorithm 2:** Replication algorithm in case of *degraded functionality*

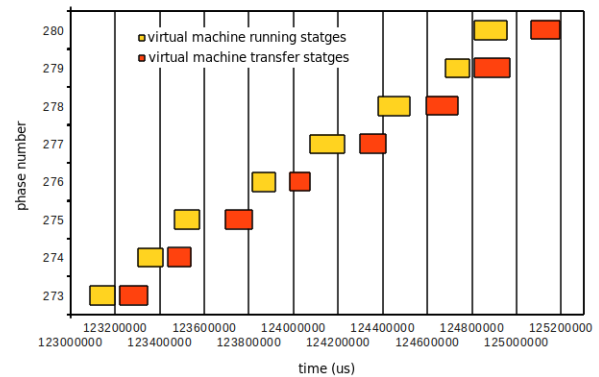


Fig. 3. Stages superimposing under usual stress

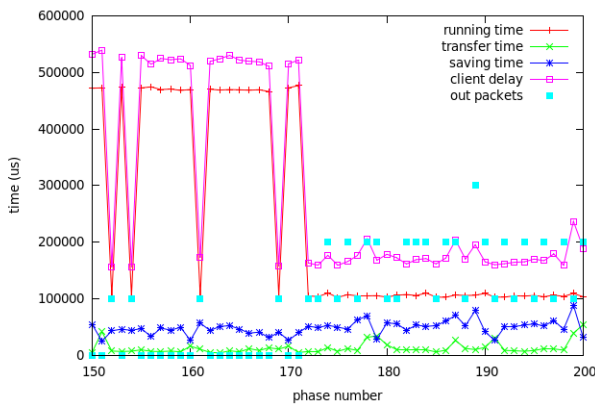


Fig. 4. Client delay reducing on increasing output packets number

In Figure 3 we illustrated a Gantt chart to show how some of the consecutive phases are superimposing in practice and reduce overall client delay. In Figure 2 we see how the phases are superimposing based on theoretical formulas. Our practical results are thus similar. If the replication of the system would be made using only the asynchronous strategy without the adaptive algorithm, the length of the running stages would be of fixed length as they were configured by the user or the system administrator. In the Figure 3 on the  $x$  axis is represented the running time in micro seconds. Time 0 corresponds to the starting of the replication process. The first time represented is the second 123 and the distance between the vertical grids is  $200ms$ . In 4 phases out of 8 there is an overlapping between the transmission stage of the previous phase and the running stage of the next phase. The process which is running in background is modifying around 350 pages per each iteration. If the replication strategy would use an synchronous algorithm the total running time would be larger because the phases would succeed one after another.

In Figure 4, starting with phase 174 a large number of inputs (client requests) was simulated using the Linux command `ping target -i 0.1`. This command makes the virtual machine to generate around 2 or 3 output packets in each phase and simulates the fact that 9 clients are making a request each second. As seen from the figure, when there are output packets, our algorithm instantly reduces the running time in order to create a very small delay to the clients, around  $200ms$ , when the replication algorithm is configured to run the virtual machine for at least  $100ms$  in order to ensure the required CPU efficiency. In this graphic we can also see that if there is a small number of dirtied pages, the saving time of the state to a local buffer in memory is greater than the transfer time of the state to the network. This is because dirtied pages are generated in a process and sent to another process using shared memory and the Linux scheduler decides when to switch between processes.

In Figures 5 and 6 are represented client delays in two different situations: (1) when there are 9 requests per second, simulated with the command `ping -i 0.1`, and (2) in the

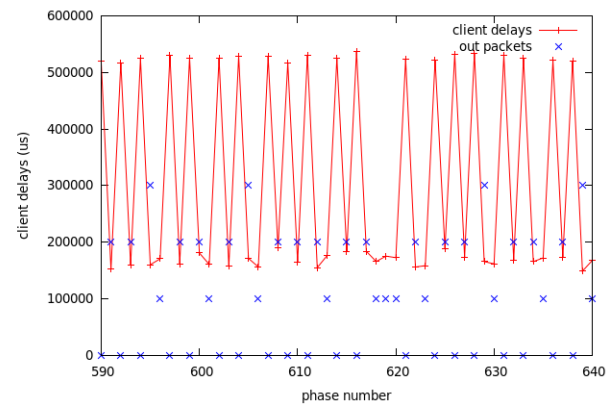


Fig. 5. Delays in context of 9 client requests per second

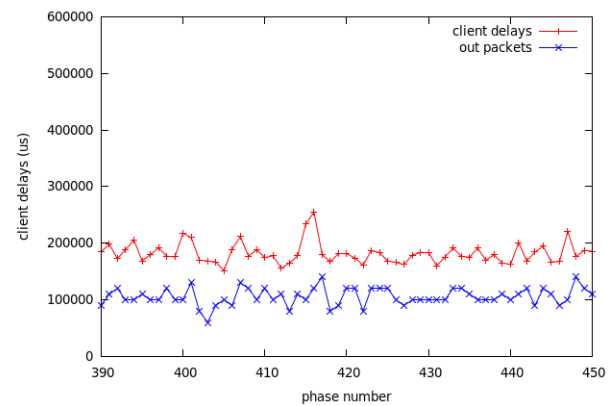


Fig. 6. Delays in context of 100 client requests per second

situation where 100 client requests are generated per second, sent by clients simulated with the command `ping -i 0.01`. Output network packets are not plotted against the  $y$  axis, which represents the time, but they are plotted against a secondary fictional  $y$  axis show there number multiplied with  $10^4$ , just to see them easily. As seen from the Figure 5, there is a quick response in how the the client delays are reduced when there are output packets. Otherwise, the client delays are not reduced. In Figure 6 the delays are less than  $200ms$  for all 100 clients, which are making one request per second.

The test in the Figure 7 shows how the system behaves when the number the output packets remains constant and the number of modified pages is increased. The number of modified pages is plotted against a fictional  $y$  axis showing their number. In the first phases, from 40 to 90, the number of modified pages is constant and around 120. From the phase 90 to 130, the number of modified pages will increase up to 4000 modified pages per phase. The page number is increased running in the VM a test program, which is modifying many pages. In this case, the system will not be able to generate a very good response time because it will run in degraded functionality, but the modified pages will have a increased locality of memory modifications, and for this



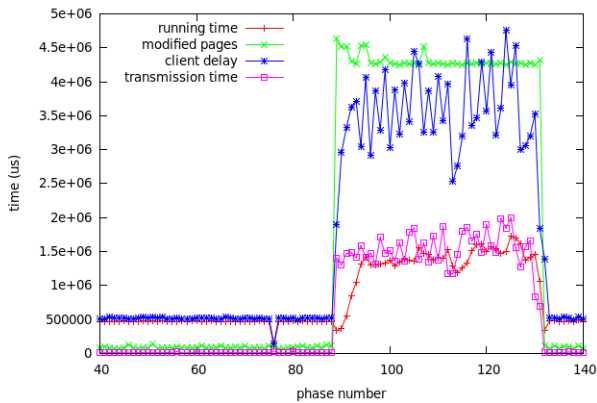


Fig. 7. Dirtied page variation, no clients connected

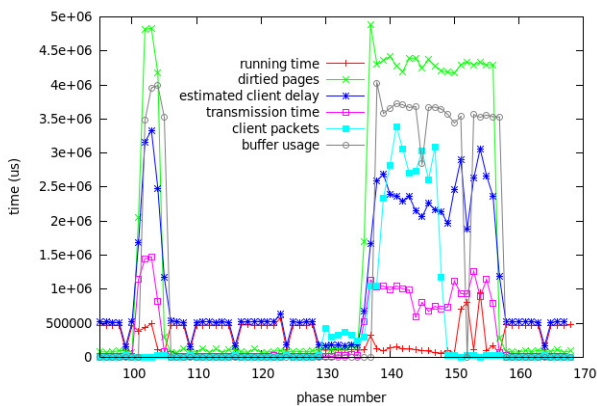


Fig. 8. Variation of running time in a complex scenario

reason we can increase the running time without generating too many additional dirtied pages. The major benefit of increased running time in this case is a better CPU usage efficiency. We can observe an increase in CPU efficiency and also while the CPU is running a phase, the previous phase is being transferred by the network interface at the same time. This is also confirmed by the fact that the transmission stage is equal to the running stage.

In Figure 8 is represented an all case scenario for our system. In this test, a program which is generating around 4800 pages per iteration is being run in two random moments. Also, while the program was running, clients connections were simulated as represented in the figure. We have to mention that the network bandwidth during the idle periods, when no program is running, is around  $1.3MiB/s$  and during the degraded functionality is around  $18MiB/s$ . When the adaptive algorithm was deactivated, the idle network traffic was around  $7MiB/s$ . This means that the adaptive algorithm reduces network traffic during idle periods by up to 5.3 times, compared to a simple asynchronous replication protocol.

Another test we made was to copy a large archive of  $125MB$  to the protected VM and then extract the data and compare the results in cases our protection system was

activated and not activated respectively. The copying speed over the network of the archive was around  $404KB/sec$ . After the file was copied the contents was extracted. The extraction period was around 50 seconds, when there were no other clients connected and around 58 seconds, when there were simulated 100 requests per second. We unzipped the same file on the system without the replication being activated and we measured 10 seconds. We conclude from this test that our replication system is 5 up to 5.8 times slower than the system with protection deactivated.

## V. RELATED WORK

Virtual machine replication based on Xen has been explored in [15], [16], Remus [14] and Kemari [17]. The main improvement our system brings over other solutions is the adaptation property of the replication algorithm. We are not aware of any other similar strategy.

The system in [16] is the most similar with our system. Actually, we developed further it. The adaptive replication algorithm is overall more efficiently than its non-adaptive variant, reducing in case of a degraded network link between primary and backup the client response time.

Remus is also very similar with our system. Actually it was introduced as a high-availability mechanism in newer versions of Xen. They reported in [14] a possible improvement consisting in dynamically modification of the rate at which the protected VM operates, in order to reduce the number of modified pages per replication phase, which would result in a reduced transfer time and, consequently, a reduced response time. Our strategy is better in cases of VM manifests high locality of memory changes, since it let the VM run at normal rate, by enlarging the replication phases. The proposed Remus optimization can be combined with ours in case the locality of memory changes and network bandwidth are very poor. What our system also makes better than Remus is that it reduces the frequency of replication phases, when there are few or no output packets, just to get a better CPU usage efficiency. Although, they have a very efficient saving method of VM states, which automatically lead to better CPU usage, but actually this is complementary to our strategy and can be integrated with it in order to get an even better efficiency.

## VI. CONCLUSION

This paper proves the fact that the asynchronous adaptive replication algorithm can improve the performance, client response time and reduce network bandwidth of high availability systems in situations where the environment changes are very often.

A drawback of our system is the buffering time. In present, the buffering time is too large and for a small number of modified pages it can be greater even than the network transfers. Some improvements can be made by implementing a better buffering technique.

## REFERENCES

- [1] R. Guerraoui and A. Schiper, "Fault-tolerance by replication in distributed systems," in *Reliable Software Technologies - Ada-Europe'96*. Springer-Verlag, 1996, pp. 38–57.
- [2] A. Bartoli and O. Babaoglu, "Constructing highly-available internet services based on partitionable group communication," 2001. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.17.218>
- [3] K. P. Birman and T. A. Joseph, "Reliable communication in the presence of failures," *ACM Trans. Comput. Syst.*, vol. 5, no. 1, pp. 47–76, February 1987.
- [4] F. Cristian, B. Dancy, and J. Dehn, "Fault-tolerance in the advanced automation system," in *EW 4: Proceedings of the 4th workshop on ACM SIGOPS European workshop*. New York, NY, USA: ACM, 1990, pp. 6–17.
- [5] T. Anker, D. Dolev, and I. Keidar, "Fault tolerant video on demand services," in *In Proceedings of the 19th International Conference on Distributed Computing Systems*, 1999, pp. 244–252.
- [6] M. Marwah, S. Mishra, and C. Fetzer, "Fault-tolerant and scalable tcp splice and web server architecture," in *SRDS '06: Proceedings of the 25th IEEE Symposium on Reliable Distributed Systems*. IEEE Computer Society, 2006, pp. 301–310.
- [7] —, "Enhanced server fault-tolerance for improved user experience," in *Dependable Systems and Networks With FTCS and DCC, 2008. DSN 2008. IEEE International Conference on*, 2008, pp. 167–176.
- [8] Y. Saito, "Jockey: A user-space library for record-replay debugging," 2005.
- [9] Z. Guo, X. Wang, J. Tang, X. Liu, Z. Xu, M. Wu, F. M. Kaashoek, and Z. Zhang, "R2: An application-level kernel for record and replay," 2008.
- [10] T. C. Bressoud and F. B. Schneider, "Hypervisor-based fault tolerance," in *SOSP '95: Proceedings of the fifteenth ACM symposium on Operating systems principles*, vol. 29, no. 5. ACM Press, December 1995, pp. 1–11.
- [11] G. W. Dunlap, S. T. King, S. Cinar, M. A. Basrai, and P. M. Chen, "Revirt: enabling intrusion analysis through virtual-machine logging and replay," *SIGOPS Oper. Syst. Rev.*, vol. 36, pp. 211–224, 2002.
- [12] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," in *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*. ACM Press, 2003, pp. 164–177.
- [13] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in *NSDI'05: Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*. USENIX Association, 2005, pp. 273–286.
- [14] B. Cully, G. Lefebvre, D. Meyer, M. Feeley, N. Hutchinson, and A. Warfield, "Remus: high availability via asynchronous virtual machine replication," in *NSDI'08: Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*. USENIX Association, 2008, pp. 161–174.
- [15] A. Coleșa and B. Marincăș, "Strategies to transparently make a centralized service highly-available," in *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP'09)*, 2009, pp. 339–342.
- [16] A. Coleșa, I. Stan, and I. Ignat, "Transparent fault-tolerance based on asynchronous virtual machine replication," in *Proceedings of The 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC '10)*. IEEE Computer Society, 2010, pp. 442–448.
- [17] Y. Tamura, "Kemari: Virtual machine synchronization for fault tolerance using domt," June 2008.



# Service Modelling for the Internet of Things

Suparna De, Payam Barnaghi  
Centre for Communication  
Systems Research, University of  
Surrey, Guildford GU2 7XH.  
United Kingdom  
Email: {S.De,  
P.Barnaghi}@surrey.ac.uk

Martin Bauer  
NEC Laboratories Europe,  
Software and Services Research  
Division, Kurfürsten-Anlage 36,  
D-69115 Heidelberg, Germany  
Email: Martin.Bauer@neclab.eu

Stefan Meissner  
Centre for Communication  
Systems Research, University of  
Surrey, Guildford GU2 7XH.  
United Kingdom  
Email: S.Meissner@surrey.ac.uk

**Abstract**—The Internet of Things envisions a multitude of heterogeneous objects and interactions with the physical environment. The functionalities provided by these objects can be termed as ‘real-world services’ as they provide a near real-time state of the physical world. A structured, machine-processible approach to provision such real-world services is needed to make heterogeneous physical objects accessible on a large scale and to integrate them with the digital world. This paper presents a semantic modeling approach for different components in an IoT framework. It is also discussed how the model can be integrated into the IoT framework by using automated association mechanisms with physical entities and how the data can be discovered using semantic search and reasoning mechanisms.

## I. INTRODUCTION

THE vision of the Internet of Things (IoT) relies on the provisioning of real-world services. The services are provided by a plethora of heterogeneous objects that are directly related to the physical world. Advancements in networking technologies and device capabilities enable a large number of physical world objects to have the communication and computation capabilities to connect and interact with their surrounding environment. The data and/or services offered by such objects can provide information about the physical world and allow interaction with it. These real-world data/services need to be defined and made available in a homogeneous way to allow integration of the data from different sources and to support autonomous reasoning and decision making mechanisms. Existing research initiatives have focussed on sensor (and actuator) middleware architectures that offer sensor measurement data services on the Web and/or at the application level. To extend this to heterogeneous physical world objects’ data, this paper identifies the following requirements: a) identification of the various possible concepts in the IoT framework and their structured representation b) an access mechanism that offers a homogeneous interface to heterogeneous IoT objects with diverse capabilities, and c) automated machine-interpretability of the various interactions and integration with existing applications. This is necessary in order to integrate the physical world objects

This paper describes work undertaken in the context of the IoT-A project, IoT-A: Internet of Things – Architecture (<http://www.iot-a.eu/public>) contract number: 257521.

with the digital world and facilitate horizontal collaboration with existing software services.

The information model presented in this paper captures the components of the IoT domain and provides a formal representation to the interactions. The paper is organised as follows: Section II presents relevant state-of-the-art in the IoT domain and sensor modeling. The proposed information models are detailed in Section III. The applicability of the models to infer associations with physical objects and to be utilized in a search framework is presented in Section IV. The implications of the modeling approach are discussed in Section V. Section VI concludes the paper and discusses the future work.

## II. RELATED WORK

Research initiatives and standardization activities in areas allied to the IoT vision have mainly focused on sensor descriptions and observation data modeling. The SENSEI project [1] aimed at realizing ambient intelligence in future networks and service environments by developing a framework of universal service interfaces for wireless sensor and actuator networks (WSANs). The core modeling concept considered in SENSEI is ‘resource’, with all sensors, actuators, and processors being modeled as resources [2]. A resource model captures resource functionalities, and where and how they can be accessed, in a conceptual view. The concrete instantiation of this information is contained in the resource description, which is published in a resource directory that acts as a service repository. Resources are described by a number of keywords. The syntax and semantics of the interfaces are captured in the advanced resource description, which is an ontology including concepts such as location, type (Sensor, Processor, Actuator), and operations of a resource. For each operation, it specifies the inputs that a resource takes in order to provide an output, the pre-conditions and post-conditions derived from invoking an operation and the temporal availability of the operation. The SENSEI resource model forms the basis of the models proposed in this paper, which are extended to encompass possible key concepts of the IoT domain.

There have been a number of works focusing on representation models for sensor data using ontologies, such as [3], [4]. OntoSensor [3] constructs an ontology-based

descriptive specification model for sensors by excerpting parts of SensorML [5] descriptions and extending the IEEE Suggested Upper Merged Ontology (SUMO) (<http://www.ontologyportal.org/>). However, it does not provide a descriptive model for observation and measurement data. The work presented in [4] proposes an ontology-based model for service oriented sensor data and networks. However, it does not specify how to represent and interpret complex sensor data. The SensorData Ontology developed in [6] is built based on Observations & Measurements and SensorML specifications defined by the OGC Sensor Web Enablement (SWE) [7].

W3C's Incubator Group on Semantic Sensor Networks (SSN) (<http://www.w3.org/2005/Incubator/ssn/>) has introduced an ontology [8] to describe sensors and sensor networks. The ontology represents a high-level schema model to describe sensor devices, their capabilities, platform and other related attributes in the semantic sensor networks and the sensor Web applications. The SSN ontology, however, does not include modeling aspects for features of interest, units of measurement and domain knowledge that are related to sensor data and need to be associated with the sensor data to support autonomous data communications and efficient reasoning and decision making processes. In fact, the SSN ontology describes sensor devices, observation and measurement data and the platform aspects; however extensions to other components in the IoT domain are not specified in the ontology.

The CSIRO sensor ontology [9] was the precursor of the W3C SSN sensor ontology. It provides a semantic description of sensors in terms of the sensor grounding (platform, dimensions, calibration, power-source and access mechanism) and operation specification (operation, process and results). Concepts for sensor measurements are not part of the ontology. Moreover, similar to the SSN ontology, concepts for domain knowledge, units of measurement, location etc. are not included. Thus, more modeling concepts are needed to link the sensor descriptions to sensor measurements and then to the observed entity in the IoT domain. Sensor observations and measurements are modeled in the SemSOS O&M-OWL ontology [10]. The key concepts modeled are observation, process, feature (abstraction of real-world entity) and phenomenon (property of a feature that can be sensed or measured). The O&M concepts are aligned to SensorML and the feature and phenomenon concepts pertain to the weather domain. A similar approach to separate the observations from the entity being observed is presented in the SEEK Extensible Observation Ontology (OBOE) [11], which has a core observation ontology, a units extension, and a further extension for domain use (coastal ecosystems). Each observation is modeled to have a measurement, which is that of an entity's characteristic. An entity is supposed to serve as an extension point into domain models, with one particular example provided for a coastal ecosystem domain. The concepts in the OBOE ontology would require to be extended to include generic features of possible IoT entities. Also, placeholders to include sensor descriptions from other ontologies would be required.

The SemSerGrid4Env project has developed a service ontology that represents sensor web services provided by a sensor grid infrastructure [12]. In that model, Web Services are classified by the datasets they expose. SemSorGrid4Env considers that datasets conform to definitions such as OGC [7] or GeoJSON (<http://geojson.org/geojson-spec.html>). The service interface is defined according to ISO 19119 standard [13] specifying service operations together with their parameters. To annotate sensor observation values gathered by services with spatio-temporal meta-data, concepts from NASA's SWEET ontology (<http://sweet.jpl.nasa.gov/>) are used. To describe the physical phenomena observed by the sensor service, the concepts 'Property' and 'FeatureOfInterest' are borrowed from SSN sensor ontology. The SemSorGrid4Env Service ontology is suitable to describe sensor services about natural phenomena. To be able to describe arbitrary 'things' including human made artifacts, a more general description is needed.

Ontology Web Language for Services (OWL-S) [14] is a minimalistic approach for describing semantic Web Services. It is a service description framework that provides both rich expressive descriptions and well-defined semantics. OWL-S provides the main attributes to describe services and their functional attributes. It describes the characteristics of a service by using three top-level concepts, namely service profile, service-grounding, and service model. The profile is meant to be published to service repositories. It offers provider information, a functional description (inputs and outputs, preconditions and effects), and non-functional properties such as categorisation and quality rating. The service model describes the service's operation and enables invocation, composition, and monitoring of a service. It describes whether the service is atomic or composed of other atomic services. The grounding specifies how the service is invoked technically by the service consumer including a network address of the service endpoint. It also provides information about data-types used in the operations of services. It should be noted that although OWL-S uses Web Service Description Language (WSDL) [15] as its grounding mechanism, it is not restricted to WSDL as the only service technology. The OWL-S ontology is very flexible to use and thus it serves as upper ontology for the IoT-adapted Service Model proposed in this paper.

### III. IOT INFORMATION MODEL

An IoT framework can benefit from structured models that detail various concepts and provide abstractions of the components and their attributes. This section defines the main abstractions and concepts that underlie the IoT domain and describes the relationships between them.

The main tenet of the IoT is extension of the Internet into the physical world, to involve interaction with a physical entity in the ambient environment. The entity constitutes 'things' in the Internet of Things and could be a human, animal, car, store or logistic chain item, electronic appliance or a closed or open environment. The 'entity' is the main focus of interactions by humans and/or software agents. This interaction is made possible by a hardware component, a

‘device’, which either attaches to an entity or is part of the environment of an entity so it can monitor it. The device allows the entity to be part of the digital world by mediating the interactions. The actual software component that provides information on the entity or enables controlling of the device, is a ‘resource’. As implementations of resources can be highly dependent on the underlying hardware of the device, a ‘service’ provides a well-defined and standardised interface, offering all necessary functionalities for interacting with entities and related processes. The services expose the functionality of a device by accessing its hosted resources. Other services may invoke such low-level services for providing higher-level functionalities, for instance executing an activity of a specified business process. The relations between services and entities are modeled as associations. These associations could be static, e.g. in case the device is embedded into the entity; they could also be dynamic, e.g., if a device from the environment is monitoring a mobile entity. These identified concepts of the IoT domain and the relations between them are depicted in Figure 1.

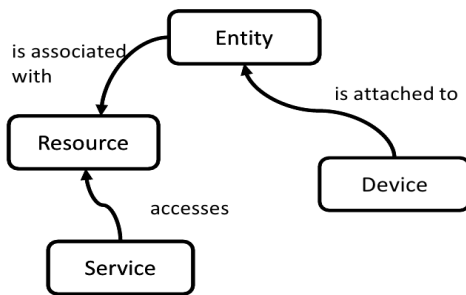


Fig. 1 IoT model: key concepts and interactions

The identified concepts need to be modeled in a format that provides interoperable and automated human and machine interpretable representations. The Semantic Web community has introduced formal definitions specified as ontologies that model different information in a domain, enable knowledge sharing and support automated reasoning. Specifically, the Web Ontology Language - Description Logic (OWL-DL), rooted in the decidable fragment of first-order logic, provides a powerful platform for a formal and machine-processible structure to represent the information that are collated from diverse sources.

Based on the identification above, of the main concepts in the IoT domain, this paper proposes a suite of ontologies that models entity, resources and IoT services. These will serve as a high-level model that references and builds upon existing vocabularies, as have been reviewed in section II. The concepts related to other relevant domains, such as sensors, observation and measurement and location, can be included from other ontologies. Where appropriate, properties are included to allow linking the proposed ontologies to external ontologies; for example, the global location URI of an entity could link to the relevant location instance in the GeoNames ontology (<http://www.geonames.org/ontology/documentation.html>),

where the given location is more fully described. This enables reusability of ontologies and fosters modularity.

A. Entity Model

An entity can have certain aspects that need to be taken into account. For example, when one needs to know about the location of an entity or the features of interest that data is available for. The OWL-DL representation has been used to define the entity model. The entity ontology is available at <http://purl.oclc.org/net/unis/EntityModel.owl>. A diagram of the main attributes in the entity model is shown in Figure 2.

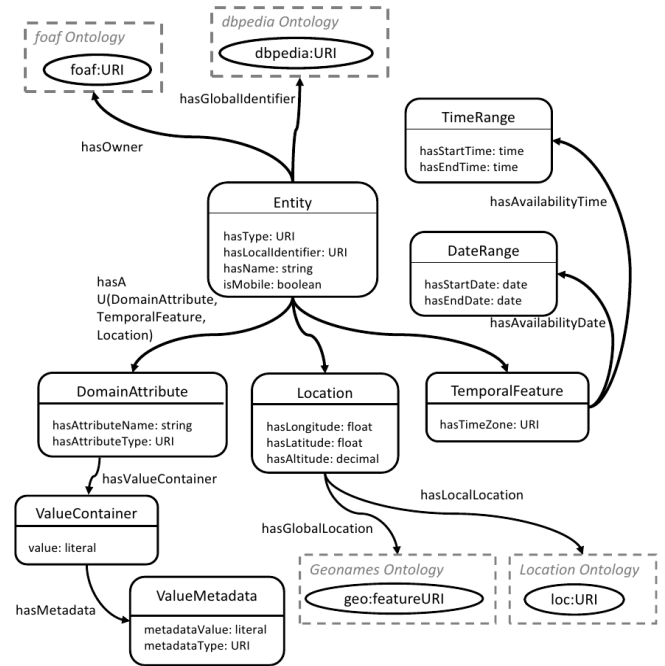


Fig. 2 The Entity model

An entity can have certain features, which include domain attributes, temporal features and location (Entity:hasA U(DomainAttribute, TemporalFeatures, Location)). Moreover, an entity instance can have multiple values for the domain, temporal or location feature. The observable features of an entity are specified by domain attributes that encapsulate the attribute name (hasAttributeName), attribute type (hasAttributeType) and one or more values in a value container (hasValueContainer). Each value container has the literal value specification (value), which is connected to metadata information. The metadata information can, for instance, be used to specify the units of measurement for the value, its timestamp or a notion of its quality. Temporal features are specified through time zone and through object properties to the time range (in terms of start and end time) and date range (start and end date) concepts. The location is defined in terms of the geographical coordinates (hasLatitude, hasLongitude, hasAltitude). The location concept also has properties that link to global (hasGlobalLocation) and local location (hasLocalLocation) ontologies. The local location ontology provides detailed location description, such as rooms and buildings on a campus, whereas the global location ontology URI links the

entity to existing high level location ontologies such as GeoNames, which provides toponyms or place names for cities, districts, countries and universities. Additionally, an entity has datatype properties that specify the URI of an owner (hasOwner) where the URI could point to a foaf ((<http://www.foaf-project.org/docs>) profile, a literal name (hasName) and a Boolean property to denote if the entity could be mobile (isMobile). An important attribute of an entity is the entity type (hasType). The local identifier (hasLocalIdentifier) property points to a local naming schema or literal representation of the entity and the global identifier (hasGlobalIdentifier) property is a placeholder to associate the entity to Linked Open Data (<http://linkeddata.org>) platform; for instance, to a DBpedia (<http://dbpedia.org/>) entry.

An illustrative example of an entity instance that implements the entity model is available at [http://purl.oclc.org/net/unis/U38\\_Entity.owl](http://purl.oclc.org/net/unis/U38_Entity.owl). The instance is that of a room with ID 'RoomU38'. The entity type (<http://www.owl-ontologies.com/LocationModel.owl#Room>) and localIdentifier (<http://www.owl-ontologies.com/LocationModel.owl#U38>) are mapped from a location ontology. The globalIdentifier links to the DBpedia entry for the institution of which the room is a part of, i.e. 'University of Surrey' in this case ([http://dbpedia.org/resource/University\\_of\\_Surrey](http://dbpedia.org/resource/University_of_Surrey)). The local location ([http://purl.oclc.org/net/unis/U38\\_Entity.owl#U38](http://purl.oclc.org/net/unis/U38_Entity.owl#U38)) is also specified from the location ontology and specifies the building location of the room. The globalLocation property links to the GeoNames feature URI of the town (<http://www.geonames.org/2647793/>). The room has an ambient temperature attribute, with attribute type 'Temperature' (<http://purl.oclc.org/NET/ssnx/qu/dim#Temperature>). The attribute value is '17' and the associated metadata specifies that the unit of measurement is degreeCelsius, in terms of the metadata type (<http://purl.oclc.org/NET/ssnx/qu/dim#Temperature-Unit>) and metadataValue (<http://www.qudt.org/qudt/owl/1.0.0/unit/Instances.html#DegreeCelsius>).

### B. Resource Model

A resource is the core software component that represents an entity in the digital world. Figure 3 details the resource description model. The resource model is available at <http://purl.oclc.org/net/unis/ResourceModel.owl>.

The resource concept has datatype properties that specify its name (hasName), an ID (hasResourceID) and a timezone defined in an external ontology (hasTimeZone). A resource also has a functional location property (hasResourceLocation) that links to the Location concept. This location could be the location of the device the resource runs on. The functional restriction denotes that a resource can only have a link to one location instance. The definition of the location concept is similar to the one defined in the entity model. The link to the resource type is denoted in terms of the type property (hasType) to the ResourceType concept. The resource type can be an instance of either of the following types: sensor, actuator, or tag. When the type is a sensor, the hasType property serves as a link to an instance of a sensor that conforms to an available sensor ontology

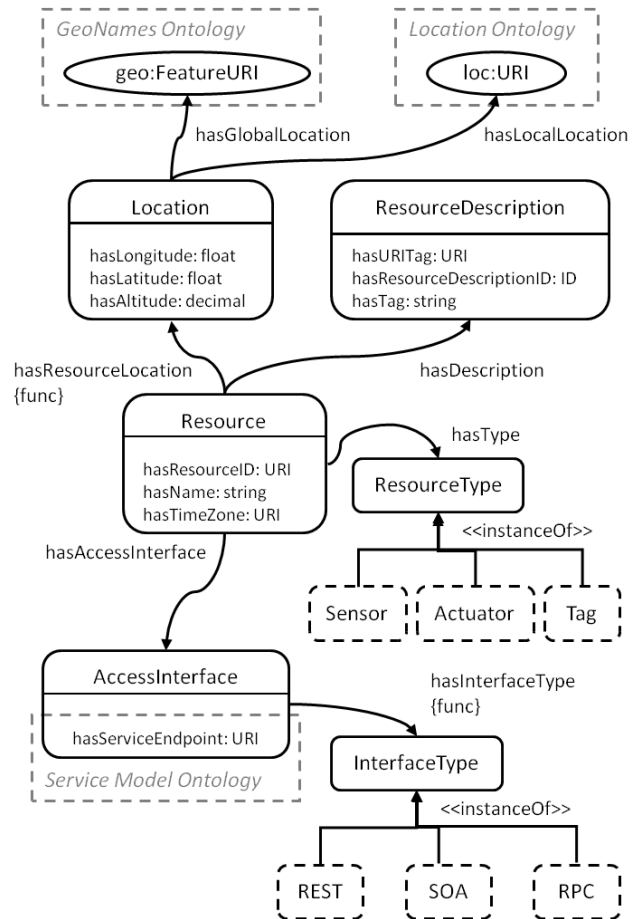


Fig. 3 The Resource model

(e.g. SSN sensor ontology). This allows linking the resource concept to external ontologies which define the related concepts without the need of repeating them in the proposed ontology suite. The interface to the resource (hasAccessInterface) is specified by the AccessInterface concept, which is further specified by an InterfaceType. The InterfaceType concept is defined as a set of instances which reflect technologies widely used in distributed systems, such as REST, SOAP, and RPC. The hasServiceEndpoint property links the resource model to the service model that exposes the resource functionalities to the IoT world.

Let 'U38\_Temp\_Sensor\_Resource' be an example resource which hosts the temperature sensing capabilities in the location 'BaBuildingLocation'. The location has geographic properties of longitude, latitude, and altitude as well as links to a local ontology modeling the buildings on University of Surrey campus and to the GeoNames entry for Guildford that localises the resource on a global scale. The sensor resource is further described by the 'ResourceDescription\_U38\_temp\_sensor' which contains a DBpedia classification of this resource and some tags describing the resource in plain text (temperature sensor in room 38 BA). The example resource is classified as 'Sensor' by the property hasType and it exposes the 'AccessInterface\_U38\_temp\_sensor' to IoT-users which is declared as a RESTful interface by 'hasInterfaceType'. The access interface of this resource contains the locator of the

service endpoint, which is part of the Service Model. The example resource presented here can be found at [http://purl.oclc.org/net/unis/U38\\_Temp\\_Sensor\\_Resource.owl](http://purl.oclc.org/net/unis/U38_Temp_Sensor_Resource.owl).

C. IoT Service Model

Resources are accessed by services which provide functionality to gather information about entities they are associated with or manipulate physical properties of their associated entities.

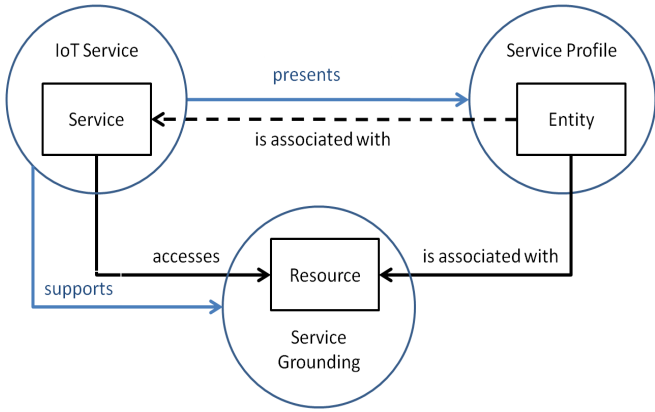


Fig. 4 The adapted OWL-S service ontology for IoT domain

The OWL-S specification has been designed as upper ontology for the Semantic Web Services. According to this specification, Semantic Web resources provide services which are described by their service profile, service model, and service grounding. Assuming potential IoT users are interested in information about the real world entities, they will search with terminology concerning entities of several domains. A search will return the service description containing a link to the resource offering the service that is able to satisfy user’s information request. Thus, a service profile must contain information about the entity it is associated to as well as the link to the resource that provides the service about the entity. We use the OWL-S profile’s object properties for this purpose. However, it must be noted that the association to an entity is not asserted (or may not be known at all) when the service is published; the link is asserted dynamically when an association is inferred. Mapping of OWL-S components to the identified IoT components (as demonstrated in Figure 1) is shown in Figure 4. The service profile describes services by their inputs, outputs, preconditions, and effects (IOPE). IoT sensing services provide output data service consumers are interested in (hasOutput). If a service needs any input to be processed by a resource it can be specified by a property (hasInput). Attributes of any entity can be used to describe the meaning of input and output parameters. Thus the IOPE properties of service profile link the Service Model to an Entity Model. Actuation services change properties of entities from an initial state to a desired state. The service profile’s initial states are specified as precondition (hasPrecondition) and desired states are determined as resulting condition (hasEffect). These two object properties have a logic expression, a predicate, as range denoting a condition about an entity attribute. Such conditions, like ‘equalTo’ can be evaluated to

true or false. A service will only be invoked if its precondition is evaluated to true.

We extend the existing profile with two more properties and their respective objects. ‘ObservationArea’ denotes the geographic area the service can observe (for sensors) or operate in (for actuators). With ‘ObservationSchedule’ it can be described when the service is able to operate and when it is planned to be out of work. The schedules can be used for maintenance, similar to SSN’s OperatingRange or can be utilized for saving energy on the resource providing the services.

The resource is accessed over the Internet through a suitable interface, such as using a Web Service. The service endpoint is identified by a locator (URI) in the resource’s AccessInterface. IoT users have access to this service endpoint the resource exposes, if not explicitly forbidden by privacy policies. The technical details that users need to know in order to access the service are specified in the service grounding. Since those details are dependent on the implementation of services and used technologies, they are not depicted in Figure 5. Typical information placed there are communication protocol, port number and the data types used for parameters that need to be sent to the service, as well as coming from the service, as depicted in Figure 5.

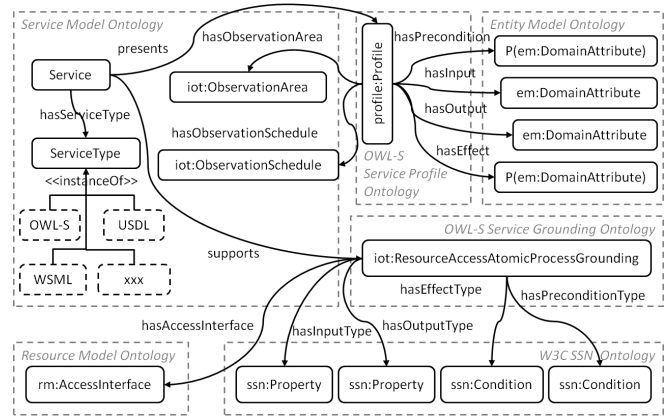


Fig. 5 Service Grounding

The ResourceAccessAtomicProcessGrounding specifies the mapping from domain specific entity attributes to properties observable by sensors. To each of the entity attributes assigned in the service profile an observation and measurement type can be assigned by their respective relations (hasInputType, hasOutputType, hasEffectType, and hasPreconditionType). The property hasInterfaceType determines the interface type as defined in the Resource Model. The IoT service model presented here is available at <http://purl.oclc.org/net/unis/OWL-IoT-S.owl>.

Let ‘U38\_Temp\_Sensor\_Resource’ be the example resource that exposes the ‘U38\_TempSensor\_Service’. This service has a type ‘OWL-S’ as specified by the hasServiceType property. The U38\_TempSensorService\_Profile presents the service profile and supports the U38\_TempSensorServiceProcessGrounding. The profile has links to U38\_ObservationArea as well as U38\_ObservationSchedule. The service output is described by the AmbientTempAttribute of the example entity ‘RoomU38’ which is defined



using the Entity Model proposed in this paper. The link to the temperature sensor resource is established through the service grounding. The service grounding is realized by `AccessInterface_U38_temp_sensor` which is part of the Resource Model for the example temperature sensor. The data type of the temperature measurement of this resource is determined by the range of property `hasOutputType` that is defined as a union of W3C SSN's 'Property' and a SENSEI Observation and Measurement type 'Temperature'.

The example service presented before is available at [http://purl.oclc.org/net/unis/U38\\_TempSensor\\_Service.owl](http://purl.oclc.org/net/unis/U38_TempSensor_Service.owl).

#### IV. USING THE INFORMATION MODELS

##### A. Dynamic Associations

In the presented information models, physical entities and services that provide information or allow the interaction with the entities, are not connected through fixed links that are directly part of the entity or resource models, but instead are linked through separately modeled associations.

Having separate associations provides a higher level of flexibility. Services may be associated with multiple entities at the same time, e.g., a temperature sensor may provide the indoor temperature of a room and at the same time the ambient temperature of all the people who are currently in the room. As can be presumed from the example, the set of people in the room is changing, thus the valid associations can also change dynamically. For a small resource-constraint device providing the actual service, it might be a significant burden if it has to handle the resulting changes. Instead dynamic associations can be handled in a server infrastructure like a cloud, where communication and computing resources are plentiful. An additional advantage is that privacy can be better protected as services associated to people should not be visible to everybody, information that may again be harder to protect on a resource-constraint device.

In order to support dynamic associations, the associations first need to be discovered and then their validity has to be monitored. For this purpose, relevant aspects of both the entity and the device, which hosts the resource through which the service is provided, have to be monitored.

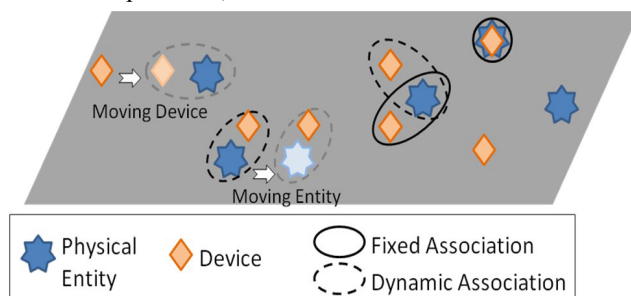


Fig. 6 Associations between physical entities and services provided through devices

Fig. 6 shows different associations between physical entities and services provided through devices. As both the physical entities and the devices can be mobile, the respective location or proximity of the entity and the device are relevant, but not necessarily sufficient indicators that a dynamic asso-

ciation is valid. Location information is explicitly modeled in both the Entity Model and the Resource Model, enabling both the specification of geographic coordinates as well as symbolic locations. Ownership or same movement patterns are examples for other relevant aspects that have to be taken into account for discovering dynamic associations.

An association also has to contain information about what aspect of the physical entity is being associated with the service. The `ResourceType` specifies what the service can do, e.g., provide information about the aspect in the case of a sensor, or change the aspect in case of an actuator.

##### B. Reasoning and Semantic Search

Utilizing the information represented in the form of the models and using them in IoT application and services also depends on finding relevant data and discovering entities, resources and/or services based on different scenarios. The semantic data can be represented in the form of Linked Data; i.e. links between entities, resources, service descriptions and also domain knowledge represented in the form of location ontologies, application data and resource in the Linked Open Data. In [16], we describe a Linked Data platform used for sensor descriptions that are represented and accessed in the form of linked data. Processing and reasoning large-scale semantic descriptions is also another important aspect to make the represented information more available to the end-users. In [17], we discuss a probabilistic machine learning mechanism to process semantic service descriptions for indexing and searching semantically described services. The introduced models provide similar type of descriptions so a similar method for indexing and searching the large-scale semantic data in the IoT domain can be adopted. Reasoning of resource, entity and service descriptions in relation to other data in the IoT domain and resources that describe application domain and environment attributes also enables to analyze the descriptions and supports autonomous communication and decision making processes. In [18], we have discussed some scenarios and concepts that utilize the sensor data and resource descriptions in the IoT domain.

#### V. DISCUSSION

This paper focuses on describing the IoT component and data description models and captures relations between different data provider and data descriptor components in the IoT field. Our main objective is describing the entity, resource and service models for the IoT domain. We have also described how these models can be related to each other and can be also associated with the domain knowledge. The main advantage of introducing semantic models for the IoT component descriptions is providing interoperability in data and service levels. The models do not limit the data and/or service providers in what they can provide or provision; they, however, enable data/service providers to provision machine-interpretable data and descriptions such as what is provided, what the data/service is related to, where is the location of a data or a service provider, who is the provider. The models in general enable to describe spatial, temporal and thematic data related to data which is in line with the aspects that are also defined for the Semantic Sensor Web [19].

The semantic modeling and OWL/RDF descriptions solve the interoperability issues within the stakeholders that have agreed and/or provide data using the models. We have aligned our descriptions with the key players and existing standards and representation models in this domain. For other types of existing and future description models, it will be still possible to provide an alignment to map the descriptions across different IoT resource description frameworks. This however depends on the features that are described in different models and it would be applicable as long as the required and provided data can match to the designated attributes and assumptions that we have made in designing the models.

Timeliness of data and reasoning services is also another issue that needs to be considered while using semantic modeling and annotated data in the IoT domain. In large-scale deployments, identifying the relevant resources that can provide required data/services and reasoning with the domain knowledge can be a time consuming process. Effective utilization of these models depends on how efficiently the discovery and reasoning processes can perform as the number of components and the volume of descriptions increases.

Power and resource constraints and limited capabilities of the underlying devices is also another issue that should be considered when semantic data modeling is used in the IoT domain. In the introduced framework, we assume that the models are used to describe resources, entities and services and the semantic data is stored and utilized on powerful machines, e.g. gateway nodes or middleware components. This enables the devices to perform independently while the descriptions make their capabilities, descriptions and data more processible and interpretable for software agents and human users. The observation and measurement data can be also discussed in the middleware level and/or on the sensor node level within the capillary networks and then different techniques can be used to support effective communication of this data over lower power and low bandwidth devices and networks.

Manual versus automated annotations and associations processes is also another important issue in dealing with the detailed semantic models. The important question is that who will provide this semantic annotation and how this data for each component will be associated to other data and resources in the IoT domain and also to the existing data on the cyber world (i.e. the Web data). In [20], we discussed a middleware solution that uses predefined template models to provide semantic annotation for known types of sensors. A similar approach can be adapted for known types of resources, entities and services in the IoT domain. Association of the resources can be also supported by off-line reasoning processes that analyze the annotations and find the relation between different entities, resources and services based on different aspects such as location, type, and domain attributes.

## VI. CONCLUSION AND FUTURE WORK

The models proposed in this paper are designed based on our previous work and experiences in the SENSEI project

and SSN ontology modeling and can support a general association between different components in the IoT domain. The models provide a semantic annotation framework so the legacy data can be also enhanced using these descriptions. The semantic annotation allows that the model data is represented as linked data and can be associated with the existing data on the Web and in particular Linked Open Data.

Future work will involve development of a resolution framework that allows searching the large scale data of the instances of the models in the IoT domain and will facilitate automated inference of dynamic associations.

## REFERENCES

- [1] "SENSEI - Integrating the Physical with the Digital World of the Network of the Future," IST-FP7, 2010. <http://www.sensei-project.eu/>
- [2] C. Villalonga, Ed., D2.5 Adaptive and Scalable Context Composition and Processing, *Public SENSEI Deliverable*, 2010.
- [3] D. J. Russomanno, C. Kothari, and O. Thomas, "Sensor ontologies: from shallow to deep models," in *Proceedings of the Thirty-Seventh southeastern Symposium on System Theory*, 2005.
- [4] J. H. Kim, K. Kwon, K. D.-H., and S. J. Lee, "Building a service-oriented ontology for wireless sensor networks," in *Proceedings of the Seventh IEEE/ACIS International Conference on Computer and Information Science*, 2008, pp. 649-654.
- [5] OGC, "OpenGIS® Sensor Model Language (SensorML) Implementation Specification," *Open Geospatial Consortium, Inc.* 2007. <http://www.opengeospatial.org/standards/sensorml>.
- [6] P. M. Barnaghi, S. Meissner, M. Presser, and K. Moessner, "Sense and sens'ability: Semantic data modelling for sensor networks," in *Proceedings of the ICT Mobile Summit*, 2009.
- [7] OGC, "Open Geospatial Consortium (OGC) Sensor Web Enablement: Overview and High Level Architecture," *OGC white paper*, 2007.
- [8] "W3C SSN Incubator Group Report". 2011. [http://www.w3.org/2005/Incubator/ssn/wiki/Incubator\\_Report](http://www.w3.org/2005/Incubator/ssn/wiki/Incubator_Report)
- [9] H. Neuhaus and M. Compton, "The Semantic Sensor Network Ontology: A Generic Language to Describe Sensor Assets," presented at the *Pre-Conference Workshop on Challenges in Geospatial Data Harmonisation (AGILE 2009)*, Hannover, Germany, 2009.
- [10] C. Henson, J. K. Pschorr, A. P. Sheth, and K. Thirunaran, "SemSOS: Semantic Sensor Observation Service," in *Proc. of the 2009 International Symposium on Collaborative Technologies and Systems (CTS 2009)*, Baltimore, MD, 2009.
- [11] S. Bowers, J. S. Madin, and M. P. Schildhauer, "A Conceptual Modeling Framework for Expressing Observational Data Semantics," Q. Li et al. (Eds.): ER 2008, LNCS 5231, 2008, pp. 41-54.
- [12] R. Garcia-Castro, C. Hill, and O. Corcho, "SemserGrid4Env Deliverable D4.3 v2 Sensor network ontology suite," Feb 28 2011.
- [13] G. Percivall, "ISO 19119 and OGC Service Architecture," in *Proceedings of FIG XXII International Congress*, 2002.
- [14] "OWL-S: Semantic Markup for Web Services", *W3C Member Submission* 2004. <http://www.w3.org/Submission/OWL-S>.
- [15] "Web Services Description Language (WSDL), 1.1", *W3C Note*, 2001. <http://www.w3.org/TR/wsdl>
- [16] P. Barnaghi, M. Presser, and K. Moessner, "Publishing Linked Sensor Data," presented at the *Proc. 3rd International Workshop on Semantic Sensor Networks (SSN), in conjunction with the 9th International Semantic Web Conference (ISWC 2010)*, 2010.
- [17] G. Cassar, P. Barnaghi, and K. Moessner, "Probabilistic Methods for Service Clustering," presented at the *Proceedings of the 4th International Workshop on Semantic Web Service Matchmaking and Resource Retrieval*, 2010.
- [18] W. Wang and P. Barnaghi, "Semantic annotation and reasoning for sensor data," presented at the *Proceedings of the 4th European conference on Smart sensing and context (EuroSSC2009)*, Guildford, UK: Springer-Verlag, 2009.
- [19] A. P. Sheth, C. Henson, and S. S. Sahoo, "Semantic sensor web," *IEEE Internet Computing*, vol. 12, pp. 78-83, 2008.
- [20] F. Ganz, P. Barnaghi, F. Carrez, and K. Moessner, "Context aware management for sensor networks," in *Proceedings of the Fifth International Conference on COMMunication System softWARE and middleWARE (COMSWARE11)*, 2011.





# FastFIX: An Approach to Self-Healing

Benoit Gaudin and Mike Hinchey

Lero—The Irish Software Engineering Research Centre, University of Limerick, Ireland  
firstName.lastName@lero.ie

**Abstract**—The EU FP7 FastFIX project tackles issues related to remote software maintenance. In order to achieve this, the project considers approaches relying on context elicitation, event correlation, fault-replication and self-healing. Self-healing helps systems return to a normal state after the occurrence of a fault or vulnerability exploitation has been detected. The problem is intuitively appealing as a way to automate the different maintenance type processes (corrective, adaptive and perfective) and forms an interesting area of research that has inspired many research initiatives. In this paper, we propose a framework for automating corrective maintenance and present its early stage development, based on software control principles. Our approach automates the engineering of self-healing systems as it does not require the system to be designed in a specific way. Instead it can be applied to legacy systems and automatically equips them with observation and control points. Moreover, the proposed approach relies on a sound control theory developed for Discrete Event Systems. Finally, this paper contributes to the field by introducing challenges for effective application of this approach to relevant industrial systems.

## I. INTRODUCTION

SOFTWARE maintenance aims to modify a software system after it is deployed in production ([1], [2]). In [3], the authors identify three different types of maintenance: adaptive, perfective and corrective. Adaptive maintenance is performed to make the computer program usable in a changed environment. Perfective maintenance mainly tackles performance and maintainability issues. Corrective maintenance is performed to correct faults. Over the last 20 years the complexity of both software and communication infrastructures has increased at an unparalleled rate. This level of complexity means that software systems are more prone to unexplained failures, require more support and maintenance, and cost more to deploy and manage. A fundamental challenge faced by the software industry is how to ensure that these hugely complex software systems require less maintenance and human intervention. With concepts such as self-healing, autonomic and self-adaptive systems provide an answer by reducing human intervention and reducing the apparent complexity of systems.

Several surveys on self-healing have been published to describe the State-of-the-art of this field (e.g. [4], [5], [6]). According to these surveys, the major trends towards finding a solution to the self-healing problem rely on redundancy that may concern both architecture and code resources. These

The research leading to these results has received funding from the European Community's Seventh Framework Programme managed by REA—Research Executive Agency <http://ec.europa.eu/research/rea> ([FP7/2007-2013] [FP7/2007-2011]) under grant agreement n° [258109]. This work was also supported, in part, by Science Foundation Ireland grant 03/CE2/1303\_1 to Lero - the Irish Software Engineering Research Centre ([www.lero.ie](http://www.lero.ie)).

approaches somehow assume that systems are designed with adaptive capabilities and are therefore better suited to address adaptive and perfective maintenance. In this article, we focus on self-healing for corrective maintenance.

We propose a control theoretic approach to self-healing in order to deal with corrective maintenance. Control makes it possible to drive the system in a range of desired behaviors. It represents an interesting approach to avoiding behaviors that lead to failures. This is achieved by dynamically disabling some of the implemented features. Moreover, the proposed approach automatically synthesizes supervisors charged with controlling the software. Hence, this automates the computation of a new suitable range of software behaviors whenever corrective maintenance needs to be performed, e.g., a failure has been reported and behaviors exhibiting this failure need to be removed or avoided.

Section II introduces the FastFIX project ([7]) goals and the different research aspects that are investigated: context elicitation, event correlation, fault replication and self-healing. Section III presents the early stage development and approach considered for self-healing, which is based on control theory.

Finally, challenges to be tackled in order to implement effective and efficient control theoretic self-healing features are discussed in Section IV. Most of these challenges relate to supervisory control theory and its applicability to software systems.

## II. FASTFIX: MONITORING CONTROL FOR REMOTE SOFTWARE MAINTENANCE

The FastFIX project aims to provide methods and a platform for improved remote maintenance of software applications. The FastFIX platform monitors the execution of applications, their environment and user behaviors. It also provides techniques that analyze the collected data in order to identify symptoms of execution errors, performance degradation, or changes in user behavior.

This platform comprises both a client part which interacts locally with the target application and a server part which receives data from the client in order to perform analyses.

Collecting information from the target application, as well as its environment and users, is the basis for the FastFIX analyses. Therefore, context elicitation and user modeling play a crucial role in the project. These challenges are tackled through lightweight software sensor deployment into the runtime environment, together with facilities to interpret the user behaviors from data representing their interaction with the application.

These sensors can also provide information about the execution of the application itself and its environment, i.e., method calls, variable values, timestamps, etc. This data can then be analyzed by an event correlation component in order to detect anomalies representing possible attacks or application malfunctions. Rule-based systems are often used in order to perform event correlation. However these systems face issues whenever the complexity of the system to be monitored and the amount of possible correlations is large. More specifically, managing a large amount of rules in order to ensure consistency and proper priorities as well as to avoid redundancy, is a very challenging task. The FastFIX project will tackle this issue and investigate rule-based correlation engines that are easier to define and maintain.

The data collected by the FastFIX platform is also used in order to replicate faults whenever they occur. Indeed, fault replication represents an interesting feature in order to diagnose issues. It is first an appealing approach as it avoids manual fault replication from the symptoms reported by the user, which can be incomplete, inaccurate or even irrelevant to the error. Moreover, fault replication techniques address the replication of faults related to concurrency. This is of high interest as these types of faults are usually difficult to reproduce, and hence to diagnose and fix.

The information about the target application collected at runtime is used in order to perform self-healing. When a failure occurs, the self-healing capabilities make it possible to automatically modify the application behaviors so that this failure cannot occur in future executions. Performing such automation is a challenging task and better suits types of faults for which no new behavior creation is required. For this reason, the FastFIX self-healing mechanism is flexible and also allow for humans-in-the-loop in order to tackle those software fixes that cannot be automated.

Finally, as the collected data is sent to the FastFIX server for analyses, the system execution and user information must be sent from the client machine to the maintenance team. As this information may contain sensitive personal data, it is important to ensure user confidentiality. This is tackled in FastFIX using obfuscation techniques (e.g. [8]). Obfuscation techniques aim to abstract the actual variable values into *restricted domains* so that it is not possible anymore to precisely determine what values were used by the user. However, the domain in which the value is abstracted is accurate enough in order to replicate the application execution in a similar way as the one performed with actual user data.

Figure 1 illustrates how the different aspects tackled by FastFIX can be combined and act in a complementary manner in order to achieve effective and efficient remote software maintenance. On the client side of the diagram are the user, the FastFIX target application and the FastFIX client itself. This client monitors the user interactions with the target application as well as the application execution itself. From this information, it is able to perform analyses and identify application failures or changes in the user behaviors. It can then provide feedback and recommendations to the user themselves, or send

the collected information to the FastFIX server for further processing. This data can be preprocessed in order to be obfuscated so that no sensitive user information is sent to the server.

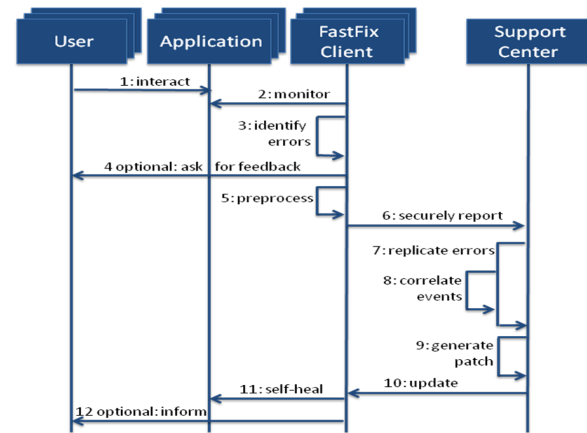


Fig. 1. The FastFIX overview.

The information sent to the FastFIX server can then be used in order to perform event correlation, fault replication and automatic patch generation, through self-healing. The outcomes of these analyses are then reported back to the FastFIX client so that actions are taken on the target application, e.g. applying patches or providing user recommendations.

The rest of this paper focuses on the FastFIX approach to self-healing, which relies on Supervisory Control Theory for discrete event systems.

### III. A CONTROL THEORETIC APPROACH TO SOFTWARE SELF-HEALING

In computing systems, control theory has traditionally been applied to data networks, operating systems, middleware, multimedia and power management ([9]). This section proposes a control-based approach for the self-healing of software systems.

With this approach, systems can be automatically equipped with autonomic features and therefore follow the autonomic feedback loop of Figure 2(a) at runtime. In particular, sensors and actuators are automatically added to the software system in order to realize the *Data Collection* and *Action* phases of Figure 2(a). The *Analysis* phase is related to control theory. The system sensors and actuators also implement the feedback control loop presented in Figure 2(b) and two types of analyses can therefore be achieved: runtime control decision and automatic supervisor synthesis. Runtime control decision can ensure the avoidance of known undesired behaviors during execution.

Supervisor synthesis is used to automatically modify the system behaviors. It represents the core technique for corrective maintenance in our approach. The overall proposed approach is detailed in the remainder of this section.

Our self-healing approach consists of two different phases: a pre-deployment phase which is performed before the system

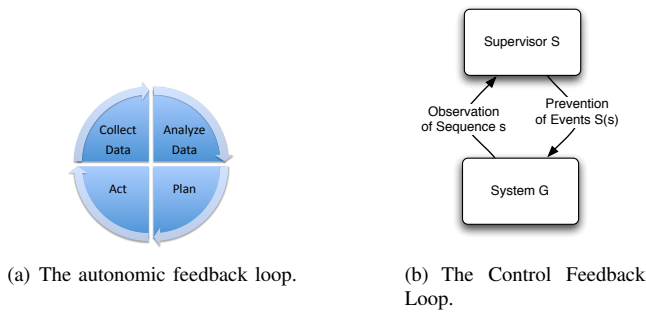


Fig. 2. The runtime autonomic and control feedback loops.

is deployed and where self-healing features are added to the software; and a post-deployment phase corresponding to the automatic or semi-automatic execution of the maintenance process where the system self-healing features are employed.

The latter part itself consists of the system control at runtime as well as supervisor synthesis whenever new runtime system specifications need to be ensured, e.g., when a fault has occurred. Overall the presented approach can be seen as a three phase approach: pre-deployment, control and synthesis. The pre-deployment phase prepares the system for control and synthesis. However, the concepts related to the pre-deployment phase depend on the ones related to the control and synthesis phases. We will therefore discuss the pre-deployment phase last.

#### A. Control Phase

We first consider the control phase which follows the principle illustrated in Figure 2(b). In this diagram, the supervisor observes and controls the current behaviors of the system. These behaviors are represented as sequences of events.

As a basic case, we consider that the events that can be observed by the supervisor consist of method calls. This can be further augmented for instance with other program statements such as conditions and also values passed to method parameters. Therefore we consider that the behavior of a software application is described by the sequence of method calls that occur at runtime<sup>1</sup>. Each time a method is executed, the supervisor is aware of it and can update its knowledge regarding the current behavior of the system.

Implementing Figure 2(b) requires the addition of observation (sensors) and control (actuators) points. In order to achieve this, we consider embedding some code that models the supervisor into the software application. More specifically, the model of the supervisor can be considered as an object whose current state can be updated whenever a method of the application to be controlled is called. Moreover, control can be performed by preventing method executions as modeled in the supervisor. Figure 3 illustrates this idea where a *Supervisor* type is added to the software application. This type (or class) also provides a method *accepts* which given the name of a

method returns true if and only if the supervisor will allow that this method is called from the current state. Whenever a method is allowed, its body is executed and the current state of the supervisor object is updated.

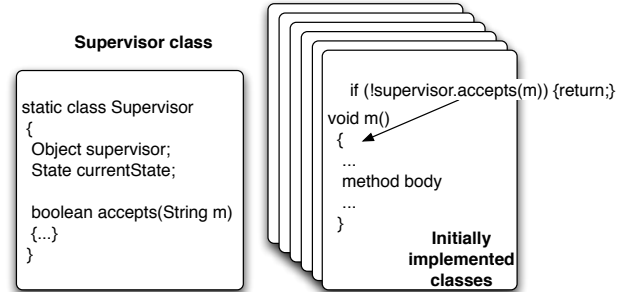


Fig. 3. A possible code instrumentation offering observation and control points.

If the method is not allowed by the supervisor, then it must not be executed. A basic approach to achieving this is described in [10], [11] and consists of returning abruptly as indicated in Figure 3. Such an approach allows for dynamic restriction of the system executions, e.g., a method execution may be prevented after a given sequence and allowed after another one.

#### B. Synthesis Phase

Paragraph III-A describes the principle and possible mechanisms for controlling software application behaviors at runtime by means of a supervisor. The design of such a supervisor corresponds to determining how the application behaviors must be modified in order to avoid undesired behaviors. However designing such a supervisor is a challenging task and prone to error. Moreover, the high complexity of software applications makes it difficult to manually take into account all the possible failures that can occur and need to be prevented. For this reason, supervisors may need to adapt at runtime so that they take into account newly observed undesired behaviors, hence performing corrective maintenance. Such an approach is further described in Figure 4(a) and considers automatic synthesis of such supervisors. More specifically, we consider techniques that automatically compute the model of a supervisor given a model of the application behavior and a model representing a set of desired behaviors<sup>2</sup>. Supervisory Control Theory (SCT) on Discrete Event Systems introduced by Ramadge and Wonham ([12]) offers such a framework and techniques for the automatic synthesis of supervisors.

SCT is a formal theory that aims to automatically design a model for a supervisor ensuring some safety property. Supervisory Control Theory defines notions and techniques that allow for the existence and automatic computation of a model of the supervisor, given a model of the system as well as the property

<sup>1</sup>For simplicity, we discuss the approach with this basic setting. The general case is discussed in Section IV.

<sup>2</sup>Behaviors that do not belong to this set are undesired.

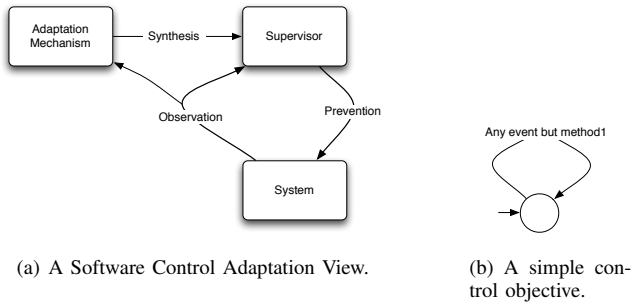


Fig. 4. Software control Adaptation view and a simple control objective.

to be ensured. In this theory, models of a system  $G$  are represented by languages over alphabets of events, denoted  $L(G)$ . These languages correspond to sets of sequences of events, each representing a possible behavior/execution of the system.

Although not as expressive as languages, Finite State Machines (FSM) are used to model the possible behaviors of the system as well as the supervisor and the properties to be ensured by control. Regarding the modeling of supervisors, Figure 2(b) shows that they can be seen as a function that takes a given sequence  $s$  and returns to the system a set of allowed events after  $s$ . The function  $S$  representing the supervisor can be encoded by a FSM  $G_S$  such that for all  $s \in L(S)$ ,  $S(s)$  represents the set of events that can be triggered from the state reached in  $G_S$  after sequence  $s$ . It is worthwhile noting that in the case where events only represent method calls, program variables are not taken into account in the application behavior model. The state space of the corresponding FSM does not therefore correspond to the one of the program variables. Instead the model encodes loops and branching points in the program, limiting the state space explosion issue related to large systems (more details are provided in Section III-C and more particularly in Figure 5).

Supervisors ensure a given property, called the *control objective*. Such a property is modeled as a FSM as well, generating a set of “safe” behaviors and meaning that the behaviors that are not encoded by this FSM are undesired. For instance, Figure 4(b) represents a very simple control objective which models that `method1` must never be executed.

The main goal of Supervisory Control Theory is to automatically synthesize a model of a supervisor that ensures that the system behaviors are all included in the ones described by the control objective. The theory also considers that not every event can or should be disabled by a supervisor. Such events are said to be uncontrollable. In order to take such events into account, the alphabet of the system is assumed to be composed of a set of *controllable* events ( $A_c \subseteq A$ ) and *uncontrollable* events ( $A_u \subseteq A$ ). Each event of the system is either controllable or uncontrollable. Controlling a system consists of restricting its possible behaviors taking into account the controllable nature of the system events. In order to achieve this, Ramadge and Wonham (see for example [13]) introduce a property called *Controllability*. A system  $G'$  whose behaviors

correspond to a subset of those of  $G$  is controllable w.r.t  $A_u$  and  $G$  if  $L(G').A_u \cap L(G) \subseteq L(G')$ . A controllable set of behaviors  $G'$  ensures that no sequence of uncontrollable events can complete a sequence of  $G'$  into a sequence of  $G$  that is no longer in  $G'$ . In other words, the controllability condition ensures the synthesized supervisor can be effectively implemented with respect to the available controllable events. We now define the basic supervisory control problem, which can be stated as in the following.

**Basic Supervisory Control Problem (BSCP):** Given a system  $G$  and a control objective  $K$ , compute the maximal controllable set of behaviors included in those of both  $G$  and  $K$ .

Ramadge and Wonham (see for example [13]) have shown that a solution to the BSCP exists if and only if the maximal controllable set of behaviors included in those of both  $G$  and  $K$  is not empty. They also provide an algorithm computing this FSM which encodes a most permissive supervisor ensuring the control objective (see for example [13]). This algorithm can be seen as a function that takes as inputs a set of uncontrollable events  $A_u$ , a FSM representing the control objective  $K$  and a FSM representing the behaviors of the system  $G$ . In our proposed approach, corrective maintenance is applied by modifying the application behaviors. Determining the set of behaviors to be ensured by control is performed through solving the BSCP. The obtained model is then used to control the application. Part of the mechanism involved in achieving this is described in Section III-A and part of it is performed during the pre-deployment phase and is described in Section III-C.

### C. Pre-Deployment Phase

The pre-deployment phase aims to prepare the software application before deployment so that control and synthesis can be performed at runtime. This preparation is not application specific and the same processing is applied to any software systems under consideration. It consists of two subtasks: code instrumentation and model extraction. Each of these tasks is performed in an automated fashion.

Code instrumentation is performed in order to introduce observation and control points as well as to embed a supervisor in the application. These features will allow for software control such as depicted in Figure 2(b). Intuitively, automatically instrumenting source code for this purpose consists of automatically augmenting the application source code with statements such as in Figure 3, i.e., embedding a supervisor into the system as well as adding conditional statements in each method body so that method calls can be observed and method body execution can be controlled at runtime.

Moreover, model extraction from source code is performed in order to obtain a model of the behaviors of the system. As mentioned in Paragraph III-A, application behaviors are represented with sequences of method calls. An over-approximation can be obtained from the source code by considering methods, branching and loops as illustrated in Figure 5.

### D. Overall Approach

The proposed overall approach is depicted in the diagram of Figure 6. The left hand side of this diagram represents the

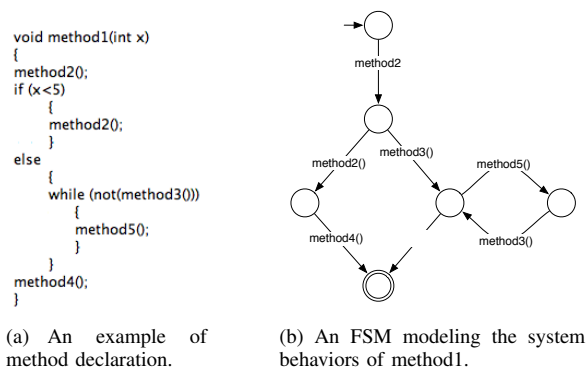


Fig. 5. Illustration of FSM extraction.

pre-deployment phase at which code is instrumented in order to introduce observation and control points as well as data structures that make it possible to represent and manipulate supervisor models. A binary (or Bytecode) application with these facilities can then be obtained through compilation. During the pre-deployment phase, a model of the behaviors is also automatically extracted from the source code by analyzing control flows and method calls in the application.

During the runtime and maintenance phase, the software artifacts (source code and binary code or Bytecode) are modified no more. Only models of a supervisor representing their possible runtime behaviors are manipulated in order to control the application so that only desired behaviors can be executed. If no control is necessary at first, then the extracted model of the application can be used as a supervisor model. This will certainly have no effect on the implemented application behaviors.

Some unknown possible failures of the system may occur at runtime, requiring the application to be healed. The observation of such a failure indeed indicates that the system behavior is not satisfactory and needs to be corrected.

In this approach, this correction is performed by modifying the supervisor that interacts with the application at runtime. Using Supervisory Control Theory as introduced in Paragraph III-B, this can be automatically achieved when a control objective is provided (in this approach, a model of the possible behaviors of the software application was computed in the pre-deployment phase and is therefore assumed to be available). In some situations, this control objective can be automatically derived from observations of failures during the system execution (see for example [10]). In general, control objectives can also be provided by expertise. The accuracy and relevance of the expertise involved in designing a control objective will have an impact on the accuracy and relevance of the corrective solution applied to the system. For instance, diagnosis can help design a more accurate control objective. However, in cases where deep analyses and diagnostics cannot be conducted (e.g., when the amount of time that is necessary to perform this task is too long), then a simple control objective excluding the  $u$  previously-observed undesired sequences of

method calls can be submitted to the supervisor synthesis algorithm. Of course this latter option may correspond to a coarse control of the application, unnecessarily removing proper (acceptable) behaviors.

The control objective of Figure 4(b) illustrates the case where it is desired to prevent occurrences of method1. Although in some situations such an objective represents the most relevant property to ensure in the system, it may also represent an approximation due to lack of knowledge. The root cause of the failure that leads to the design of this control objective may not indeed come from method1 but from other methods calling method1. If the developers can only observe that the failure occurs when method1 is executed, then preventing the occurrence of method1 appears to be the most straightforward way to avoid the failure.

In any case, the algorithm solving the BSCP provides a new model of a supervisor which will be used by the application in order to prevent the future occurrence of undesired behaviors. In general, a restart of the application is necessary in order to take into account the newly computed supervisor model.

The control theoretic approach for self-healing proposed in this section raises several challenges. Some of these challenges correspond for instance to automating the introduction of autonomic features into legacy applications; automatically extracting relevant and accurate models from source code; applying supervisory control theory on large systems; designing accurate control objectives, etc. They also relate to different fields of computer science such as software engineering (e.g., software modeling, logging, maintenance), formal methods and control theory. Some of these challenges are detailed in Section IV.

#### IV. CHALLENGES

The control theoretic self-healing approach introduced in previous sections poses several challenges. Most of these challenges are directly or indirectly related to performance and complexity. These issues are related to the system size, the system model size, the efficiency of the analyses and supervisor synthesis as well as the need for a low overhead during runtime execution.

The approach in Section III is flexible enough to allow for complexity reduction by considering only sub-parts of the system to be observed, controlled and modeled and also by approximating the system and control objective models through abstractions. However, reducing the amount of information available to the framework described in Figure 6 alters the quality of the supervisors that can be automatically synthesized and therefore the relevance of the self-healing solution to be applied. Therefore, trade-offs between scalability and relevance of the approach have to be determined. For this purpose, challenges related to system observability and controllability, to system modeling, to designing control objectives, to concurrency and to correction types to be applied, are discussed in the rest of this section.



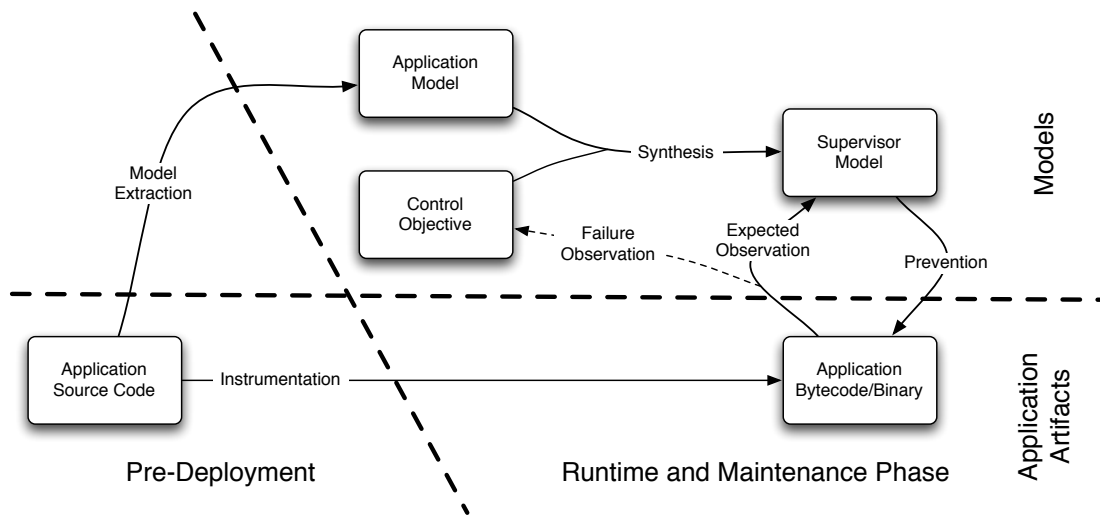


Fig. 6. A Software Control Approach to Self-Healing.

#### A. Controllability and Observability.

Section III introduces a control theoretic approach for the automation of corrective maintenance processes. This approach models the possible executions of the system as sequences of method calls. The approach relies on a supervisor that observes some of the method calls at runtime (observability). Moreover the control mechanism disables the occurrence of some method calls in order to ensure some properties of the application (controllability).

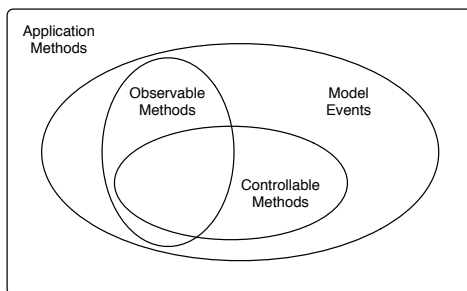


Fig. 7. Observable and Controllable Software Methods.

Figure 7 represents the relationship between observable and controllable methods as well as how they relate to the sets of application methods and the methods that are part of the application model (representing model events). First the model events correspond to a subset of the application methods. Second, observable and controllable methods represent a subset of the model events. The observable methods are those whose call can be observed by the supervisor at runtime. The controllable methods represent those whose execution can be prevented by the supervisor.

The amount of model events and observable events has an impact on the size of the application model. Moreover,

the complexity of the supervisor synthesis as well as the runtime overhead increase with this model size. On the other hand, the relevance of the supervisor synthesis increases with the amount of information present in the application model. Therefore, as our approach aims to systematically automate the introduction of method observability, one important challenge is to find approaches to determine the best trade-off between the amount of methods to be made observable and the amount of information required for relevant synthesis.

Moreover, defining controllable methods is also a challenging task. Although it is technically feasible to prevent any software method to be executed at runtime, this is however unreasonable for some methods as it may have an impact on the application's integrity. For instance, preventing the execution of a method that returns an object, which will be later processed during the program execution, may introduce new possible failures in the system. Therefore, procedures (i.e., methods that do not return any object) are better suited to be controllable. This however does not fully guarantee the application's integrity as procedures may modify global variables. Techniques such as slicing (e.g., [14]) could be considered in order to determine methods whose execution has a very limited impact on the global execution of the application. Another type of method that possesses such a property are methods capturing events from graphical interfaces and executing code in reaction to button clicks, for instance. These methods represent good candidates for controllable methods as they are not called from other parts of the application or third party applications.

#### B. Finite State Machines and Variables.

In Sections III and IV-A, we represent application models as Finite State Machines, where the transitions represent method calls. Although this view of the system behaviors makes it possible to take into account the past execution in order to



decide on the control actions to be taken, it does not explicitly take into account the system variables. This approach has an interesting upside: the state space of the model is in general smaller than the state space of the application. Indeed, with this approach, the states of the model do not encode a possible tuple of values of the application variables. Instead states only encode branchings and loops of the program (as illustrated in Figure 5).

The downside of this approach is that information on the system behaviors is not as accurate as if variable values were taken into account. For instance, disabling the occurrence of a method call by control may depend on the values of the parameters with which the method is called (if any). Therefore, taking into account some of the application variables in the approach while preserving its scalability is a challenging task.

Several works have considered supervisory control on FSM with variables (e.g., [15], [16], [17], [18]). Although Extended Finite State Machines offer a compact way of representing potentially large, or even infinite, system state spaces, the supervisor synthesis takes into consideration the system state space itself. In order to tackle this issue, abstractions of the variable values rather than the possible values themselves should be considered for analysis. This can be done in the same spirit as for Abstract Interpretation ([19]) or data obfuscation techniques (e.g., [8]).

### C. Automatic Extraction of Application Models.

The approach introduced in Section III relies on the automatic design of a model of the application behaviors. In its basic form, this model can be a Finite State Machine whose transitions represent method calls. As explained in Section IV-B, such a model can be extended in order to take system variables into account. Extended FSM can then be considered as a way to model the application behaviors.

Some tools have been implemented in order to extract and analyze models represented as EFSM. For instance, PROMELA allows for program modeling with FSMs. PROMELA models can be used as input to the SPIN tool, which can then model-check this model against some properties. Bandera ([20]) is a tool that allows for FSM extraction from Java code. Bandera offers the possibility of exporting the extracted models into the PROMELA format. More recently in [21], the authors proposed an efficient approach for model extraction from programs. The approach makes it possible to deal with different but syntactically similar programming languages such as C++ and Java.

In all these approaches, however, only some particular parts of the programming language are considered. The approach described in Section III requires that the model obtained of the application is complete; i.e., any observable program execution should be encoded in the model. This characteristic is related to the fact that the extracted model is used for on-line monitoring and must take into account all the possible system behaviors. Therefore an important challenge for model extraction consists of obtaining a complete application model. This requires that the model complies with the specification of

the language compiler or virtual machine so that features such as threads and graphical components are treated appropriately. This aspect is also related to the notion of concurrent behaviors and is detailed further in Section IV-E.

### D. Improving Application Models from Runtime Observations.

As mentioned in Section IV-C, the model of the application behaviors should be complete in order to be used to monitor the application at runtime. However although some variables of the system are taken into account in the model, some others may still be abstracted, leading to a model that is an over-approximation of the possible application behaviors.

However, some information regarding both the path of the model taken as well as the variable values observed during the system execution represent valuable information regarding actual possible behaviors. For instance, if a transition of the FSM modeling the system results from over-approximating the actual system behaviors, it will not be triggered. Therefore if after a large number of executions it is observed that some transitions have never been triggered, one may conclude that these transitions are over-approximating the system behaviors and should not be taken into account for analysis. This leads to an improvement of the application model from the observations made at runtime.

This is an important point as although model completeness is important in order to ensure adequate monitoring of the system at runtime, it may unfortunately also induce some over-restrictive control. In other words, software control may be over-restrictive when taking into account parts of the model that actually do not correspond to actual application behaviors.

Improving the model relevance is therefore an important challenge in order to ensure the most accurate control on applications. Considering the approach detailed in Section IV-B, such improvements can be obtained from variable observations and determination of relationships between parameter domains of the application methods. Such relationships can be learned through probabilistic approaches (e.g., [22]) or system identification techniques ([23]).

### E. Multi-threading and Concurrent Control.

Most software applications possess several components running on different threads. They can therefore be modeled as a composition of FSMs, each modeling a component. In this section we consider the control of concurrent systems. Classical supervisory control techniques require that a single FSM represent the system behaviors. Such an FSM can be obtained by computing the composition of the FSM representing each component. However, this computation leads to a state explosion problem and represents an important challenge of supervisory control theory. Some work on control of concurrent systems have been conducted (e.g., [24], [25], [26]) and even in the case of models with variables in [18]. However, more work is required in this area in order to improve the state-of-the-art. Moreover, one specificity of the approach described in Section III is that the number of components running concurrently varies over time. Finally, the

composition mode between components is different from the usual synchronous and parallel composition that is considered in classical supervisory control.

#### F. Designing Control Objectives.

Our proposed approach relies on the synthesis of supervisors from a model of the system behaviors and a control objective. This control objective is represented by a FSM and encodes safety properties over the system behaviors. It is possible, for instance, to describe what methods must not be executed after some given executions. If the control objective also provides information on the variables of the system, then it allows for the description of complex conditions under which some method calls must not be executed.

As mentioned in Section III-D and illustrated in Figure 6, the control objective may be obtained manually, and automating its design is a difficult challenge.

Some results in this direction have been obtained in [10] in the specific case of un-handled exceptions. As a general matter, tackling the automatic design of control objectives is very much related to automatic fault and anomaly detection (e.g., [27]) as well as automatic diagnosis. Therefore techniques related to automatic diagnosis can contribute to automating control objective designs and should be further investigated in the context of automatic supervisory control.

Control policies can also consider performing some actions whenever control is applied to the system, creating new behaviors. This can be subject to different strategies from which one needs to be selected.

#### V. CONCLUSION

This paper describes the EU FP7 FastFIX project, which tackles issues related to remote software maintenance. In order to achieve this, the project considers approaches relying on context elicitation, event correlation, fault-replication and self-healing. After introducing the general objectives addressed within FastFIX, we describe its self-healing approach and early development, which aim to automate the generation of patches, hence reducing time and cost related to some of the corrective maintenance tasks.

This self-healing approach relies on control theory. We describe its different components and phases and introduce some of its challenges. This paper points out the challenges that are related to supervisory control theory. It also describes some challenges, such as automating the design of control objectives as well as introducing new behaviors into the application.

#### REFERENCES

- [1] B. P. Lientz, E. B. Swanson, and G. E. Tompkins, "Characteristics of application software maintenance," *Commun. ACM*, vol. 21, pp. 466–471, June 1978. [Online]. Available: <http://doi.acm.org/10.1145/359511.359522>
- [2] M. Davidsen and J. Krogstie, "Information systems evolution over the last 15 years," in *Advanced Information Systems Engineering*. Springer, 2010, pp. 296–301.
- [3] J. Radatz, "IEEE standard glossary of software engineering terminology," *IEEE Std 610121990*, vol. 121990, 1990.
- [4] D. Ghosh, R. Sharman, H. Raghav Rao, and S. Upadhyaya, "Self-healing systems - survey and synthesis," *Decis. Support Syst.*, vol. 42, no. 4, pp. 2164–2185, 2007.
- [5] O. Shehory, J. Martinez, A. Andrzejak, C. Cappiello, W. Funika, D. Kondo, L. Mariani, B. Satzger, M. Schmid, A. Andrzejak *et al.*, "Self-Healing and Recovery Methods and their Classification," *Self*, 2009.
- [6] H. Psailer and S. Dustdar, "A survey on self-healing systems: approaches and systems," *Computing*, vol. 91, pp. 43–73, 2011, 10.1007/s00607-010-0107-y. [Online]. Available: <http://dx.doi.org/10.1007/s00607-010-0107-y>
- [7] "Fastfix project consortium: Fastfix project homepage, [www.fastfixproject.eu/](http://www.fastfixproject.eu/)."
- [8] D. Bakken, R. Rameswaran, D. Blough, A. Franz, and T. Palmer, "Data obfuscation: anonymity and desensitization of usable data sets," *Security & Privacy, IEEE*, vol. 2, no. 6, pp. 34–41, 2004.
- [9] J. Hellerstein, Y. Diao, S. Parekh, and D. Tilbury, *Feedback control of computing systems*. Wiley-IEEE Press, 2004.
- [10] B. Gaudin, E. Vassev, M. Hinchey, and P. Nixon, "A control theory based approach for self-healing of un-handled runtime exceptions," in *8th International Conference on Autonomic Computing (ICAC 2011)*, Karlsruhe, Germany, June 2011.
- [11] B. Gaudin and A. Bagnato, "Software maintenance through supervisory control," in *34th annual IEEE Software Engineering Workshop*, June 2011.
- [12] P. J. Ramadge and W. Wonham, "Supervisory control of discrete event processes," in *Feedback Control of Linear and Nonlinear Systems*, ser. LNCIS, vol. 39. Springer-Verlag, Berlin, Germany, 1982, pp. 202–214.
- [13] W. M. Wonham, "Notes on control of discrete-event systems," Department of Electrical and Computer Engineering University of Toronto, Tech. Rep. ECE 1636F/1637S, July 2003.
- [14] M. Weiser, "Program slicing," in *Proceedings of the 5th international conference on Software engineering*. IEEE Press, 1981, pp. 439–449.
- [15] M. Skoldstam, K. Akesson, and M. Fabian, "Supervisory control applied to automata extended with variables-revised," *Relatório técnico, Goteborg: Chalmers University of Technology*, 2008.
- [16] T. Le Gall, B. Jeannot, and H. Marchand, "Supervisory control of infinite symbolic systems using abstract interpretation," in *44nd IEEE Conference on Decision and Control (CDC'05) and Control and European Control Conference ECC 2005*, Seville (Spain), December 2005, pp. 31–35.
- [17] R. Kumar and V. Garg, "On computation of state avoidance control for infinite state systems in assignment program framework," *Automation Science and Engineering, IEEE Transactions on*, vol. 2, no. 1, pp. 87–91, 2005.
- [18] B. Gaudin and P. Deussen, "Supervisory control on concurrent discrete event systems with variables," *American Control Conference, 2007. ACC'07*, pp. 4274–4279, 2007.
- [19] P. Cousot and R. Cousot, "Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints," in *Conference Record of the Fourth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. Los Angeles, California: ACM Press, New York, NY, 1977, pp. 238–252.
- [20] J. Corbett, M. Dwyer, J. Hatcliff, S. Laubach, C. Pasareanu, and H. Zheng, "Bandera: Extracting finite-state models from Java source code," in *Software Engineering, 2000. Proceedings of the 2000 International Conference on*. IEEE, 2002, pp. 439–448.
- [21] N. Gruska, A. Wasylkowski, and A. Zeller, "Learning from 6,000 projects: lightweight cross-project anomaly detection," in *ISSTA '10: Proceedings of the 19th international symposium on Software testing and analysis*. New York, NY, USA: ACM, 2010, pp. 119–130.
- [22] R. Michalski, J. Carbonell, and T. Mitchell, *Machine learning: An artificial intelligence approach*. Morgan Kaufmann Pub, 1983.
- [23] L. Ljung and E. Ljung, *System identification: theory for the user*. Prentice-Hall Upper Saddle River, NJ, 1987, vol. 280.
- [24] Y. Willner and M. Heymann, "Supervisory control of concurrent discrete-event systems," *International Journal of Control*, vol. 54, no. 5, pp. 1143–1169, 1991.
- [25] M. deQueiroz and J. Cury, "Modular supervisory control of large scale discrete-event systems," in *Discrete Event Systems: Analysis and Control. Proc. WODES'00*. Kluwer Academic, 2000, pp. 103–110.
- [26] B. Gaudin and H. Marchand, "An efficient modular method for the control of concurrent discrete event systems: A language-based approach," *Discrete Event Dyn Syst*, vol. 17, no. 2, pp. 179–209, Apr 2007.
- [27] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

# Autonomic Execution of Computational Workflows

Tomasz Haupt, Nitin Sukhija, Igor Zhuk  
Mississippi State University  
Center for Advanced Vehicular Systems, Box 5405  
Mississippi State, MS 39762  
USA  
Email: {haupt, nitin, igorzhuk}@cavs.msstate.edu

**Abstract**—This paper describes the application of an autonomic paradigm to manage the complexity of software systems such as computational workflows. To demonstrate our approach, the workflow and the services comprising it are treated as managed resources controlled by hierarchically organized autonomic managers. By applying service-oriented software engineering principles, in particular enterprise integration patterns, we have developed a scalable, agile, self-healing environment for execution of dynamic, data-driven workflows which are capable of assuring scientific fidelity despite unavoidable faults and without human intervention.

## I. INTRODUCTION

Support for scientific workflows is now recognized as a crucial element of cyberinfrastructure, facilitating e-Science. Typically sitting on top of a middleware layer, scientific workflows are means by which scientists can model, design, execute, debug, re-configure and re-run their analysis and visualization pipelines.

There are many ways of implementing scientific workflows [1, 2]; however, with the advent of Grid and Cloud computing, most of the current efforts adopt Service-Oriented Architectures (SOA). Consequently, research on workflow management systems highlights methodologies of service composition and orchestration. To that end, this paper focuses on particular aspects of service-oriented workflow system development, namely, the scientific fidelity, fault tolerance, adaptivity, and management of complexity. The ideas presented in this paper are illustrated by an exemplary implementation of an adaptive computational workflow.

Scientific fidelity refers to a software system's ability to deliver reliable, trustworthy computational results; i.e., the end user can trust that the output is not distorted by erroneous information resulting from unreported failures of the workflow and/or its components. To achieve such fidelity, the system for executing the computational workflows must be capable of detecting faults and abnormalities and performing corrective actions, whenever

feasible. The system can react to faults and abnormalities either by protecting against faults before they occur (possible when an abnormality has been detected), or by recovering after a fault has happened. In the latter case, the detection of a fault may also help detect an abnormality, which could then prompt a corrective action to prevent future failures of the same type.

In addition to a direct recovery from a point failure by automatic fixing the cause of the problem and retrying, it is desirable that the system has a capability to respond to an abnormality by adaptation. It may include use of an alternative service instance, correction of the request due to a change of the service interface, the selection of an alternative algorithm to be used by the service (or the code submitted by that service), or the modification of the workflow specification, i.e., the change of the execution path, perhaps using alternative or optional workflow nodes. Since the adaptations forced by the failures may be data-driven and thus unpredictable, enforcing the scientific fidelity is of critical importance.

Unfortunately, the enforcement of scientific fidelity adds to the complexity of the system; if not managed properly, this added complexity might actually decrease the reliability and maintainability of the overall system, thereby defeating its ultimate purpose. The situation is further complicated by the fact that the end user, a domain specialist that composes and runs the workflow, may not know or care about possible failure modes below the application level or the methods for remedying them. Conversely, an IT specialist maintaining the system typically has very little, if any, knowledge of the business logic of the workflow.

Herein, we address scientific fidelity, fault tolerance, adaptivity, and the management of complexity, applying (1) the concepts of Autonomic Computing, in particular self-management and self-healing, and (2) service-oriented software engineering, in particular exploiting the capabilities of the Enterprise Service Bus for dynamic message routing.

The remainder of this paper is organized as follows. In Section II we describe the concepts of Autonomic Computing (AC). In Section III we define dynamic computational workflows and explain the benefits of applying an AC paradigm to manage the complexity of the

---

This work has been supported by the U.S. Department of Energy, under contract DE-FC26-06NT42755 and NSF Grant No. 826547

system while assuring the scientific fidelity of the results. In Section IV we discuss the concepts of the Service-Oriented Software Engineering (SOSE), including Enterprise Service Bus (ESB) and Enterprise Integration Patterns (EIP) and their potential for enabling AC, and in section V we present our implementation of an autonomic workflow. Finally, in Section VI we offer our conclusions.

## II. AUTONOMIC COMPUTING

Autonomic Computing (AC) concepts [3, 4] have been effectively used to manage enterprise systems and applications; now they provide a promising approach to address the challenges of complexity management. Analogous to the human body, where the autonomic nervous system responds to stimuli by adapting the body to its needs and to the environment without involving the conscience, AC-driven complexity management is achieved by creating self-managing environments capable of dynamically adapting to unpredictable changes using only high-level guidance or intervention from the users. Following this concept, each element of a computational system is managed by its own autonomic control loop, involving monitoring, analysis, planning, and execution (M-A-P-E, cf. [1]), realizing a set of predefined system policies. These individual control loops will then collaborate, i.e., communicate and negotiate with other autonomic managers which control other aspects of the computations.

Furthermore, as Parashar expressed it, “the autonomic approach mimics nature’s way of managing the complexity: complex patterns emerge from the interaction of millions of organisms that organize themselves in an autonomous, adaptive way by following relatively simple behavioral rules. In order to apply this approach, the organization of computations over large complex systems, computations must be broken into small, self-contained chunks, each capable of expressing autonomous behavior in its interactions with other chunks” [4]. The goal of autonomic computing, then, is to manage complex computations via sets of predefined, simple rules that define the system’s responses to failures and unpredictable changes in the computational environment, thus providing means for recovery from faults and/or adaptation of the system without direct human intervention.

## III. COMPUTATIONAL WORKFLOWS

A computational workflow is a sequence of computational and data management tasks in a scientific application. Organizing the scientific analysis into a workflow significantly reduces the complexity of the application: the monolithic and thus difficult to maintain application is decomposed into simpler, independent modules (workflow nodes) focused on specific aspects of the problem at hand. Individual components can be reused for different applications (workflows), and the business logic of the overall application can be tuned and improved by

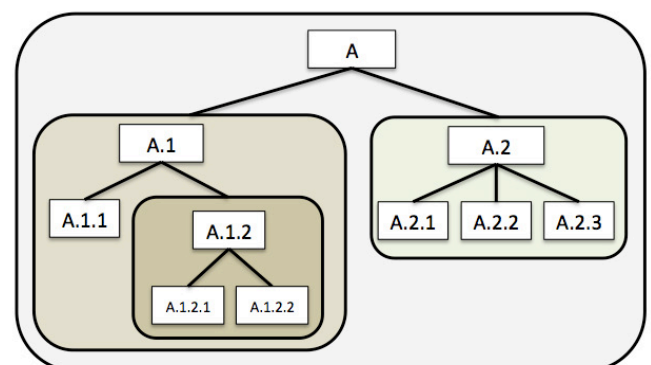
reconfiguring the workflow, i.e., changing the sequence of tasks.

Our goal is to provide a workflow execution environment with the capability to recover from faults of the workflow components and consequently to prevent erroneous data from failed components from entering the final result set (“scientific fidelity”) or crippling the business logic of the workflow. Furthermore, we envision the workflow execution environment as capable of autonomous “self-healing,” that is, correcting non-fatal failures without human intervention.

The autonomic execution of a workflow is even more important in the case of dynamic workflows in which the sequence of the components changes unpredictably (e.g., is data driven), and the same component can be invoked many times. The multistep design optimization (MDO) is an example of a dynamic workflow.

### A. Multistep Design Optimization

Many complex engineering systems are more readily optimized when they are decomposed into a number of subsystems with partitioned design variables and separate objective functions and design constraints. Following the Analytical Target Cascading (ATC) approach [5, 6], the resulting workflow has a layered architecture of decomposed systems, as schematically shown in Figure 1. The hierarchy can be expanded to include several levels, each containing multiple elements. This hierarchical architecture, applicable to integrated product-material design, offers autonomy to each element to optimize its own objective function according to an element-specific set of constraints, which are, in turn, based upon either inputs from lower-level elements and design targets or demands imposed by corresponding upper-level elements. Because the number of design variables in each element represents a fraction of the total set, the dimensionality of each element optimization problem is reduced. Hierarchically decomposed systems are naturally suitable for parallel computing and decentralized optimization approaches, but they require a careful coordination strategy in the ensuing iterative solution process to ensure satisfaction of system-level design criteria



**Figure 1:** Idealized hierarchical workflow for multistep design optimization.

and proper convergence to an optimum design.

### B. Idealized dynamic workflow

The details of ATC and its application for design optimization are beyond the scope of this paper. What is of interest here is the structure of the resulting dynamic workflow. The workflow comprises a number of nodes (cf., Figure 1), and each node implements the same pattern: given initial values, it performs an optimization of the subsystem by submitting a job to minimize its objective function. Depending on the results of the subsystem optimization, the children nodes are dispatched, or the results are returned to the parent node. This dependency on the optimization results at each level makes the overall computations dynamic: at the beginning of the process, it is unknown how many times each node will be invoked, and consequently, the sequence of job submissions is unpredictable.

ATC defines the rules for controlling the execution of the workflow, that is, the sequence of invoking workflow nodes and convergence criteria. However, these rules implicitly assume that all submitted jobs complete successfully and deliver trustworthy results. A failure of a single job may cripple the entire workflow, wasting all the results obtained before the fault occurred. Even worse, an unreliable result caused by an unreported failure may distort the end results.

### C. Job Execution Service

Since the core functionality of the workflow node is submitting a job, let us examine an example implementation of a Globus-based Job Execution Service (JES) [7], as shown in Figure 2. Given a job descriptor (a string in Resource Specification Language (RSL) [8]) as the service request argument, the service selects the target machine (e.g., site 1 or 2), performs data staging, and submits the job to the Globus Resource Allocation Manager (GRAM) [9] server at the selected site. If the submission succeeds, the job submission service enters the job id (returned by GRAM) to the job monitoring (JM) service, and responds with an acknowledgement. Otherwise, it responds with a job

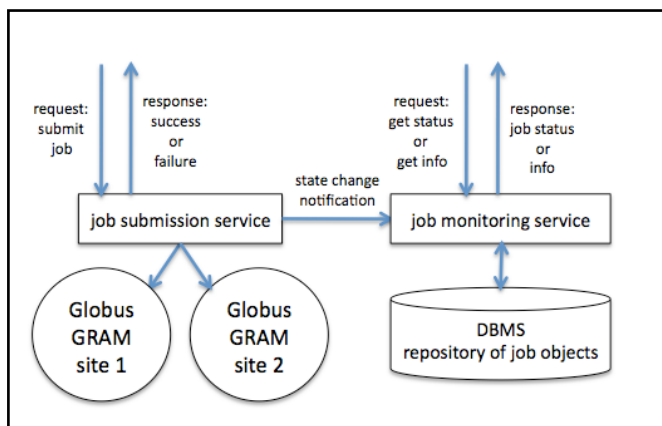


Figure 2: Job Execution Service

submission failure message. All changes of the state of the job (pending, running, completed) reported by GRAM are forwarded to the JM service. The submitting client then polls for job status by sending requests to the JM service until the job is completed. At that moment, the client retrieves job information comprising of the actual location of the job stdout, stderr, and any other available output files.

### D. Failure modes

A job submitted through the JES may fail (i.e., no or unreliable results are produced) in many different ways. Following the patterns recognized in [10], we can group these failures into four categories or levels:

1. The service may not be responding to or reporting an internal error, that is, a service level failure.
2. The job submission may fail because of expired credentials, errors in RSL, shutdown of the target machine, or other specific job submission service level failure.
3. The job may crash (non-zero exit value) because of, for example, missing input data, insufficient memory, time limit, or other system level failure.
4. The job may complete with exit value=0 but still produce unreliable results, such as non-converged optimization or other application level failure.

Although demonstrated here for JES, this categorization is generic and can be applied to any type of service.

Many of these faults can be remedied programmatically. For example, in the non-responding service, a peer service can be invoked instead. Expired credentials can be refreshed; memory requirements or execution time limits can be tuned in a re-generated RSL; lack of convergence can be remedied by selecting another algorithm, changing the initial values, or modification of the constraints on the values of design variables.

Recovering from these failures could be incorporated into the workflow specification, but it would add unnecessary complexity to an already complex set of ATC rules. Furthermore, the domain expert who applies the ATC rules may not know or understand the failure modes and the remedies that could or should be applied, while the IT professional responsible for the deployment and maintenance of the services typically has little, if any, understanding of the ATC rules. It is thus desirable to manage the complexity of the ATC workflow (or any other computational workflow) by separating failure recovery from the business logic of the workflow, thereby designating fault recovery as a property of the execution environment and not of the workflow itself. This property, often referred to as self-healing, can be achieved by applying AC concepts.

### E. Autonomic execution of jobs

The complexity of computational workflow management due to unpredictable job failures can be addressed by treating jobs as managed resources. Following the AC approach, the job should be managed by its own autonomic control loop that would guarantee that the results generated



by the job meet criteria specified in predefined system policies. To achieve that, the JES must be augmented with additional functionality for assessing the quality of the results. To earn the qualification of being autonomic, the manager implementing the control loop to enforce the scientific fidelity of the results must be independent of the business logic defined in the workflow specification.

The taxonomies of failure modes help design monitors and analyzers of M-A-P-E autonomic managers, while the taxonomy of remedies allows design of the planners. Typically the planners would modify the service request (e.g., the job specification) and re-invoke the managed service (e.g., resubmit the job). These taxonomies will be necessarily open, as it is unreasonable to expect that all possible failure modes will be captured at the design time. Furthermore, the repertoire of remedies will grow as the knowledge of the system increases. Consequently, the design of the system must allow for adaptive runtime changes (defined by configuration files and/or policies) and learning.

The autonomic job manager envisioned here acts *reactively*: it responds to faults after they have actually happened. Such a manager should be complemented with *proactive* behavior: corrective actions taken before a predictable fault occurs (e.g., as in [11]). For example, the availability of the disk space could be monitored regularly (independently of whether a job is submitted or not), and if the available space is less than a predefined threshold value, some corrective action is taken so that when a job is submitted, it will not crash because of lack of disk space.

It follows that the AC paradigm requires adding a large number of new components: monitors, analyzers, planners, and executors. Therefore, if the system is not designed carefully, the complexity will move from the workflow's business logic to the execution environment, defeating one of our principal goals.

#### IV. SERVICE-ORIENTED SOFTWARE ENGINEERING

The Service-Oriented Computing (SOC) paradigm uses services to support the development of rapid, low-cost, interoperable, evolvable, and massively distributed applications [12]. Services are autonomous, platform independent entities that can be described, published, and discovered. The visionary promise of SOC is that it is possible to easily assemble application components into a loosely coupled network of services that can create dynamic business processes and agile applications which span organizations and computing platforms [13].

##### A. Enterprise Service Bus

The requirements to provide capable and manageable integration of services are coalescing into the concept of the *Enterprise Service Bus* (ESB) [14, 15], implementing Java Business Integration (JBI) [16] specification. An ESB is a software construct that provides fundamental services for complex architectures via an event-driven and standards-based messaging engine (the bus). With ESB, requestors and

service providers are no longer interacting directly with each other; rather they exchange messages through the bus, and the messages can then be processed by mediations (e.g., message transformation, routing, monitoring). Mediations implement the integration and communication logic, and they are the means by which ESB can ensure that services interconnect successfully. As a result, the ESB acts as the intermediary layer between a portal server and the back-end data sources with which the data portal interacts [12].

##### B. Self-managing of Service-Oriented Systems

During the last few years, the issue of self-management and support for adaptivity of service-oriented systems has attracted attention of many researchers [17-21]. Most of the proposed solutions to support this autonomic behavior place the service bus in the center of the architecture, taking advantage of dynamic routing features offered by most implementations of the bus.

For example, S-Cube [19] adopts a *publisher-subscriber* [22] pattern to manage the flow of messages in the bus. A central Service Adaptation and Monitoring (SAM) module subscribes to events fired by monitors of all managed resources. Based on the signature of the received event (context and runtime values) and adaptation strategies retrieved in real time from the Adaptation Manager, SAM automatically selects a suitable adaptation action and invokes it by firing an event (a one-way message) to be consumed by the adaptation gateway, which in turn, dynamically routes the message to a service capable of performing the corrective action. For the purposes of S-Cube, the adaptation strategy is an XML document implementing the router *slip pattern* [22], that is, it specifies the sequence of services to be invoked and message transformations needed in between.

The Ceylon autonomic system [11] exploits the flexibility of the *publisher-subscriber pattern* even further by implementing planners (in the M-A-P-E paradigm) capable of correlating independent but related events.

Many authors recognize the arising problem with developing such systems: heterogeneity of messages traveling through the bus and, associated with it, the increasing complexity of the dynamic routing. Multifactor-driven hierarchical routing (MDHR) [20] distinguishes three layers for message routing on an ESB: *the message layer* for standard ESB mechanisms for message delivery (content-based routing, itinerary-based routing, or static routing); *the application layer* that encapsulates legacy applications; and *the business layer* allowing for external mechanisms for message routing as defined by domain specific language of a business process. The virtualization of services supported by ESB is also exploited by the DRESR framework [21] to allow for dynamic changes in business and service processes.

The heterogeneity of messages and the resulting complexity of routers comes from different "types of service variability" [10] that require an adaptation of the system at one of four levels: (1) workflow composition (e.g., using

optional or alternative steps); (2) composition (e.g., alternative implementations to be bound at runtime); (3) interface variability (mismatch between actual and published service interfaces); and (4) logic variability (alternative business logic of a service). Handling the messages, which are carrying information about a system state change, an abnormality, or a failure at one of these levels, requires identifying a “signature” from a message and using it to select alternative services as defined in a registry and which are capable of modifying the workflow or service endpoints, applying a transformation, or changing a service configuration as needed.

The complexity of recognizing the message content needed to apply content-based routing seems to originate from the design feature that is common for the above-described implementations: the centralization of the adaptation control, leading to an unnecessary complexity of the autonomic environment. In this paper we propose an alternative approach, based on the foundations of AC. We propose a decomposition of the central complex decision maker, such as S-Cube’s SAM, into a large number of small components implementing simple behavioral patterns, and use of the full power of a rich set of Enterprise Integration Patterns (EIP) [22] offered by ESB to integrate them into a dynamic, autonomic system. By mimicking biological systems, the inherent complexity of an autonomic system can be thereby reduced to a collection of easy to maintain and configure services, each following simple rules.

## V. COMPUTATIONAL WORKFLOW AS A MANAGED RESOURCE

Within the SOA paradigm, a workflow is a composite service, that is, a service that combines other services, where the ‘constituent’ services interact with each other through an exchange of messages.

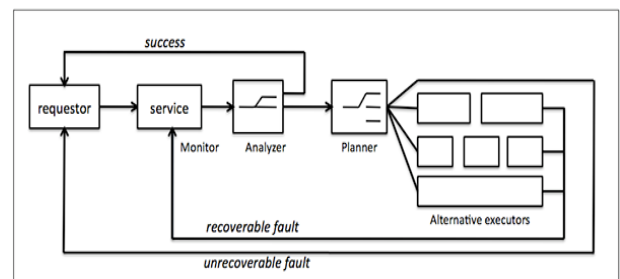
A message traveling in the ESB is a Java object implementing `javax.jbi.messaging.NormalizedMessage` interface. This interface mandates, among other things, the message properties (“headers”) and message content (“body”). A special case is a Fault message (`javax.jbi.messaging.Fault` interface that extends `NormalizedMessage` interface). A Fault message is created when the service cannot complete the processing of a request. It may happen for many reasons, such as missing or invalid data, insufficient resources, a bug in the implementation, or other unpredicted circumstances. This mechanism can be further exploited by introducing exceptions at the business level: if the result to be returned by the service does not satisfy requirements specified by a predefined policy, an exception is thrown. The content of the Fault message provides the details of the exception that triggered the service failure.

The Java objects representing the messages within the bus are converted to “wire-ready” messages (e.g., SOAP over HTTP) by ESB binding components when communicating with the external service requestor and provider (cf. Figure 4).

### A. Service as a managed resource

Catching exceptions by the service implementation itself is a form of service **monitoring**. The clear distinction between successful and fault messages can be used as a simple rule for content-based routing: unless the event message is of type `Fault`, the message is routed to the service specified in the routing slip (*Routing Slip* [22] pattern); otherwise, it is routed to an alternative endpoint (*Detour* [22] pattern). As a consequence, this simple router acts also as an **analyzer**, the second element of an autonomic manager. The intention here is straightforward: should a fault happen, the system makes an attempt to recover from it by applying a detour and, when the problem is resolved, the requestor gets a successful, trustworthy response without knowing that a corrective action has been autonomically performed. The detour results in forwarding the `Fault` message to a **planner**, that is, a dedicated service, which is capable of identifying the cause of the failure and of selecting one of a set of predefined but configurable corrective actions. The corrective actions are driven by a policy (e.g., articulated as XML documents) so that the planner service can translate the signature of the failure encoded in the content of the `Fault` message into a sequence of actions to be taken following the *routing slip* pattern. Since the planner must understand the signatures of failures, the functionality of the managed service and the planner are tightly correlated, and therefore each managed service should be associated with a corresponding planner. Should the planner fail to recognize the fault or devise a plan for corrective action (e.g., no policy defined), it throws an exception. In general, there is no reason to define a planner for the planner service; therefore, the router sends an “unrecoverable fault” message to the requestor: at this point, there is nothing that can be done to recover from the failure.

The planner should be a separate service because the monitor (i.e., the managed service itself) is capable of only identifying *what* is wrong, but, in general, does not have enough information to determine *why* the exception happened. For example, the service can easily recognize that the input data is invalid, but it is outside the service’s scope to determine what steps need to be taken to correct the data. Furthermore, the determination of the corrective



**Figure 3:** The concept of autonomic manager for a single service



actions may require a correlation of information coming from several monitors.

The services defined in the routing slip serve as **executors** of the autonomic manager. The intent here is to remove the conditions that led to the fault of and then to re-invoke the managed service. For example, in the case of a job submission, invoking a sequence of services may be necessary to modify the job's RSL description and/or its input files, and then, to re-submit the job. Any already deployed service can be used as an executor, if its functionality happens to serve the purpose (e.g., RSL generator service); otherwise dedicated services must be developed and deployed. In addition, if applicable, a service acting as the executor of the autonomic manager may make an attempt to adapt the system to avoid the same type of faults in the future.

Finally, all actions need to be logged into the database (*message store* [22] pattern), the knowledge component of the autonomic manager. It is necessary to allow the end user to monitor the progress in real time (e.g., through an interactive GUI), and to identify the sources of unrecoverable faults. Furthermore, the planners use the database to correlate responses from different services, such as monitors of the system state (e.g., is there enough disk space available?), or to break infinite loops or deadlocks if the sequence of the applied corrections does not converge.

To summarize, a service is managed by a M-A-P-E autonomic manager, schematically shown in Figure 3: should the service fail to complete successfully (the monitor functionality), the service response is detoured (the analysis functionality) to a planner service that determines the sequence of corrective actions to be preformed by executors. Once the conditions leading to the fault have been removed, the managed service is re-invoked. Should the planner or executors fail to recover from the fault, an "unrecoverable fault" message is returned to the service requestor.

We have realized this autonomic behavior using Apache ServiceMix [23] implementation of the ESB. The requests from external requestors are received by a Binding Component, as shown in Figure 4. Binding components are standard JBI components that plug into NMR and provide transport independence to NMR and Service Engines (SE).

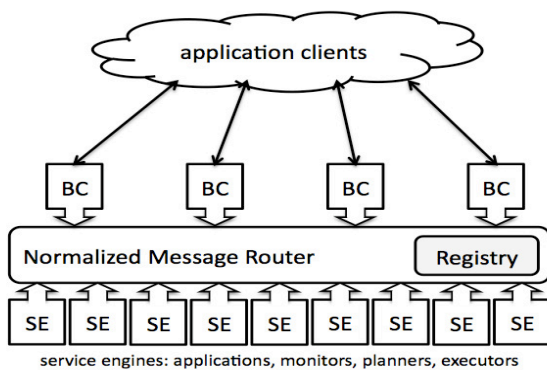


Figure 4: The architecture of Enterprise Service Bus

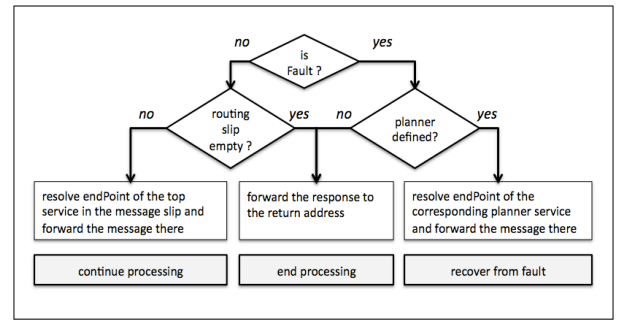


Figure 5: Routing algorithm

The role of binding components is to isolate communication protocols from JBI containers so that Service Engines are completely decoupled from the communication infrastructure.

The routing decisions in our implementation of the standard `org.apache.service.jbi.nmr.broker` interface are based on three message properties: routing slip, return address and fault, following a simple algorithm shown in Figure 5. In the absence of a fault, the first element in the routing slip is resolved via registry to an endpoint of a Service Engine (e.g., JES). The fault messages are routed to the corresponding planner Service Engines with the endpoint defined in the service registry. If the routing slip is empty, or the endpoint of the planner cannot be resolved, the fault message is returned to the requestor, using the return address embedded as the message property.

This implementation treats all services symmetrically, that is, the router is not aware of the business logic implemented by a service. In particular, it does not distinguish between the managed services and the various components of autonomic managers at different levels. The router only distinguishes between regular and fault messages, and it follows the routing slips created by invoked services and embedded as the message property. This approach reduces the complexity of the services, their managers, and the router to a set of simple rules.

Note that because of the symmetrical treatment of the services, the elements of the autonomic managers are also autonomically managed: if the event message generated by a planner or an executor is of type `Fault`, the router recognizes the failure and re-routes the message so that corrective actions can be taken. This feature is rarely discussed in literature.

#### B. Scientific fidelity of a service response

A faultless completion of a service does not necessarily guarantee that the service's response satisfies criteria specified in a policy. An autonomic validation of the response must be performed as well. As an example, a job executed via JES may produce unreliable results (e.g., the minimization process has not converged). It would be a software engineering mistake to add the capability of

detecting application-specific failures to the otherwise generic JES.

The validation of the results is therefore performed by a dedicated, validating service, which is automatically invoked after the service that produces the results completes. It can be easily achieved with ESB, exploiting its support for the virtualization of services. The original request (e.g., “run a job”) is dynamically re-routed to a process manager service (implementing *process manger* [22] pattern) that inserts a routing slip to the message (in this example, run job, verify the exit value, and validate output). As a result, the sequence of services specified in the slip is autonomically executed: should the service’s response not satisfy the criteria, the validating service fires a Fault message, which, in turn, prompts the router to schedule a detour to the associated planner in order to initiate corrective action. Ultimately, the final response of the service to the original requestor is either trustworthy or it is an explicit fault message (“unrecoverable fault”), should no corrective action or other resolution be found.

### C. Workflow as a managed resource

Any workflow engine, e.g., a BPEL-based [24] one, will benefit from the autonomous execution of services, in particular, when the results produced by the services are autonomically verified. However, the complexity of dynamic workflows, especially those for which the determination of the subsequent actions can be defined by applying a set of rules (as is the workflow for hierarchical multistep design optimizations), can be reduced by applying the same autonomic approach as we have for a single service, that is, by treating the workflow as a managed resource. This creates a hierarchy of resource managers, with the workflow autonomic manager consuming “unrecoverable fault” messages generated by the individual services’ autonomic managers.

Figure 6 shows the autonomic execution of a single node (node A.1 in Figure 1) of the idealized multistep optimization workflow. The execution begins with a planning activity based on the predefined rules (here, ATC). The planning is performed by a dedicated process manager service that updates the routing slip of the received message. There are three possible outcomes of this manager service: (1) the subsystem represented by A.1 node system is already optimized, and its optimized values are to be returned to its parent (here, node A); (2) the subsystem needs to be optimized by first optimizing its children (subsystems A.1.1 and A.1.2), followed by submitting a job to minimize objective function of subsystem A.1; (3) the optimization of the subsystem failed. The first case is handled by sending a message to the next service in the routing slip of the event that triggered this planning activity. The second case is processed by sending two messages, one with “process A.1.1” and “optimize A1,” and the other with “process A.1.2” and “optimize A1” added to the top of the routing slip. The last case results in sending a Fault message, which

triggers an autonomic recovery attempt. In each case, one or more messages are sent to the bus, and the router delivers them to the recipient following the simple algorithm shown in Figure 5.

Services labeled “process A.1.1” and “process A.1.2” are nodes in the workflow, and they are implemented in the same way as discussed for node A.1 (recursion). The “optimize A1” is actually a composite service: it aggregates (through updating of a routing slip) services for the creation of the job descriptor, the job input files, and job execution, which was discussed in detail above. Each of these services may fail, which results in sending a fault message that is appropriately routed for autonomic recovery (for clarity, this is not shown in Figure 6). The “optimize A1” is triggered by two independent events: successful completion of either “optimize A.1.1” or “optimize A.1.2” service (*aggregator* [22] pattern). In our implementation, the message store was used to correlate the events. When one of the children nodes sends the message, the store is searched for the message from the other child. If it is not found, the service exits without sending any events. For scientific fidelity, it is paramount that the messages from the children nodes are delivered to the “Optimize A.1” if, and only if, their results are trustworthy. Similarly, the decision whether or not subsystem A.1 has been optimized is based on the trustworthy result of “optimize A.1.”

The system is easy to implement and can be deployed gradually, in small steps. The critical first step is to implement the ESB router capable of routing messages according to the routing slip embedded in the messages and of detouring the fault messages. Initially, this custom router preserves the original functionality of the system; for example, a fault the message is routed to the requestor unchanged since no corresponding planner is defined in the registry. Then the autonomic behavior can be added by deploying planners and process manager services, one by one, as experience with detecting failures and devising recovery procedures is accumulated. Furthermore, the self-healing property of the system can be progressively enhanced by adding “stand-alone” monitors which add to the message store updates on the status of other system resources that influence the reliability of the system.

It is important that the implementation of the services

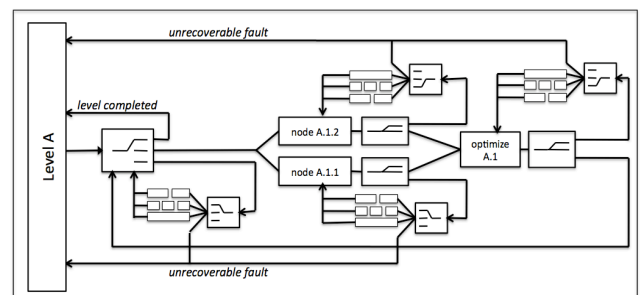


Figure 6: Autonomic execution of a multistep optimization workflow node

accommodate processing of policy documents that define the criteria for the determination of trustworthiness of results and specify the corrective actions. Following these guidelines enables updates of the policies at runtime that result in behavioral changes of the system, leading to a truly adaptive autonomic system.

## VI. SUMMARY

In this paper we have described an autonomic environment for execution of dynamic, rule-based computational workflows. This environment not only makes the best effort to recover from faults, it also guarantees the scientific fidelity of the results, in particular, that the final outcome of complex computations are not distorted by erroneous information resulting from unreported system- and/or application-level failures of the workflow and/or its components.

The autonomic behavior has been achieved by harnessing Service-Oriented Software Engineering, most notably, by employing the Enterprise Service Bus, exploiting the Java Business Integration specification, and applying Enterprise Integration Patterns. By defining a set of very simple rules that apply to autonomous, loosely coupled services, we have generated a very complex autonomic behavior involving iterative, and possibly recursive, sequences of service invocations, thus mimicking biological systems. Scientific fidelity is achieved by enforcing service responses that meet criteria specified by configurable policies.

Failures to meet the criteria are caught as exceptions that result in firing fault messages. The custom ESB router, without any knowledge of the workflow's business logic of the workflow, detours all fault messages to specialized services that, based on the signature of the fault, plan corrective actions through inserting routing slips to messages. Those multiple planner services are independent of each other, each addressing specific problems and making decisions based on the policies defined in the configuration files that can be modified (adapted) at run time.

## REFERENCES

- [1] Jia Yu and Rajkumar Buyya, A Taxonomy of Scientific Workflow Systems for Grid Computing, Special Issue on Scientific Workflows, SIGMOD Record, ACM press, Volume 34, Number 3, 2005.
- [2] E. Deelman and Y. Gil. "Managing Large-Scale Scientific Workflows in Distributed Environments: Experiences and Challenges," Proceedings of the Workshop on Scientific Workflows and Business Workflow Standards in e-Science, The Second IEEE International Conference on e-Science and Grid Computing, Amsterdam, The Netherlands, December 4-6, 2006
- [3] J. O. Kephart, D. M. Chess, "The Vision of Autonomic Computing," Computer 36, 1 (2003), pp. 41-50.
- [4] M. Parashar, "Autonomic Grid Computing: Concepts, Requirements, and Infrastructure," in "Autonomic Computing", M. Parashar, S. Harriri, (Eds), CRC Press 2007
- [5] E.H. Miller, N.F. Michelena, M.K. Kim, and P.Y. Papalambros, "A System Partitioning and Optimization Approach to Target Cascading," Proceedings of the 12th International Conference on Engineering Design, Munich, Germany, 1999.
- [6] M.K. Kim, N.F. Michelena, P.Y. Papalambros, P.Y., and T. Jiang, "Target Cascading in Optimal System Design," *Journal of Mechanical Design*, Vol. 125, pp. 474-480 2003.
- [7] T. Haupt, A. Voruganti, A. Kalyanasundaram, and I. Zhuk. 2006. Grid-Based System for Product Design Optimization. In *Proceedings of the Second IEEE International Conference on e-Science and Grid Computing (E-SCIENCE '06)*. IEEE Computer Society, Washington, DC, USA, 46-52.
- [8] Globus Toolkit 2.4, Resource Specification Language (RSL): [http://www.globus.org/toolkit/docs/2.4/gram/rs\\_l\\_spec1.html](http://www.globus.org/toolkit/docs/2.4/gram/rs_l_spec1.html)
- [9] Globus Toolkit 2.4, Globus Resource Allocation Manager (GRAM): <http://www.globus.org/toolkit/docs/2.4/gram/>
- [10] H.J. La, J. S. Bae, S.H. Chang, S. D. Kim, "Practical Methods for Adapting Services Using Enterprise Service Bus," ICWE 2007, LNCS 4607 (L. Baresi, P. Fraternali, G.J. Houben, eds.), pp. 53-58, 2007
- [11] Y. Maurel, A. Diaconescu, P. Lalanda, "CEYLON: A Service-Oriented Framework for Building Autonomic Managers," in *Proceedings of the 2010 Seventh IEEE International Conference and Workshops on Engineering of Autonomic and Autonomous Systems (EASE '10)*. IEEE Computer Society, 2010, pp 3-11.
- [12] M. Papazoglou, P. Traverso, S. Dustar, F. Leymann, "Service Oriented Computing: State of the Art and Research Challenges," *IEEE Computer*, Vol. 40 (2007), Issue 11, p. 38
- [13] Leymann, F., "The (Service) Bus: Service Penetrate Everyday Life," 3<sup>rd</sup> Intl. Conf. on Service Oriented Computing ISCOC'05, Amsterdam, the Netherlands, Dec. 13-16, 2005, LNCS 3826 Springer-Verlag Berlin Heidelberg 2005
- [14] D. Chappel, "Enterprise Service Bus: Theory and Practice," O'Reilly Media, 2004
- [15] F. Leymann, "Combining Web Services and the Grid: Towards Adaptive Enterprise Applications," Proc. CAISE/ASMEA'05, Porto, Portugal, June 2005
- [16] Java Community Process, JSR 208 "Java Business Integration," <http://jcp.org/aboutJava/communityprocess/final/jsr208/index.html>
- [17] B.A. Christudas, "Service Oriented Java Business Integration: Enterprise Service Bus integration solutions for Java Developers," Packt Publishing, 2008.
- [18] P. Martinez-Julia, D.R. Lopez, and A.F. Gomez-Skarmeta. 2010. The GEMBus Framework and Its Autonomic Computing Services. In *Proceedings of the 2010 10th IEEE/PSJ International Symposium on Applications and the Internet (SAINT '10)*. IEEE Computer Society, Washington, DC, USA, pp. 285-288
- [19] L. Gonzalez and R. Ruggia. 2010. Towards dynamic adaptation within an ESB-based service infrastructure layer. In *Proceedings of the 3rd International Workshop on Monitoring, Adaptation and Beyond (MONA '10)*. ACM, New York, NY, USA, pp. 40-47.
- [20] X. Mi, X. Tang, X. Yuan, D. Chen, and X. Luo. 2009. Multifactor-Driven Hierarchical Routing on Enterprise Service Bus. In *Proceedings of the International Conference on Web Information Systems and Mining (WISM '09)*, Wenyin Liu, Xiangfeng Luo, Fu Lee Wang, and Jingsheng Lei (Eds.). Springer-Verlag, Berlin, Heidelberg, pp. 328-336.
- [21] X. Bai, J.Xie, B. Chen, and S. Xiao. 2007. DRESR: Dynamic Routing in Enterprise Service Bus. In *Proceedings of the IEEE International Conference on e-Business Engineering (ICEBE '07)*, Hong Kong, pp. 528-531.
- [22] G. Hohpe, B. Woolf, "Enterprise Integration Patterns," Addison-Wesley, 2004.
- [23] Apache ServiceMix, <http://servicemix.apache.org/home.html>
- [24] Business Process Execution Language (BPEL), <http://www.ibm.com/developerworks/library/specification/ws-bpel/>

# An Analysis of mOSAIC ontology for Cloud Resources annotation

Francesco Moscato

Second University of Naples

Dep of European and Mediterranean Studies  
Caserta, Italy

Email: francesco.moscato@unina2.it

Rocco Aversa

Second University of Naples

Dep of Information Engineering  
Aversa, Italy

Email: rocco.aversa@unina2.it

Beniamino Di Martino

Second University of Naples

Dep of Information Engineering  
Aversa, Italy

Email: beniamino.dimartino@unina.it

Teodor-Florin Fortiș

Institute e-Austria, Timișoara, Romania, and

West University of Timișoara, Romania

Email: fortis@info.uvt.ro

Victor Munteanu

Institute e-Austria

Timișoara, Romania

Email: vmunteanu@info.uvt.ro

**Abstract**—The easiness of managing and configuring resources and the low cost needed for setup and maintaining Cloud services have made Cloud Computing widespread. Several commercial vendors now offer solutions based on Cloud architectures. More and more providers offer new different services every month, following their customers needs. Anyway, it is very hard to find a single provider which offers all services needed by end users. Furthermore, different vendors propose different architectures for their Cloud systems and usually these are not compatible. Very few efforts have been done in order to propose a unified standard for Cloud Computing. This is a problem, since different Cloud systems and vendors have different ways to describe and invoke their services, to specify requirements and to communicate. Hence a way to provide a common access to Cloud services and to discover and use required services in Cloud federations is appealing. mOSAIC project addresses these problems by defining a common ontology and it aims at developing an open-source platform that enables applications to negotiate Cloud services as requested by users. The main problem in defining the mOSAIC ontology is in the heterogeneity of terms used by Clouds vendors, and in the number of standards which refer to Cloud Systems with different terminology. In this work the mOSAIC Cloud Ontology is described. It has been built by analysing Cloud standards and proposals. The Ontology has been then refined by introducing individuals from real Cloud systems.

## I. INTRODUCTION

CLOUD Computing is an emerging model for distributed systems. It refers both to applications delivered as services and to hardware, middleware and other software systems needed to provide them. Nowadays the Cloud is drawing the attention from the Information and Communication Technology (ICT) thanks to the appearance of a set of services with common characteristics which are provided by industrial vendors. Even if Cloud is a new concept, it is based upon several technologies and models which are not new and are built upon decades of research in virtualization, service oriented architecture, grid computing, utility computing or distributed computing ([26], [34], [42]). The variety of technologies and architectures makes the Cloud overall picture confusing [26]. Cloud service providers make resources accessible from

Internet to users presenting them *as a service*. The computing resources (like processing units or data storages) are provided through virtualization. *Ad-hoc* systems can be built based on users requests and presented as services (*Infrastructure as a Service*, IaaS). An additional abstraction level is offered for supplying software platforms on virtualized infrastructure (*Platform as a Service*, PaaS). Finally software services can be executed on distributed platforms of the previous level (*Software as a Service*, SaaS). Except from these concepts, several definitions of Cloud Computing exist ([41], [19], [27], [24], [37], [33]), but each definition focuses only on particular aspects of the technology. Cloud computing can play a significant role in a variety of areas including innovations, virtual worlds, e-business, social networks, or search engines but it is actually still in its early stages, with consistent experimentation to come and standardization actions to effort. In this scenario, vendors provide different Cloud services at different levels usually providing their own interfaces to users and Application Programming Interfaces (APIs) to developers. This results in several problems for end-users that perform different operations for requesting Cloud services provided by different vendors, using different interfaces, languages and APIs. Since it is usually difficult to find providers which fully address all users needs, interoperability among services of different vendors is appealing.

Cloud computing solutions are currently used in settings where they have been developed without addressing a common programming model, open standard interfaces or adequate service level agreements or portability of applications. Neglecting these issues current Cloud computing forces people to be stranded into locked, proprietary systems. Developers making an effort in Cloudifying their applications cannot port them elsewhere.

In this scenario the mOSAIC project (EU FP7-ICT programme, project under grant #256910) aims at improving state of the art in Cloud computing by creating, promoting and exploiting an open-source Cloud application programming

interface and a platform targeted for developing multi-Cloud oriented applications. One of the main goal is that of obtaining transparent and simple access to heterogeneous Cloud computing resources and to avoid locked-in proprietary solutions.

In order to attain this objective a common interface for users has to be designed and implemented, which should be able to wrap existing services, and also to enable intelligent service discovery. The keystone to fulfil this goal in mOSAIC is the definition of an ontology able to describe services and their (wrapped) interfaces.

Ontologies offer the means of explicit representation of the meaning of different terms or concepts, together with their relationships. They are directed to represent semantic information, instead of content. Different languages can be considered for the specification of ontologies, including DAML, OIL, RDF and RDFS, OWL or WSML.

The Web Ontology Language (OWL) is a standard from [32], [18], based on XML, RDF and RDFS. With OWL complex relationships and constraints can be represented in ontologies. With important revisions to the language, OWL 2 became the W3C recommendation in 2009, introducing features to improve scalability in applications. [25]

Different efforts to formalize Semantic Web developments exist. Web Service Modeling Ontology (WSMO) [14] “provides the conceptual underpinning and a formal language for semantically describing all relevant aspects of Web services in order to facilitate the automatization of discovering, combining and invoking electronic services over the Web” [40]. WSML was offered as a companion language to WSMO, for representing modelled ontologies by a common terminology for Web Services interactions [22], [40]. The Semantic Web Services Framework (SWSF) offers a similar approach, with its two major components, the Semantic Web Services Language (SWSL) and the Semantic Web Services Ontology (SWSO) [17].

Semantically-enabled services offer the means for intelligent selection of services, with automation of different tasks, including service discovery, mediation, invocation, or composition. Current research efforts are enhancing typical web services technologies in order to provide a semantically-enhanced behaviour in developments like OWL-S [30], WSDL-S and METEOR-S [38], [36], WSML [22], WSMO [40], or SWSF [17].

## II. MOSAIC PROJECT

The Open Cloud Manifesto [10] identifies five main challenges for Cloud: data and application interoperability; data and application portability; governance and management; metering and monitoring; security.

Actually, the main problem in Cloud computing is the lack of unified standards. Market needs drive commercial vendors to offer Cloud services with their own interfaces since no standards were available at the moment. Vendors solutions have arisen as commonly used interface for Cloud services but interoperability remains an hard challenge, like portability of developed services on different platforms. In addition vendors

and open Cloud initiatives spent few efforts in offering services with negotiated quality level.

The mOSAIC project tries to fully address the first two challenges and partially addresses the next two ones by providing a platform which:

- enables interoperability among different Cloud services,
- eases the portability of developed services on different platforms,
- enables intelligent discovery of services,
- enables services composition,
- allows for management of Service Levels Agreement (SLA).

The architecture of mOSAIC platform is depicted in Fig.1: it provides facilities both for end-users (at the left of Fig.1) and for services developers and managers (depicted on the right side of Fig.1)

From the end-users’ point of view, the main component is the *Cloud Agency*. This consists in a core set of software agents which implement the basic services of this component. They include:

- negotiation of SLAs;
- deployment of Cloud services;
- discovery and brokering of Cloud services.

In particular, *Client Agent* is responsible for collecting users’ application requirements, for creating and updating the SLAs in order to grant always to best QoS. The *Negotiator* manages SLAs and mediates between the user and the broker; it selects protocols for agreements, negotiates SLA creation, and it handles fulfilment and violation. The *Mediator* selects vendor agents able to deploy services with the specified user requirements; it also interfaces with services deployed on different vendors’ providers. The *Provider Agent* interacts with virtual or physical resources at provider side. In mOSAIC the Cloud Agency was built upon the MAGDA [16] toolset, which provides all the facilities to design, develop and deploy agent-based services. The *semantic engine* uses information in the Cloud Ontology to implement a semantic-based Cloud services discovery exploiting semantic, syntactic and structural schema matching for searches.

In the Cloud developers and managers perspective, the main components of mOSAIC Architecture are the *API execution engine* and the *Resource Manager*. The first one offers a unique API to use Cloud Services from different vendors when using and developing other services. The API execution engine is able to wrap storage, communication and monitoring features of Cloud platforms. In particular, Virtual Clusters (VC) [23] are used as resource management facility. They are configured by software agents in order to let users to configure required services. A *Resource contract* will grant user’s requirements and the Resource Manager will assign physical resources to VC on the basis of the contract.

In this architecture, the bonding element which allows for interoperability and resources description is the *Cloud Ontology*. It is the base for Cloud services and resources description and it contains all information needed to characterize API also from a semantic point of view.



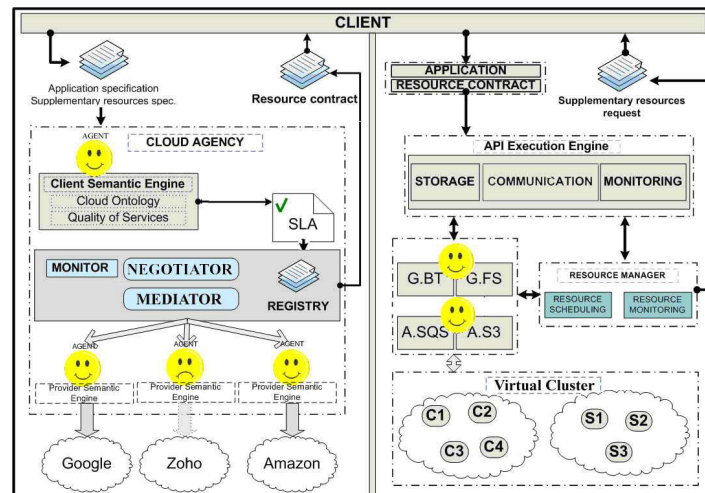


Fig. 1: mOSAIC Architecture

The Cloud Ontology is based on several Cloud taxonomies proposed in literature [15], [4], [28], [20], [29]. It is developed in OWL [32] and OWL-S languages [30]. The benefit of using an ontology language is that it acts as a general method for the conceptual description or modelling of information that is implemented by actual resources [3]. mOSAIC aims at developing ontologies that would offer the main building block to describe services at the three delivery models of Cloud Computing (i.e. IaaS, PaaS, SaaS).

### III. CLOUD STANDARDS AND MOSAIC

Nowadays several Cloud computing systems are available, both from commercial and open source communities. Some example are Amazon EC2 [1], Google's App Engine [6], Microsoft Azure [21], GoGrid [5], 3Tera [7], Open Nebula [12], Eucalyptus [35] and Nimbus [2]. Cloud systems and services offered by various vendors differ and overlap in complicated ways. Each solution provides different services. For example Amazon EC2, GoGrid, 3Tera, Open Nebula and Eucalyptus are basically IaaS Clouds offering entire instances of virtual machines to customers; Google's Apps and Microsoft Azure offer SaaS applications also providing API for development and monitoring, offering a PaaS Cloud. Nimbus was developed as an IaaS for scientific applications. Main vendors platforms and services have become standards *de facto* for Cloud computing, but several different solutions exist at different Cloud layers and interoperability is still a distant goal. In this scenario some attempts have been done to make order in the chaos of Cloud systems trying to propose standards for them.

Anyway the need for a good, complete definition of Cloud components is really felt by scientific community. In particular, in [43] the need for an Ontology defining Cloud-related concepts and relationships is outlined. In the paper an ontology for Cloud is proposed *in natural language*. Cloud layers have been defined and organized in an architectural view. The ontology starts with firmware and hardware as its foundation,

eventually delivering to Cloud applications. The paper also defines elements which belongs to different layers, like resources, virtual machines etc. Anyway no formal representation of the Ontology is reported in the paper.

Another similar taxonomy for Cloud Systems has been presented in [39], where only Cloud layers and some requirements like fault tolerance and security have been discussed.

A more detailed Taxonomy has been described in [20]. It is a simple taxonomy, with only main concepts related to Cloud Computing defined in a graphical schema. This work in progress is anyway more complete than the previous ones.

One of the few attempts to provide a formal ontology for Cloud System comes from Unified Cloud Interface (UCI) Initiative that have released a very simple OWL ontology [13] for Cloud Systems but it consist only of few concepts.

Cloud Computing Interoperability Forum (CCIF) [9] aims at defining an open, vendor neutral and standardized Cloud interface for the unification of various Cloud APIs. This should be done creating an API wrapping other existent APIs. CCIF proposes to define an OWL/RDF ontology to describe a semantic Cloud data model in order to address Cloud resources uniquely. The ontology is still under development and no draft version are available at the moment.

Similarly, Open Grid Forum (OGF) [11] is another open initiative which aims at the creation of a practical solution to interface existing Cloud IaaS. OGF is defining interfaces (the Open Cloud Computing Interface: OCCI [11]) to provide unified access to existing IaaS resources. The main goal of OCCI is the creation of hybrid Clouds operating environments independent from vendors and middlewares. The main formalism used to define Cloud models is UML and the work is still in a preliminary stage. OCCI's documents defines specifications for cloud core elements and interfaces to resources, including a model using a RESTful API to access, use and manage them. In OCCI core model, *everything is a resource*. The main elements of the base OCCI model are: Entity, Resource, Link and Action. The Entity is an abstract type for

Resource and Link type; Resources identify object in cloud environment, while Link are used to specify relationships among Resources. Actions define operations applicable to Entities. The OCCI model is developed in UML, but the main elements of the model are used to describe a graph structure that is similar to an OWL ontology definition. The model is not yet complete and several properties and relationships between Entities cannot be described.

The National Institute of Standards and Technology (NIST) is also working at Cloud standards definitions. NIST main aim in defining cloud standards is to provide specifications about interoperability, portability and security requirements, standards and guidance. As part of the NIST plan, a reference Architecture and Taxonomy standards have been proposed. In addition, a roadmap to address security in Cloud systems has been formalized. The NIST definition of cloud computing architecture, includes basically five essential characteristics (on-demand self-service, broad network access, resource pooling, rapid elasticity, measured Service), three service models (IaaS, PaaS, SaaS) and four deployment models (public, private, community and hybrid cloud). Furthermore NIST proposes a taxonomy, organized in layers, where main cloud concepts are organized. In the first level roles for cloud are identified: service provider, consumer, broker, auditors and carriers are listed. They are related to usage scenarios analysed by NIST that focuses on interoperability of federating cloud providers. At second level, activities that actors in the first level enact are defined, while components are addressed in the last two levels. The mOSAIC Ontology, which will be described in detail in Section IV inherits most of the elements defined in other proposals, in order to maintain a high degree of compatibility with them. Anyway, in the NIST taxonomy, too much Roles (Actors in mOSAIC) have been reported and some of them are not easily distinguishable. In addition The taxonomy proposed by NIST is not really a hierarchy since elements in lower layers are not always specialization of upper layer elements. For this reason, mOSAIC inherits only the main actors definition from NIST proposal, and does not maintain the taxonomy proposed by NIST when no *subclass* relationship exists between classes.

Distributed Management Task Force (DTMF) proposes a standards incubators in order define a set of architectural semantics that unify the interoperable management of enterprise and cloud computing. DTMF mainly addresses use cases and cloud reference architecture, analysing the interfaces between cloud service providers and consumers. In use case definition, a key role has the definition of Service Level Agreement (SLA). The interactions between services consumers and providers are detailed. Actors in DTMS proposals are similar to Roles for NIST definition. DTMS also addresses interfaces and data artefacts as a mean for interfacing actors.

Recently IBM [31] has provided a draft document describing a reference architecture for Cloud systems. It recalls the NIST and OCCI standards, integrating some parts with new elements. In particular, the IBM architecture adds Business processes as element of Cloud Architecture. Business Process

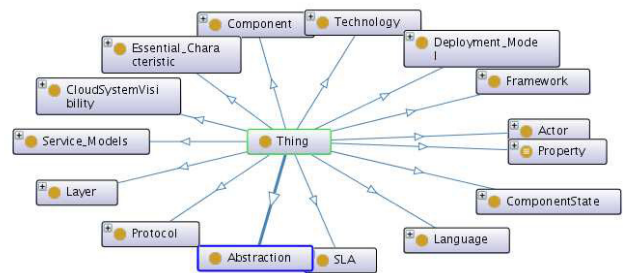


Fig. 2: Top Level Concepts in mOSAIC Ontology

has been introduced as Cloud architecture layer and as new class of actors that can use them in order to create composed services. Proper Business Support Services (BSS) have been defined in order to support business process definition and execution. Furthermore, IBM architecture inherits NIST actors taxonomy, maintaining three main types of actors: services Consumer, Provider and Creator. It also addresses the problem of defining Quality of Services and SLA.

Microsoft Azure [21] and Google APP Engine [6] provide an environment for developing and deploying services and applications following the Cloud philosophy. Azure offers a Platform as a Service (PaaS) and an Infrastructure as a Service (IaaS) hybrid platform on which developers implements and deploy their services. A similar approach is adopted by Google APP Engine. They provide almost no attempt at standardization, except for a small OWL ontology (no more supported) developed by Google. Amazon with its Amazon Web Services (AWS) offers IaaS and SaaS services. It provides, a set of API for creating virtual machines and resources, and to access its services. mOSAIC ontology is able to describe these APIs as will be shown in Section IV-A

#### IV. MOSAIC ONTOLOGY

The top level of the mOSAIC Ontology is shown in Fig.2 which reports the main concepts of the mOSAIC ontology. Concepts have been identified analysing standards and proposals from literature. In the following its main concepts will be listed and described.

The **Language** class contains instances of languages used for APIs implementation (for example, Java and Python). **Abstraction** class contains the abstraction level at which services are provided as described in[43]. Here, Cloud services belong to the same layer if they have equivalent level of abstraction. **Deployment Model** class includes concepts required by Cloud NIST [43] standard for what deployment model of Cloud services concerns. **Essential Characteristics** class includes individuals which are defined by NIST. **Framework** class contains individuals that identify programming framework supporting API programming Languages. **Actor** contains subclasses where actors interacting with Cloud systems are divided. **Property** subclasses contain all elements needed for describing characteristics of Cloud resources. These are also



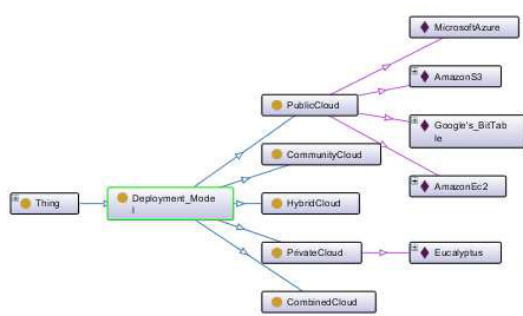


Fig. 3: Deployment Model

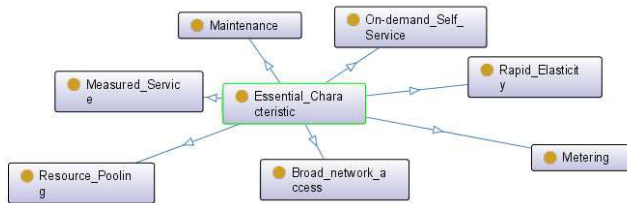


Fig. 4: Essential Characteristics

used to specify SLA requirements. **ComponentState** includes all concepts for defining the states which Cloud components and resources may assume. **SLA** class defines concepts for SLA definitions. **Protocol** class contains individuals for protocols used in communication among Cloud components. **Layers** class distinguishes firmware, hardware and software infrastructures for Cloud platforms. **Service Models** class includes all kinds of services provided by Cloud Systems. **Predicate** contains classes used for description of the behaviours of statefull Cloud components. **CloudSystemVisibility** class allows for specification of Cloud systems visibility, like private and public clouds. **Component** is the main class of mOSAIC ontology. All cloud elements (resources, services, infrastructures etc.) are its subclasses. **Technology** class contains all concepts related to technology involved in Cloud services provisioning, like virtualization.

Fig.3 shows the *Deployment\_Model* subclasses.

They include several types of deployment models for Cloud Systems: **PublicCloud** contains all individuals providing public or world wide access to their resources, like MicrosoftAzure, Amazon and Google. **PrivateCloud** instead is related to Deployment Models of framework that can provide access to private Cloud resources, like Eucalyptus.

Essential\_Characteristics are defined in the NIST standards as the features that each Cloud system must provide. They are shown in Fig. 4.

This Class contains subclasses related to the characteristics described into NIST document about Cloud features (Maintenance, On demand self service, Rapid Elasticity, Metering, Broad Network Access, Measured Services and Resource

pooling). For further description of this properties, refer to NIST [8] and IBM [31] standards documents.

The Actor class identifies cloud actors, that can be divided as in Fig.5.

Provider, Consumer and Creator subclasses follow the IBM cloud computing reference Architecture [31]. Administrator manages cloud infrastructure; Orchestrator composes Cloud Services in order to provide value added services; Developer implements new Cloud Services. Notice that the difference between Developer and Creator, is in the way they interact with cloud Providers. Developers use offline resources (tools and frameworks) in order to implement new Cloud Service. A Creator instead builds cloud services by using functionalities exposed by a Cloud Service Provider. Consumer Actors can be further divided as shown in Fig. 5b.

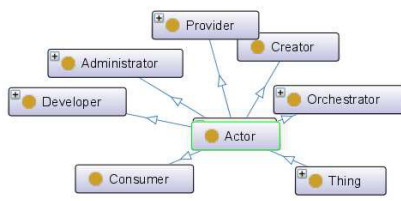
Some Property's subclasses are shown in Fig.6. They are divided into NonFunctionalProperties and FunctionalProperties that respectively define the sets of non functional and functional properties of a Cloud Component. Properties can be used to characterize Cloud Components (services, infrastructure etc.) and to request given characteristics for components when dealing with SLA.

The main non-functional properties for cloud components are: Scalability; Autonomy; Availability; QoS; Performance; Consistency; Security; Reliability.

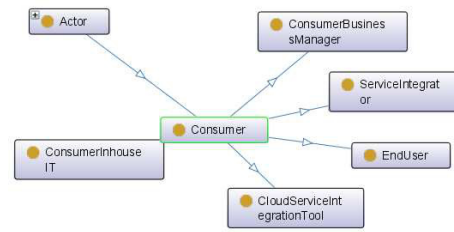
Computing Non Functional properties can be divided into CPU and Memory related properties. A deeper division identifies: CPUSpeedProperty; CPUNumberOfCores; CPUArchitecture; CPUTypeProperty and CPUFlopsProperty. These properties are used to specify the clock frequency, the number of cores, the architecture, the model and the FLOPS of CPUs respectively. The properties follow the OCCI [11] standard and API. A Data Property is defined for each of them in order to specify the value of the property for the related individuals. Properties for memory are divided into: MemoryAllocationProperty and MemorySize. The first property is used to specify memory allocation policies while the second one is used to declare (or require) the amount of memory in a Cloud infrastructure.

Subclasses of this Network Non Functional element are: NetworkLatencyProperty, NetworkDelayProperty, Network-BandwidthProperty. The first class is used to define the mean latency of a network, the second one the mean, the maximum and the minimum delay for packets and the last one is used to define the mean and the maximum bandwidth of a network. The values for individuals are defined by specifying proper data properties defined on these classes. Data non functional properties are related to disk size (DiskSpaceProperty), transfer rate (DiskTransferRateProperty) and bandwidth (DiskBandwidthProperty).

The main functional properties are: Replication (for the definition of the type of replication policies of resources); Encryption (it specifies the encryption policies of resources); BackupAndRecovery (it is used to describe the back up and recovery strategies used for a Cloud Component); Accounting (its individuals define the accounting policies for

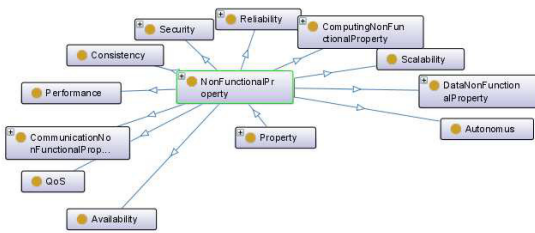


(a) Actor

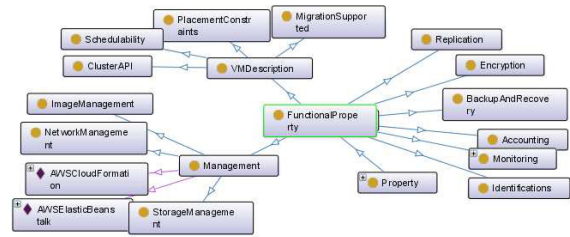


(b) Consumer

Fig. 5: Actors



(a) Non Functional Properties



(b) Functional Properties

Fig. 6: Properties

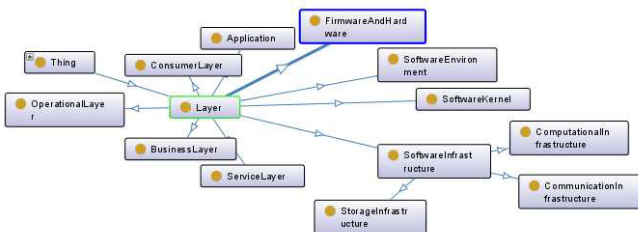


Fig. 7: Layer

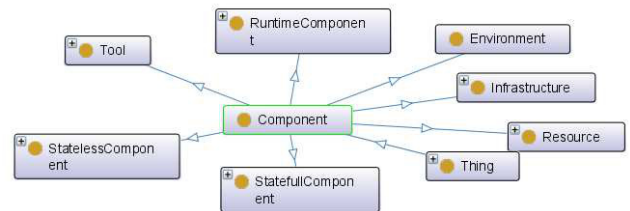


Fig. 8: Component

resources); Monitoring (this class allows for the specification of monitoring policies for resources); Identification (it contains individuals that can specify the algorithms and policies for users identification); VMDescription (used to describe virtual machines technologies and configuration eventually used in cloud infrastructure); Management (it defines the management policies for cloud resources).

Management contains the following subclasses: ImageManagement, NetworkManagement and StorageManagement. The first one is used to define the management policies of a VM image, the second one to define network management policies in a cloud infrastructure, while the third one is used to define storage management policies for cloud resources.

Layers are organized as in Fig.7.

The classes OperationalLayer, ServiceLayer, BusinessLayer, and ConsumerLayer are defined in the IBM Cloud Computing Reference Architecture. They respectively represent: the operational infrastructure layer of cloud systems; the services

layer in clouds; the business processes that participate in Cloud solutions; the layer with cloud services and resources consumers.

Furthermore, other layers are derived from OCCI and NIST definitions. They are: Application (the layer where applications lie); Firmware and Hardware (the layers with hardware and firmware of cloud infrastructures); Software kernel (the software kernel for operating systems and middlewares used in cloud infrastructures); Software infrastructure (the layer of computational, storage and communication infrastructures).

Service\_Models subclasses includes all models for services in Cloud. Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Service as a Service (SaaS) are the classical models defined in the NIST standard, while the last, BPaaS (Business Process as a Service) is defined in the IBM Cloud Computing Reference Architecture.

Component Subclasses are reported in Fig.8.

They are divided into: Tool (this contains all tools used for

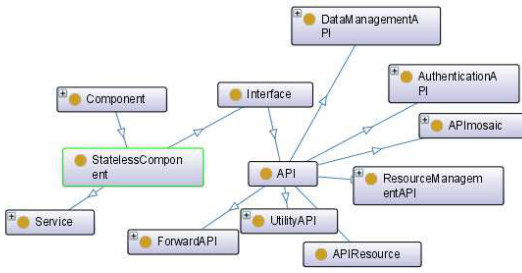


Fig. 9: Stateless Component

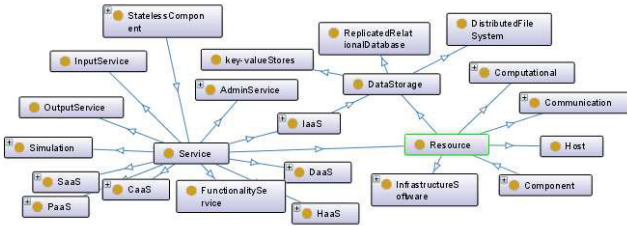


Fig. 10: Resource and Services

Cloud services development or cloud resources management); *RunTimeComponent* (this class contains the elements for defining *mOSAIC* run-time components); *Environment* (used to define individuals concerning the cloud environment used by cloud services); *Infrastructure* (describes the component in the cloud infrastructure); *StatefullComponent* and *StatelessComponent*; *Resource* (it collects all resource classes in the cloud ontology).

*StatelessComponent* class is expanded in Fig.9. Basically stateless component in the ontology are *Services* and *Interfaces*. *Services* will be described later and a general taxonomy for *APIs* is reported in the figure.

Stateful component are omitted for brevity.

The *Resource* class is the most complex in the *mOSAIC*, since, following the *OCCI* documentation, in *Cloud Systems* everything is a cloud *Resource*. Hence this is a super class for other main cloud components as shown in Fig.10.

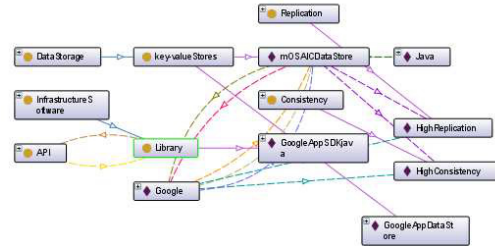
*Service* is a *Resource*. *Platform as a Service (PaaS)*, *Computing as a Service (CaaS)*, *Data as a Service (DaaS)*, *infrastructure as a Service (IaaS)*, *Hardware as a Service (HaaS)* and other service models (*Simulation*, services which offers some functionalities, admin, data input and output services) are subclasses of *Services*. For example, in Figure, *DataStorage* is provided as an *IaaS*. *Key-valueStores*, *ReplicatedRelationalDatabases* and *DistributedFileSystems* are examples of *DataStorage* services. *Cloud Component* are also considered as *Cloud Resources*, like *Hosts*, *Computational* and *Communication resources* or *InfrastructureSoftware*.

#### A. *DataStorage Example*

In this section a brief example is shown where the *mOSAIC* ontology is used to describe a simple data-storage service implemented with *Google App Engine*.

TABLE I: ObjectProperties for Individuals

Domain	ObjectProperty	Range
<i>mOSAICDataStore</i>	<i>developedWithLanguage</i>	JAVA
<i>mOSAICDataStore</i>	<i>fulfills</i>	HighReplication
<i>mOSAICDataStore</i>	<i>fulfills</i>	HighConsistency
<i>mOSAICDataStore</i>	<i>hasServiceProvidedBy</i>	Google
<i>mOSAICDataStore</i>	<i>isOfferedByProvider</i>	Google
<i>mOSAICDataStore</i>	<i>isOwnedBy</i>	Google
Google	<i>guarantee</i>	HighReplication
Google	<i>guarantee</i>	HighConsistency
Google	<i>guarantee</i>	LowConsistency
Google	<i>offersAPI</i>	GoogleAppAPI
Google	<i>own</i>	<i>mOSAICDataStore</i>
GoogleAppAPI	<i>developedWithLanguage</i>	JAVA
GoogleAppAPI	<i>developedWithLanguage</i>	Python
GoogleAppAPI	<i>developedWithLanguage</i>	Go

Fig. 11: *mOSAICDataStore* main Individuals and Relationships

The name of the realized service is *mOSAICDataStore* and it implements a key-based data store by using the *JAVA Google APP SDK*. As described in the *Google App* documentation, *Google Data Stores* offer *Replication* functionalities. In particular, they allow for the use of *Master-Slave* and *High Replication* techniques. In addition *Consistency* on replicas is assured in most cases, although full consistency is not assured. *Google* offers a framework for development of cloud applications, which is based on an *Eclipse Plug-in*. The *mOSAICDataStore* was developed by using this plug-in.

First of all, *Google* has been defined as individual of *InfrastructureProvider*, *DeveloperProvider*, *ServiceProvider* and *ResourceProvider*; *GooglePluginForEclipse* has been added as a *Framework's* individual. *GoogleApps* has been included as *PaaS* and *SaaS* service Model; *GoogleAppAPI*, an individual representing the *Google APP API* has been inserted into *API*, and *GoogleAPPSDK* as *Library*.

Table I reports some of the objectProperties defined on individuals.

In Fig.11 some of the individuals and the relationships reported in Tab. I are depicted.

## V. CONCLUSIONS

In this work we propose a detailed ontology for *Cloud systems* that can be used to improve interoperability among existing *Cloud Solutions*, platforms and services, both from end-user and developer side. The ontology has been developed in *OWL* and can be used for semantic retrieval and composition of *Cloud services* in the *mOSAIC* project. Several attempts have been done in the past to introduce a *Cloud ontology*. The ontology presented in this paper also maintains compatibility with previous works because it is built upon

existing standards and proposals analysis and it results in a more comprehensive description of all Cloud-related aspects. The Ontology has been populated with individuals from real Cloud Systems services and APIs and new individuals and elements are going to be included in the ontology with an incremental design approach.

#### ACKNOWLEDGMENT

This work has been supported by the mOSAIC project (EU FP7-ICT programme, project under grant #256910). We want to thank Daniel Bove for his role in developing the ontology.

#### REFERENCES

- [1] Amazon Elastic Compute Cloud (Amazon EC2): <http://aws.amazon.com/ec2/>.
- [2] Cloud Computing Interoperability Forum: <http://www.cloudforum.org>.
- [3] Cloud Computing Interoperability Forum, Unified Cloud Computing: <http://code.google.com/p/unifiedcloud/>.
- [4] Cloud Computing Interoperability Forum, Cloud taxonomy : <http://groups.google.com/group/cloudforum/web/ccif-cloud-taxonomy>.
- [5] GoGrid: Scalable Load-Balanced Windows and Linux Cloud-Server Hosting: <http://www.gogrid.com/>.
- [6] Google App Engine: <http://code.google.com/appengine/>.
- [7] Grid Computer Operating System For Web Applications—AppLogic, 3tera.: <http://www.3tera.com/AppLogic/>.
- [8] National Institute of Standards and Technology (NIST), Cloud Standards: <http://csrc.nist.gov/groups/SNS/cloud-computing/>.
- [9] Nimbus Science Cloud: <http://workspace.globus.org/clouds/nimbus.html>.
- [10] Open Cloud Manifesto, Spring 2009: <http://www.opencloudmanifesto.org>.
- [11] Open Grid Forum: Open Cloud Computing Interface (OCCI): <http://forge.ogf.org/sf/projects/occi-wg>.
- [12] OpenNebula, the Open Source Toolkit for Cloud Computing: <http://www.opennebula.org>.
- [13] UCI Cloud OWL Ontology: <http://code.google.com/p/unifiedcloud/source/browse/trunk/ontologies/cloud.owl?r=14>.
- [14] Web Service Modelling Ontology (WSMO): <http://www.wsmo.org>.
- [15] Appistry. Cloud Taxonomy: Applications, Platform, Infrastructure: <http://www.appistry.com/blogs/sam/cloud-taxonomy/-applications-platform-infrastructure>, 2008.
- [16] R. Aversa, B. Di Martino, N. Mazzoeca, and S. Venticinque. A skeleton based programming paradigm for mobile multi-agents on distributed systems and its realization within the magda mobile agents platform. *Mob. Inf. Syst.*, 4:131–146, April 2008.
- [17] Steve Battle, Abraham Bernstein, Harold Boley, Benjamin Grosf, Michael Gruninger, Richard Hull, Michael Kifer, David Martin, Sheila McIlraith, Deborah McGuinness, Jianwen Su, and Said Tabet. Semantic web services framework, September 2005.
- [18] Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn A. Stein. Owl web ontology language reference. Technical report, W3C, 2004.
- [19] Rajkumar Buyya, Chee S. Yeo, and Srikumar Venugopal. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. In *HPCC '08: Proceedings of the 2008 10th IEEE International Conference on High Performance Computing and Communications*, pages 5–13. IEEE Computer Society, September 2008.
- [20] C. Hoff. Cloud Taxonomy and Ontology: <http://rationalsecurity.typepad.com/blog/2009/01/cloud-computing-taxonomy-ontology.html>, 2009.
- [21] David Chappell. Introducing the Azure Services Platform. David Chappel and Associates Whitepaper (Sponsored by Microsoft Corporation): <http://www.davidchappell.com/blog/2008/10/introducing-azure-services-platform.html>, 2008.
- [22] Jos De Bruijn, Holger Lausen, Reto Kruppenacher, Axel Polleres, Livia Predoiu, Michael Kifer, and Dieter Fensel. The web service modeling language wsml. Technical report, DERI, October 2005.
- [23] U. Villano E. P. Mancini, M. Rak. PerfCloud: GRID Services for Performance-oriented Development of Cloud Computing Applications. In *Proceedings of WETICE*. IEEE Computer Society, July 2009.
- [24] Galen Gruman and Eric Knorr. What cloud computing really means. InfoWorld : <http://www.infoworld.com/article/08/04/07/15FE-cloud-computing-reality1.html>, 2008.
- [25] Bernardo Cuenca Grau, Ian Horrocks, Boris Motik, Bijan Parsia, Peter Patel-Schneider, and Ulrike Sattler. Owl 2: The next step for owl. *Web Semant.*, 6:309–322, November 2008.
- [26] Kai Hwang. Massively distributed systems: From grids and p2p to clouds. In *Proceedings of The 3rd International Conference on Grid and Pervasive Computing—gpc-workshops*, page xxii, 2008.
- [27] Jeremy Geelan. Twenty one experts define cloud computing. Virtualization : <http://virtualization.sys-con.com/node/612375>, 2008.
- [28] P. Lairds. Cloud Computing Taxonomy. In *Procs. Interop09*, pages 201–206. IEEE Computer Society, May 2009.
- [29] Alexander Lenk, Markus Klems, Jens Nimis, Stefan Tai, and Thomas Sandholm. What's inside the cloud? an architectural map of the cloud landscape. In *Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing*, CLOUD '09, pages 23–31, Washington, DC, USA, 2009. IEEE Computer Society.
- [30] David Martin, Massimo Paolucci, Sheila McIlraith, Mark Burstein, Drew McDermott, Deborah McGuinness, Bijan Parsia, Terry Payne, Marta Sabou, Monika Solanki, Naveen Srinivasan, and Katia Sycara. Bringing Semantics to Web Services: The OWL-S Approach. In J. Cardoso and A. Sheth, editors, *SWSWPC 2004*, volume 3387 of *LNCS*, pages 26–42. Springer, 2004.
- [31] P.Kopp R.Dieckmann G.Breiter S.Pappe H.Kreger A. Arsanjani M.Behrendt, B. Glasner. Introduction and Architecture Overview, IBM Cloud Computing Reference Architecture 2.0: [https://www.opengroup.org/cloudcomputing/uploads/40/23840/CCRA\\_IBMSubmission.02282011.doc](https://www.opengroup.org/cloudcomputing/uploads/40/23840/CCRA_IBMSubmission.02282011.doc), 2011.
- [32] McGuinness, D. L., van Harmelen, F. OWL Web Ontology Language Overview. W3C Recommendation: <http://www.w3.org/TR/2004/REC-owl-features-20040210/>, 2004.
- [33] Members of EGEE-II. An egee comparative study: Grids and clouds - evolution or revolution. Technical report, Enabling Grids for E-science Project : <https://edms.cern.ch/document/925013/>, 2008.
- [34] Dejan Milojicic. Cloud computing: Interview with russ daniels and franco travostino. *IEEE Internet Computing*, (5):7–9, 2008.
- [35] Daniel Nurmi, Rich Wolski, Chris Grzegorzczak, Graziano Obertelli, Sunil Soman, Lamia Youseff, and Dmitrii Zagorodnov. The eucalyptus open-source cloud-computing system. In *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, CCGRID '09, pages 124–131, Washington, DC, USA, 2009. IEEE Computer Society.
- [36] Abhijit A. Patil, Swapna A. Oundhakar, Amit P. Sheth, and Kunal Verma. Meteor-s web service annotation framework. In *Proceedings of the 13th international conference on World Wide Web*, WWW '04, pages 553–562, New York, NY, USA, 2004. ACM.
- [37] Paul McFedries. The cloud is the computer. *IEEE Spectrum Online*: <http://www.spectrum.ieee.org/aug08/6490>, 2008.
- [38] R. Akkiraju, J. Farrell, J. Miller, M. Nagarajan, M. Schmidt, A. Sheth, K. Verma. Web Service Semantics WSDL-S. A joint UGA-IBM Technical Note, version 1.0: <http://lstdis.cs.uga.edu/projects/METEOR-S/WSDL>, 2005.
- [39] Bhaskar Prasad Rimal, Eunmi Choi, and Ian Lumb. A taxonomy and survey of cloud computing systems. *Networked Computing and Advanced Information Management, International Conference on*, 0:44–51, 2009.
- [40] Dumitru Roman, Uwe Keller, Holger Lausen, Jos de Bruijn, Rubn Lara, Michael Stollberg, Axel Polleres, Cristina Feier, Cristoph Bussler, and Dieter Fensel. Wsmo - web service modeling ontology. In *DERI Working Draft 14*, volume 1, pages 77–106, BG Amsterdam, 2005. Digital Enterprise Research Institute (DERI), IOS Press.
- [41] Roy Bragg. Cloud computing: When computers really rule: <http://www.technewsworld.com/story/63954.html>, 2008.
- [42] Aaron Weiss. Computing in the clouds. *netWorker*, 11:16–25, December 2007.
- [43] Lamia Youseff, Maria Butrico, and Dilma Da Silva. Towards a unified ontology of cloud computing. In *Grid Computing Environments Workshop, 2008. GCE '08*, pages 1–10, Nov 2008.



# Multi-Agent Architecture for Solving Nonlinear Equations Systems in Semantic Services Environment

Victor Ion Munteanu, Cristina Mindruta, Viorel Negru, Calin Sandru  
Computer Science Department  
West University of Timisoara  
{vmunteanu, cmindruta, vnegru, csandru}@info.uvt.ro

**Abstract**—A semantic enabled multi-agent architecture for solving nonlinear equations systems by using a service oriented approach is proposed. The service oriented approach allows us to access already implemented methods for solving complex mathematical problems. The semantic descriptions of these services provide support for intelligent agents.

## I. INTRODUCTION

THE MAIN goal of this paper is to propose a multi-agent architecture for solving nonlinear equations systems. This architecture is designed around a semantic-based solving paradigm supported by a service oriented ontology. That allows us to define expert agents in a semantic services environment.

Similar work has been done in the MONET project [3] which had the aim to provide a set of web services together with a brokering platform in order to facilitate means of solving a particular mathematical problem. The semantic representation for the mathematical objects was done using OpenMath (MathML was cited also).

GENSS (Grid-Enabled Numerical and Symbolic Services) project [2], like MONET, tries to combine grid computing and mathematical web services using a common open agent-based framework.

In [8] is discussed the matchmaking of semantic mathematical services described using OpenMath.

The architecture we propose is being built based on past experience in designing NESS, a non-linear equations systems solver, and EpODE, an expert system dedicated to ordinary differential equations. NESS [10] is an intelligent front-end for solving non-linear equation systems, developed in CLIPS. Starting from the features of the system to be solved and of the numeric methods, human expert uses domain knowledge (numeric analysis) and heuristics to choose the most suitable method, to interpret the results (intermediary and final), and to restart the solving process in the case of failure. NESS uses task-oriented reasoning. A MAS architecture based on UPML has been proposed and instantiated for NESS [12]. EpODE was initially realized as a monolithic expert system [11] and has been re-engineered as a semantic services oriented framework [9]; the solving methodology is workflow-oriented, being realised by integrating semantic services with process modelling.

While having similar main functional objective with NESS, the architecture proposed in this paper is more flexible due to the semantic services component and to the new society of agents designed accordingly.

We have been designed a multi-agent architecture that will implement a task-oriented solving model, with a core semantic-based solving paradigm. Typically, multi-agent architecture offers flexibility, scalability and mobility, important quality attributes when dealing with a large number of software services. The agents in the architecture have capabilities ranging from semantically searching for services to providing an execution plan for the given problem. The problem that is to be solved is passed to the multi-agent system as input data. The expert agents in the system analyze the problem and propose an execution plan in order to find the solution. The execution plan is ran and adjusted if need be, and the result of the execution is returned to the user.

The execution plan contains numeric methods which can be offered by software services.

We have also designed a semantic services ontology in order to support the semantic-based solving paradigm. For semantic descriptions we have decided to use WSMO (Web Services Modelling Ontology)[1]. Our approach considers a semantic services context, which is able to offer semantic information useful to the system of agents. In this context, the proposed multi-agent system uses a specific ontology containing concepts, relations and axioms defined for the nonlinear equations systems domain, and has an extensible database of semantic descriptions for services implementing numeric methods.

The system implements a core paradigm for solving problems, based on semantic matching between problem properties and numeric method capabilities. Numeric methods are identified based on their semantic descriptions that reflect the properties of the problem for which the method is appropriate. The method selection can be realized by the user based on his own expertise, by the user based on system recommendation and estimations, or automatically by the multi-agent system.

This core paradigm is included in a more flexible approach to solve the problems, that implies coordinated activity in the society of agents and with the user. This can result for example in starting to solve a problem with a numeric method and, form

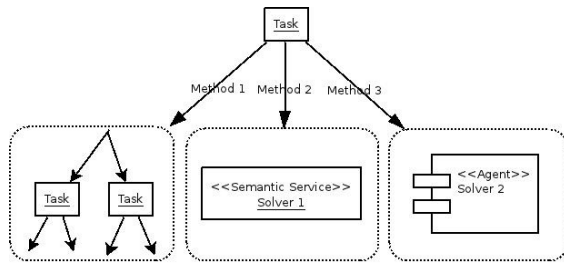


Fig. 1. Task structure

a given step, to continue with another numeric method based on intermediary results and performance of the system.

Such a flexibility is provided by the proposed multi-agent architecture and covers a large area of user skills, from users with simple mathematical skills, for which the system could provide a solution based on its own expertise, to users with very high mathematical skills, which want to experiment solving new types of problems and using new numeric methods. The experience gained by the later category of users is also captured by the multi-agent system in new methods and new characteristics of the existing ones, thus improving its capabilities.

The paper is structured as follows. Section II presents the proposed multi-agent architecture, based on the task-oriented model and integrated with the semantic infrastructure. Section III is dedicated to the semantic descriptions of the ontology, services and goals used to support the core solving paradigm. Section IV presents conclusions and future work.

## II. MULTI-AGENT ARCHITECTURE

When developing the multi-agent architecture, we had several architectural concerns in mind. The architecture must:

- Allow service operations: publishing, semantic facilitation, invoking (running) etc.
- Provide automatic semantic service selection and composition.
- Detect and recover from a failing service.
- Monitor services.
- Create solving scenarios based on previous solving experience.

### A. Task oriented reasoning

Although sometimes associated with an activity to execute, the concept of task is mostly intended to abstract a specific goal to be achieved [6], [1], [5]. In this regard, tasks definitions do not explicit the particular method to use in order to achieve the goal, but rather give a description of the state of the world to be achieved.

The operational aspect can be abstracted in the concept of a problem solving method (PSM). A PSM describes how to achieve a result based on a set of input data. The goal of a PSM can be regarded as a procedural one in order to obtain a result according to the method specification. The meta-properties of the methods to be mentioned in this context include input, output, precondition, postcondition, sub-task.

The task oriented model is the abstract methodological basis for the proposed nonlinear equations system solver, task oriented reasoning being a natural approach for our multi-agent system. Starting from a particular task, one can build a hierarchy of tasks and PSMs that can be considered as the plan for solving the root task. In our architecture, the PSMs can be implemented as semantic services or as agents (fig.1).

### B. Agents

The multi-agent system is composed of the following agents: client, reasoner, executor, monitoring, archiver, historian, service discoverer, service wrapper, and solver. These agents can be seen in fig.2.

The client agent handles the communication with the user. It exposes a graphical user interface which allows the user to interact with the system. The client agent communicates with the reasoner, executor and monitoring agents in order to manage the planning and execution.

The reasoner agent receives the problem from the client agent and creates the work plan to solve the current problem. It uses the domain ontology to create the semantic definitions of the tasks in terms of WSMO goals and will interact with the WSMX platform in order to find solving methods for a specific task. The historian agent provides the reasoner with plans that were applied in similar problems. The reasoner agent interacts with the executor agent when it needs to update the plan or when an alternative path is needed.

The executor agent handles the execution of the work plan. It communicates with the reasoner agent in order to receive and detail the work plan. The executor accesses the WSMX platform endpoint and queries it in order to find semantically compatible services for the current step in the work plan. It will communicate with the monitoring agent in order to report task progress.

The monitoring agent has the role to monitor the current task execution. It will send task information to the client agent, which in turn will notify the user about the current state of the execution. The monitoring agent will also create an execution profile and will send it to the archiver for storage.

The historian agent receives the current problem profile from the reasoner agent and it will look in the archive for similar profiles. These profiles are sent back to the reasoner so that it can make a decision based on past executions.

The service discoverer agent looks for new services that can be integrated in the system. It will look in WSIL/UDDI service directories in order to identify compatible services based on the ontology, and will search the JADE's directory facilitator for compatible solver agents.

The service wrapper handles services with no semantic description. It generates it's own WSML file for the service it is wrapping around and publishes it in the WSMX platform.

### C. WSMX Platform

The web services that the system uses expose their capabilities, ways to interact with them, and ontologies through WSMO-compliant (WSML) descriptions. These web services

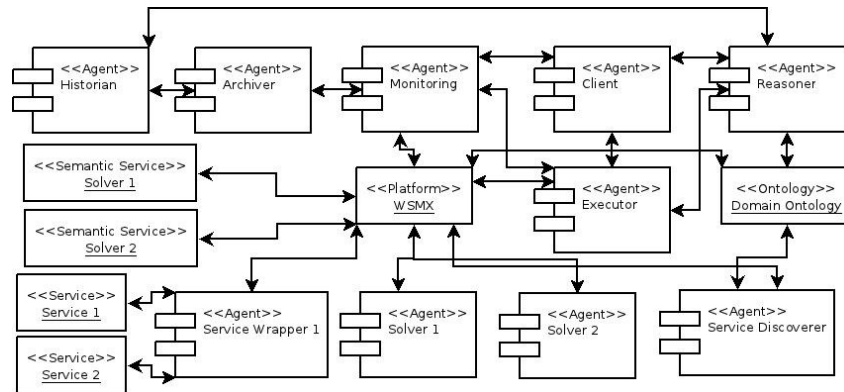


Fig. 2. Multi-agent system architecture

will be deployed on the WSMX platform. The WSMX platform can match the semantic descriptions of web services with semantic descriptions of goals provided by the agents, and can invoke the execution of those services. On the WSMX platform will be deployed semantic services, semantic agents that can wrap around non semantic services, and semantic agents that act alone. These services will execute the tasks given by the executor agent.

### III. SEMANTIC MODEL

WSMO is appropriate for modelling the proposed semantic-core solving paradigm, because it offers a clear separation between goals and services. Services will offer numeric methods or compositions of methods from our paradigm, and goals are dynamically built for each problem to be solved.

1) *Ontologies*: NESOnto (fig.3) contains concepts, relations and axioms that define problem and solution spaces of the nonlinear equations systems.

NESOnto defines the concept of problem (NES\_Problem) in relation to the concepts representing the input data of the problem (NES\_IN\_P) and the properties of the corresponding nonlinear equations system (NES\_PPProps). It also defines the concept of numeric method (NES\_Method) in relation to the concepts representing the input data of the method (NES\_IN\_M) and the properties of the method (NES\_MPProps). The restrictions imposed on the solving session are modelled with the concept NES\_Session, and the solution of the nonlinear equations system is modelled with the concept NES\_Solution.

The associated matrix is modelled with two concepts: Jacobian represents the symbolic Jacobian of the system, and JacobianVal represents the Jacobian matrix of the system computed in a given point.

The properties of the problem are of types defined in specific concepts (e.g. SystemForm), and for each of these concepts the particular instances (e.g. General, Sparse, DiagonalExplicit) have been defined.

The properties of the numeric method have been defined in the same manner. They will be used by the reasoner agent, that will refine the service selection and composition matching

them to the execution constraints represented as an instance of the NES\_Session concept.

2) *Services implementing numeric methods*: The capabilities of each service are described using NESOnto and an ontology specific to the service. One of the axioms in this specific ontology defines the relation *isSolvable* expressing the properties of the nonlinear equations systems for which the method is appropriate.

In fig.4 (a) is represented the semantic description of Broyden\_NES service that implements the numeric method Broyden for solving nonlinear equations systems. The precondition in the semantic description states that the method can be used if the relation *isSolvable* exists for the properties of the problem to be solved. The semantics of this relation for the service Broyden\_NES are defined in the axiom *isSolvableDef* of the ontology particular to the capabilities of this service, and expresses the fact that Broyden method is recommended for nonlinear equations systems of general form, with nonsingular Jacobian, and of medium or big size. This represents a part of the expert knowledge and is implemented in the semantic description of the service.

3) *Goals*: Goals are dynamically built for each problem. Each goal has its a particular ontology that contains instances of concepts from NESOnto. In order to identify the services which are able to solve the corresponding problem, the particular ontology (GoalNESSolution) contains an instance of the NES\_PPProps concept which holds the concrete properties of the given problem. In fig.4 (b), an example goal of solving a nonlinear equations system is described in WSML.

### IV. CONCLUSIONS AND FUTURE WORK

The task-oriented model makes a clear separation between tasks to be accomplished and methods for solving them. This model is implemented using a multi-agent architecture and is applied to design a solver for nonlinear equations systems.

The multi-agent system is integrated with services implementing numeric methods. The services are semantically described in terms of a domain ontology we propose for nonlinear equations systems.



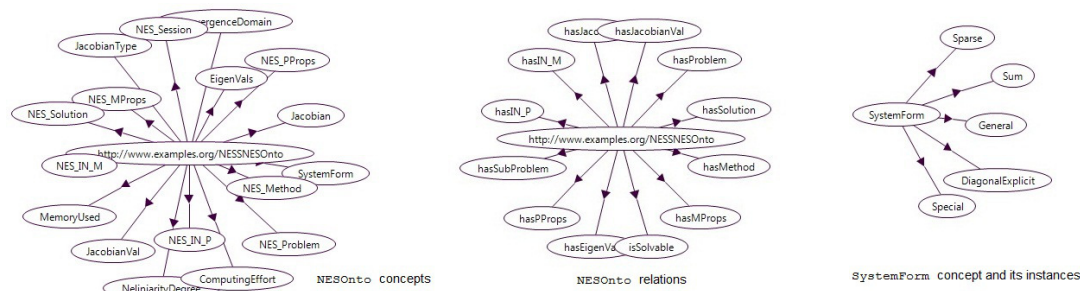


Fig. 3. Elements of NESOnto represented with WSMT

```

webService Broyden_NES
importsOntology (nes#NESOnto, BroydenCapabilityOnto)
capability Broyden_NES_cap
precondition Broyden_NES_pre definedBy
  ?p memberOf nes#NES_Problem and ?pp memberOf nes#NES_PProps and
  nes#hasPProps (?p, ?pp) and isSolvable(?pp).
postcondition Broyden_NES_post
  definedBy ?s memberOf nes#NES_Solution and nes#hasSolution (?p, ?s).
interface Broyden_NES_I
choreography Broyden_NES_chor
stateSignature signAnalyze
in concept nes#NES_Problem
out concept nes#NES_Solution
transitionRules_#
  add(?s memberOf nes#NES_Solution)
  add(@nes#hasSolution(?p, ?s))

ontology BroydenCapabilityOnto
axiom isSolvableDef
  definedBy
  ?pp memberOf nes#NES_PProps and ?pp[nes#Form hasValue nes#General] and
  ?pp[nes#JType hasValue nes#NonSingular] and ?pp[nes#size hasValue ?x]
  and ?x>10 implies nes#isSolvable(?pp).

```

(a)

```

goal ExampleGoal
nfp
dc#description hasValue "Goal of solving a system with general form,"
"any nelinearity degree, nonsingular Jacobian, of size 100."
endnfp
capability EG_1
importsOntology (nes#NESOnto, GoalNESSolution)
postcondition post_EG_1
  definedBy ?s memberOf nes#NES_Solution and
  nes#hasSolution (problem, ?s).
interface itf_EG_1
choreography Cor_EG_1
stateSignature signAnalyze
in problem
out nes#NES_Solution
transitionRules_#
  add(?s memberOf NES_Solution)
  add(@nes#hasSolution(problem, ?s))

ontology GoalNESSolution
instance problem memberOf nes#NES_Problem
  title hasValue "AnExampleProblem"
instance pProps memberOf nes#PProps
  nes#Form hasValue nes#General
  nes#JType hasValue nes#NonSingular
  nes#size hasValue 100
relationInstance nes#hasPProps(problem, pProps)

```

(b)

Fig. 4. (a) WSML descriptions for BroydenNES service and (b) ExampleGoal goal

Semantic descriptions are realized in WSML, allowing us to benefit from the clear separation between goals and services. This maps over our task-oriented model, specifically over tasks and solving methods respectively. Specialized agents dynamically build semantic descriptions of the goals based on the properties of problems to be solved, and appropriate services are discovered by semantic matching.

Other agents are implied in user interaction, in building work plans, in managing the system in order to offer solutions to different problems or support for the human expert to experiment methods to solve nonlinear equations systems.

The multi-agent system is implemented in JADE [4]. The reasoner agent uses JESS [7] as a rule engine together with the domain ontology in order to devise the execution plan. The system will be validated in the context of NESS.

In our future work we will extend the definitions of the knowledge in the domain of nonlinear equation systems and will try to combine WSML with OpenMath and MathML in expressing them.

## V. ACKNOWLEDGMENT

This work was partially supported by the Romanian Government PNII grant nr. 12118/2008 (SCIPA), by POSDRU/88/1.5/S/49516 structural funds grant, ID 49516 (2009), and by the grant POSDRU 21/1.5/G/13798.

## REFERENCES

- [1] WSMO. <http://www.wsmo.org>.
- [2] GENSS project. <http://genss.cs.bath.ac.uk>, 2004.
- [3] M.L. Aird, W.B. Medina, and J. Padget. MONET: service discovery and composition for mathematical problems. In *Cluster Computing and the Grid, 2003. Proceedings. CCGrid 2003. 3rd IEEE/ACM International Symposium on*, pages 678 – 685, may 2003.
- [4] Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. John Wiley & Sons, 2007.
- [5] B. Chandrasekaran. Design problem solving: a task analysis. *AI Mag.*, 11:59–71, October 1990.
- [6] Dieter Fensel, Enrico Motta, Frank van Harmelen, V. Richard Benjamins, Monica Crubezy, Stefan Decker, Mauro Gaspari, Rix Groenboom, William Grosso, Mark Musen, Enric Plaza, Guus Schreiber, Rudi Studer, and Bob Wielinga. The unified problem-solving method development language UPML. *Knowl. Inf. Syst.*, 5:83–131, March 2003.
- [7] Ernest Friedman Hill. *Jess in Action: Java Rule-Based Systems*. Manning Publications Co., Greenwich, CT, USA, 2003.
- [8] Simone Ludwig, Omer Rana, Julian Padget, and William Naylor. Matchmaking framework for mathematical web services. *Journal of Grid Computing*, 4:33–48, 2006. 10.1007/s10723-005-9019-z.
- [9] Cristina Mindruta and Dana Petcu. A semantic services architecture for solving ODE systems. *Symbolic and Numeric Algorithms for Scientific Computing, International Symposium on*, 0:301–307, 2010.
- [10] Viorel Negru, Stefan Maruster, and Calin Sandru. Intelligent system for non-linear simultaneous equation solving. In *Technical Report Report Series. No. 98-19. RISC-Linz*, december 2003.
- [11] Dana Petcu. EpODE. <http://www.info.uvt.ro/~petcu/epode/main.htm>.
- [12] Calin Sandru and Viorel Negru. Validating UPML concepts in a multi-agent architecture. In *Schedae Informaticae*, volume 15, pages 109–126, 2006.

## Cloud-based Assistive Technology Services

Ane Murua  
Vicomtech-ik4  
Mikeletegi Pasealekua, 57  
20009 Donostia-San Sebastián,  
Spain  
Email: amurua@vicomtech.org

Igor González  
Iriscom Sistemas S.L.  
Torreatze ID  
20110 Pasaia San Pedro, Spain  
Email: igonzalez@iriscom.org

Elena Gómez-Martínez  
R&D Department,  
Technosite-ONCE Foundation  
C/Albasanz, 16  
28037 Madrid, Spain  
Email: megomez@technosite.es

**Abstract**—Cloud computing will play a large part in the ICT domain over the next 10 years or more. Many long-term aspects are still in an experimental stage, where the long-term impact on provisioning and usage is still unknown.

While first attempts at this field focused on service provisioning for enterprises, cloud is reaching individuals nowadays. Our proposal is to go a step further and, based on the proven benefits of the Cloud, improve Internet and technology access for those people always left behind when any technological progress takes place.

This paper presents the Cloud-based Assistive Technology Service delivering to individuals who face technology accessibility barriers due to ageing or disabilities. An example of how an Assistive Service is delivered to an individual in an easy and seamless way is given as a demonstration of how the future should be. This proof of concept has been developed within the INREDIS research project.

### I. INTRODUCTION

CLOUD computing enables companies, public administrations and individuals, using networks such as the internet, to access their data and software on computers located somewhere else. It can help businesses – especially SMEs – to drastically reduce information technology costs, help governments supply services at a lower cost and save energy by making more efficient use of hardware. Cloud computing is already used widely, for example for web-based e-mail services.

Cloud computing has the potential to develop into a major new service industry, presenting great opportunities for European telecoms and technology companies. Client companies and public administrations can benefit from lower costs and state-of-the-art services by using cloud computing rather than installing and maintaining software and computing equipment of their own [1].

On the other hand, as our countries build out their broadband infrastructures to ensure that broadband reaches everyone, it is important that 'everyone' includes people with disability, literacy and aging related barriers to Internet use. We

The research described in this paper arises from a Spanish research project called INREDIS (INterfaces for RELations between Environment and people with DISabilities) [4]. INREDIS is led by Technosite and funded by CDTI (Industrial Technology Development Centre), under the CENIT (National Strategic Technical Research Consortia) Programme, in the framework of the Spanish government's INGENIO 2010 initiative. The opinions expressed in this paper are those of the authors and are not necessarily those of the INREDIS project's partners or of the CDTI.

need to be sure that we don't stop at just connecting people to the Internet - but that we also see to it that they can actually use it, and benefit from all that it has to offer [2].

In other words, Cloud has much to contribute to Assistive Technologies' (ATs) ecosystem –Section II–. This paper presents part of the work done within the INREDIS research project –Section III–, namely, a basic architecture to be used for AT selection and delivering –Section IV– and a concrete use case developed for showing what would be the end result for users of the Cloud-based Assistive Technology Services of the future –Section V–. Finally, some future work is proposed –Section VI–.

### II. THE ASSISTIVE TECHNOLOGIES' ECOSYSTEM

#### A. What Does "Assistive Technology" Stand For

According to the definition provided in ISO 9999:2007 "Assistive products for persons with disability -- Classification and terminology" [3], Assistive Products are understood to be any product (including devices, equipment, instruments, technology and software) specially produced or generally available, for preventing, compensating for, monitoring, relieving or neutralizing impairments, activity limitations and participation restrictions. Assistive Technology is technology used by individuals with disabilities in order to perform functions that might otherwise be difficult or impossible. Assistive technology can include mobility devices such as walkers and wheelchairs, as well as hardware, software, and peripherals that assist people with disabilities in accessing computers or other information technologies.

Moreover, according to the definition provided by the Class 22 of ISO 9999:2007 "Assistive products for communication and information", AT ICT products are understood to be devices for helping a person to receive, send, produce and/or process information in different forms. Included are, e.g., devices for seeing, hearing, reading, writing, telephoning, signalling and alarming, and information technology.

#### B. The Assistive Technology Market

The number of AT ICT products cataloged by European databases is well over 20.000, and the AT ICT industry in the EU is not a simple one. It is complex in various aspects, for example for the large number of products, for the large number of small firms, and for the different service provider

systems that are used to get AT ICT products to disabled end-users.

However, one area common to the vast majority of firms is the marketing challenge: how to get the right product, via the right person, and with the right instructions and training to the disabled end-user who needs it. To some extent, this is a distribution and marketing challenge common to any industry, but in the AT ICT industry in Europe, the complexity of the different service provider systems is an extremely potent force in the marketplace [4].

First AT ICT products were dependent both from the platform and the device where they were installed. The only devices where ATs were used were computers, and the way for getting them was to purchase the AT into a local reseller, to receive the AT via mail or (the few times that it was possible) to download its installation program from the web.

This situation is slowly changing, but there are still large barriers for customers [5]:

#### *Awareness*

- End users are largely unaware of the available AT solutions.
- There is a lack of dedicated training in AT products and their capabilities, resulting in end users having AT they cannot use to the full extent, or in some cases not at all.
- ATs that are easiest to obtain are also the ones most abandoned.

#### *Price*

- High purchasing costs for end users are reported as a major barrier for wider deployment by disability organizations.

#### *Mismatch between end user needs and offered AT*

- End users are not provided with the required AT, resulting in a considerable percentage of obtained ATs being discarded within a year.
- AT that is being offered does not always satisfy the actual needs of the people with disabilities, hence their refusal to use them.
- According to some surveys, almost half of the end users experience problems using ATs.

### III. INREDIS BASIC RESEARCH PROJECT

The work presented here belongs from the INREDIS project [6]. This project has carried out basic research in the field of accessible and interoperable technologies, with the scope of developing basic technologies that will enable the creation of communication and interaction channels between people with special needs and their environment.

A multidisciplinary team composed by 14 companies and 18 research partners has been involved in INREDIS The project began on 2007 and ended in December 2010.

INREDIS project aims a total integration of functional disabled users into the Information Society, including popular fields such as: domotic, urban and local mobility, shopping information, banks, digital television and other areas of interest.

The base of such approach is that a person with an adapted or personalized controller (where *controller* does mean a

fitting of a proper User Interface into a suitable hardware) should be able to interact and control different services and devices by means of an interoperability architecture. Using a personalized controller, the achievement of a proper control is eased significantly as the accessibility problem of the whole environment is reduced to just solving, if any, the accessibility issues with the user controller.

A prototype can be seen in [7] showing the above (see Fig. 1).

In this example (further described in [8]), a pluggable User Interface for people with visual impairments can be seen, which is based on a DHTML page running on a vertically handled tablet-PC.

This DHTML page is rendered in a web browser. Popular browsers, such as Microsoft Internet Explorer, Mozilla Firefox or Safari, are supported. The DHTML code has been correctly tagged so that it is compatible with screen readers such as Jaws [9]. An Universal Control Hub (UCH) [10] acts as a gateway allowing pluggable UIs to remotely control the TV set.

#### *A. Proposed Architecture*



Fig 1. DHTML based UI for accessible TV control

Fig. 2 (next page) shows the INREDIS architecture at high-level.

Firstly, user's device communicates to INREDIS architecture core, which connects up *Interoperability Platform*, containing different interoperability protocols (i.e. UCH, OSGi and Web Services).

The Interoperability Platform selects the corresponding protocol in order to interact with a specific device or service.

Once the service is known, it is necessary to adapt its user interface. So, the user's device information and the user's profile are asked to the *Adaptive Modeling Server*, which looks them up into the *Knowledge Base*.



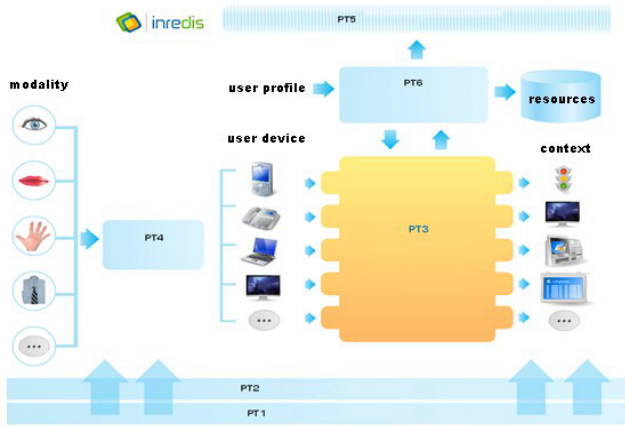


Fig 2. INREDIS Architecture (High-level)

This information, together with the context information, is headed for the *AT Middleware*, which queries to *Knowledge Base* the most appropriate AT and delivers it as a service (SaaS approach) [11].

All collected information is passed to Interface Generator in order to provide the most suitable user interface.

#### IV. INREDIS AT MIDDLEWARE

The AT Middleware is the module responsible for providing AT services within the INREDIS architecture.

Unlike the legacy purchase methods introduced at Section II, the AT Middleware instantly provides the user the right ICT AT with his desired configuration.

In this sense, as seen in Fig.3, if an AT delivery service does exist within an architecture that aims to simplify the interaction of people with disabilities with the devices and services of their environment, why not export this service and use it to solve disability, literacy and aging related barriers to Internet use as exposed on Section I?

Fig. 4 shows AT Middleware's sequence diagram when finding an AT within INREDIS.

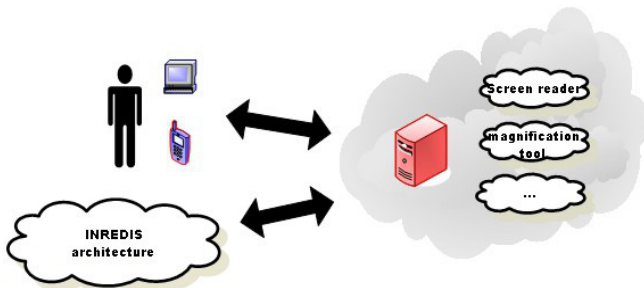


Fig 3. Using the AT Middleware (right) directly (up) and via INREDIS architecture (down)

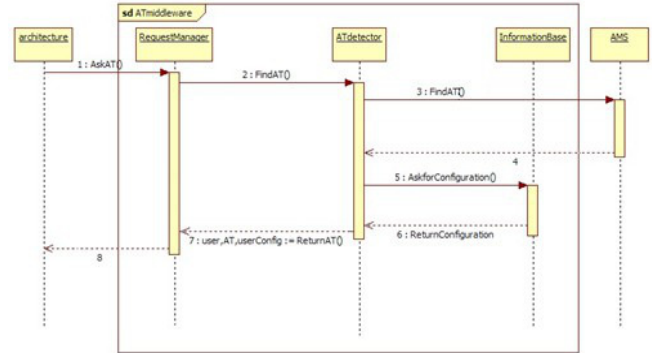


Fig 4. Using the AT Middleware (right) directly (up) and via INREDIS architecture (down)

As seen on the diagram, the AT Middleware is composed by 3 modules: Request Manager; AT Detector; AT Information Base.

The Request Manager is an OSGi bundle. It exposes a Web Service, which is the main entry point of the AT Middleware.

The AT Detector OSGi bundle queries the AT ontology [11] in order to find the most appropriate AT for an user, taking in mind its needs and preferences.

And last, the AT Information Base is composed by a database, where users' personal AT configurations are stored, and an OSGi bundle, which queries the database when needed.

But this approach won't be complete until a wide collection of ICT ATs in the cloud appear in the market. Once this is achieved, AT users will gain the following advantages:

- Access to a wider range of products (multiple products of the same type / multiple product types)
- Access to free AT ICT products
- Access to low-cost products, thanks to the pay-per-use approach
- No need to install a software for every AT product
- Seamless AT execution and use (no need for complex setup procedures or for the execution of various commands continuously)

At INREDIS project, in order to incorporate the first AT ICT products to the platform and to demonstrate that it is possible to give an AT service based on the cloud, we have chosen four common ATs, i.e. a multi-language translator, a text-to-speech (TTS) synthesizer, an avatar who communicates through sign language and an automatic captioning engine [12]. All of them have been successfully transferred to the cloud.

The AT used into the end-to-end implementation presented at Section V is the multi-language translator.

Based on the Google Translate [13] API [14] provided by Google, two cloud-based AT services have been developed. Both services are remotely accessible Web Services: one of them follows a REST approach while the other is the SOAP version of the service.

Those multi-language translation Web Services contain a single method called *translate*. Its input parameters are *textToTranslate*, *sourceLanguage* and *targetLanguage*, while the output is a string containing the translation done by the service.

#### V. PROOF OF CONCEPT: SEAMLESS ASSISTIVE TECHNOLOGY DELIVERING

As a proof of concept of cloud-based AT service delivering, an end-to-end implementation has been done. The goal is to show how the system would look in the eyes of the user, i.e., to demonstrate the benefits of this new market paradigm.

So, an Android application (app) has been adapted in order to make it capable to use cloud-based multi-language translation service when needed. The chosen app is the *Bluetooth Chat* app included into the Android SDK [15].

Basically, three main changes have been made into the original app: (1) communication with the INREDIS AT Middleware added; (2) localized resource sets for showing the UI into the language that the user has set its phone; and (3) the ability to use the cloud based multi-language translation cloud-based AT services developed. The end result for the user is that, once logged into the AT Middleware, if he wants to use the Bluetooth Chat in a place where he won't be understood, the app will be aware of that, switching on the translation without user intervention.

More on detail, the changes made into the Bluetooth Chat app have been the following:

- (1) Communication with the INREDIS AT Middleware:

This process should be as least invasive as possible for the user, even running in the background if possible.

This issue has been solved by developing another Android app (as Fig. 5 shows) that logs the user into the AT Middleware (see Fig. 6).

This app queries the user's profile in order to get its preferences and, when launching the Bluetooth Chat, sends this preferences to it. This way, the Bluetooth Chat knows whether it has to change its original operation mode and start using the cloud-based translation AT service or not.

- (2) Localized resource sets for showing the UI into the language that the user has set its phone:



Fig 5. AT Middleware app (left) and Bluetooth Chat app (right)

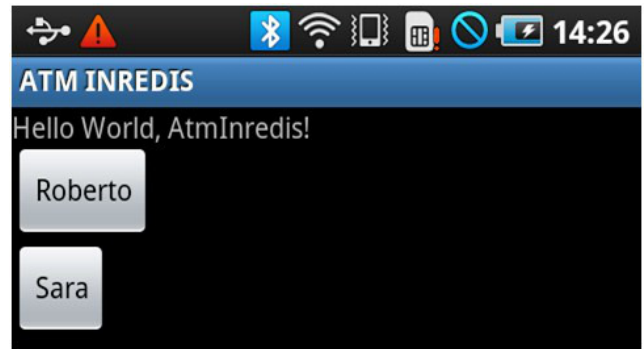


Fig 6. User login UI

The Bluetooth Chat app is, by default, in English. It makes no sense to give an application in English to a user who needs support to communicate in other languages than its own native language. For this reason, we have modified the original app, adding localized resources that make possible the use of the user's preferred language by the UI (e.g., menus, info messages,...) of the app.

We have assumed that user's preferred language is the one selected into the locale configuration of its mobile phone. By default, the Android system selects which resources to load, based on the device's locale [16]. So, multiple resource sets on different languages have been added to the original structure of the app, creating different *res/values-xx* folders, (where *xx* is the code of each different language).

If greater levels of customization want to be achieved, the language in which the user has configured its mobile phone can be bypassed. This way, the incoming information from the AT Middleware would be used to set the language of the Bluetooth Chat's UI.

- (3) The ability to use the cloud based multi-language translation cloud-based AT services developed:

Finally, it is necessary to integrate the translation service into the chat application. The intended result is that two people can communicate in their native or preferred languages. I.e.:

Being *A* an Spanish user, and *B* an English user of the system, both users run the Bluetooth Chat app in their phones and pair their devices via bluetooth. *A* writes on her device a message in Spanish and presses "enviar" (send) button. *B* receives *A*'s message in English. *B* writes on his device a message in English and presses the "send" button. *A* receives *B*'s message in Spanish.

In order to make this scenario possible, the multi-language translation Web Services previously developed have been seamlessly integrated. This way, when user clicks the "send" button, a petition is sent to the translation Web Service, containing the message to be translated and the languages of origin and destination. The petition is processed on the cloud instantly and the translated message is returned and delivered to the recipient.

Since the language translation Web Services developed can be invoked via SOAP or REST, the procedures for remotely accessing them from the chat application differ one from another.

Android's own libraries are used for querying the REST Web Service for a translation. In particular, the `android.net.Uri` [17] class is used, which eases the building of the URI of the call from the various parameters involved (e.g.: `http://192.174.15.29:8080/GoogleTranslatorREST/resources/methods?textToTranslate=hello&sourceLanguage=en&targetLanguage=es`) and processes its response.

In the case of the SOAP Web Service, Android has no specific libraries to request the Web Service. The option chosen is the use of the library `ksoap2-android` [18].

Finally, Fig. 7 shows the final appearance of the Bluetooth Chat app when using the cloud-based multi-language translation Assistive Technology service:

## VI. FUTURE WORK

Once demonstrated that the Cloud-based Assistive Technology service delivering approach is feasible, it still remains a lot of work to do in order to see this system in the market in a short period of time.

The most important work to do is to progressively migrate all the ICT ATs to the cloud. In this sense, the resources that AT developers have available are limited, so, migrating and supporting the actual users in parallel will be a big challenge for them.

There are some efforts trying to standardize the accessibility APIs present on different operating systems. Once this achieved, the task of migrating ICT ATs to different OS and to the cloud will be much easier.

## REFERENCES

- [1] N. Kroes, "Digital Agenda: Commission seeks views on how best to exploit cloud computing in Europe": <http://europa.eu/rapid/pressReleasesAction.do?reference=IP/11/575&format=HTML&aged=0&language=EN&guiLanguage=en>
- [2] GPII project website: <http://gpii.net/>
- [3] International Organization for Standardization. ISO 9999:2007 "Assistive products for persons with disability -- Classification and terminology"
- [4] Stack, J., L. Zarate, C. Pastor, N.-E. Mathiassen, R. Barberà, H. Knops, H. Kornsten (2009) Analyzing and federating the European assistive technology ICT industry, Final Report, March 2009.
- [5] Lopes, R., Bandeira, R., Carriço, L., Van Isacker, K. (2010). Towards Mobile Web Accessibility: Vision And Challenges. Proceedings of the first International ÆGIS Conference - 7-8 October 2010 (pp. 151-158).

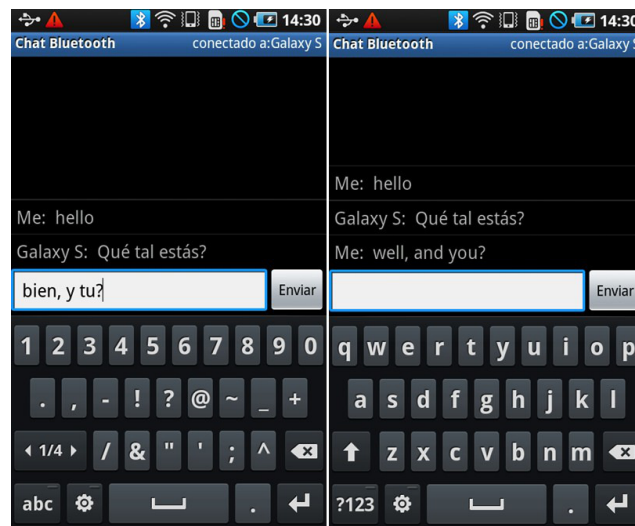


Fig 7. User sends chat message while Spanish to English translation is on

- [6] INREDIS project website: [www.inredis.es](http://www.inredis.es)
- [7] URC Based Accessible TV video. <http://youtu.be/9xGMTuVZRkY>
- [8] Epelde, G. et al.: URC based accessible TV. In: Proceedings of the 7th European conference on Interactive Television (EuroITV 2009), pp. 111–114. ACM, New York, NY, USA (2009)
- [9] Freedom Scientific. Jaws for Windows website: [http://www.freedomscientific.com/fs\\_products/software\\_jaws.asp](http://www.freedomscientific.com/fs_products/software_jaws.asp)
- [10] Zimmermann, G., and Vanderheiden, G. 2007. The Universal Control Hub: An Open Platform for Remote User Interfaces in the Digital Home. In Proceedings of Human-Computer Interaction International 2007 (Beijing, P.R. China, July 22-27, 2007). HCI'07. Springer, Heidelberg, LNCS 4551, 1040–1049.
- [11] Iglesias-Pérez, A. et al. 2010. A Context-Aware Semantic Approach for the Effective Selection of an Assistive Software. In proceedings of the IV International Symposium of Ubiquitous Computing and Ambient Intelligence (UCAml 2010).
- [12] INREDIS project. D5.5.1 deliverable
- [13] Google Translate website: <http://translate.google.com>
- [14] Google Translate API on Google code: <http://code.google.com/apis/language/translate/overview.html>
- [15] Android Developers, "Bluetooth Chat sample code": <http://developer.android.com/resources/samples/BluetoothChat/index.html>
- [16] Android Developers, "Localization": <http://developer.android.com/guide/topics/resources/localization.html>
- [17] Android Developers, "Uri public abstract class": <http://developer.android.com/reference/android/net/Uri.html>
- [18] ksoap2-android library: <http://code.google.com/p/ksoap2-android/>





# Semantic P2P Search engine

Ilya Rudomilov

Czech Technical University in Prague  
 Department of Computer Science and Engineering  
 Email: rudomily@fel.cvut.cz

Prof. Ivan Jelínek

Czech Technical University in Prague  
 Department of Computer Science and Engineering  
 Email: jelinek@fel.cvut.cz

**Abstract**—This paper discusses the possibility to use Peer-to-Peer (P2P) scenario for information-retrieval (IR) systems for higher performance and better reliability than classical client-server approach. Our research emphasis has been placed on design intelligent Semantic Peer-to-Peer search engine as multi-agent system (MAS). The main idea of the proposed project is to use semantic model for P2P overlay network, where peers are specified as semantic meta-models by the standardized OWL language from The World Wide Web Consortium. Using semantic model improve the quality of communication between intelligent peers in this P2P network. Undoubtedly, proposed semantic P2P network has all advantages of normal P2P networks and in the first place allow deciding a point with bottle-neck effect (typical problem for client-server applications) by using a set of peers for storing and data processing.

## I. INTRODUCTION

RELEVANCE and popularity of Peer-to-Peer networks (P2P) increases every year due to the exponential growth in the number of documents on the Internet and local networks. Already existing and often used Web-search engines with the client-server architecture have problems with storing, processing a large number of documents because of possibilities for centralized solutions (e.g. „bottle-neck effect“). Otherwise, P2P systems provide distributed storing and analyzing data in a set of network members (i. e. "peers").

There is no doubt about the need to find other technologies improve the efficiency of search. One way is to just access a distributed P2P model. This trend is observed not only in commercial areas, but is the subject of set of academic research [3]. High attention to these issues came just at the beginning of the 2000s with the founding The Gnutella network (originally a P2P-file distribution system), which could already be used in a search option [13]. The first of these was developed based on the Gnutella network in 2000 - the search engine InfraSearch [5] and later bought by Sun with name JXTA.

This research is supported by the Grant Agency of the CTU in Prague (grant No. SGS11/129/OHK3/2T/13) and the Visegrad Fund (No. 51100034).

## II. P2P BACKGROUND

Peer-to-Peer networks can be classified by used topology, from client-server-like centralized P2P to fully-decentralized P2P networks without central coordination and censorship. For Information-retrieval systems it means indices allocation indices of nodes content, from one centralized index (centralized P2P) to distributed indices among all nodes (decentralized P2P).

### A. Centralized P2P

Centralized P2P systems apply advantages of the client-server architecture to P2P networks. There are one or more central servers for nodes coordination (Fig. 1), ensuring of network policy and locate desired documents. Similar to client-server case, node are sending request to server for desired document, but in the centralized P2P network answer contains just addresses of nodes with desired documents for further interaction.

Thereafter the centralized P2P is susceptible to malicious attacks as another centralized systems and single point of failure. Moreover, this category of P2P networks will become a bottleneck for a large number of peers.

Single difference from client-server is in direct data transfer between nodes and thus server did not need to store all files of the network. Follow-up researchers increased competencies of nodes, but defined central server (or few) are necessary for network action.

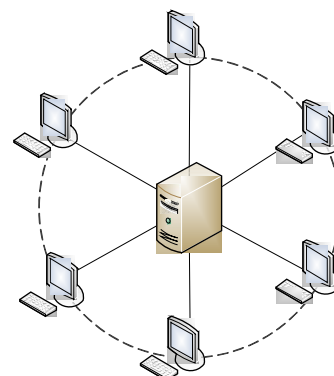


Fig 1. Topology of centralized P2P network

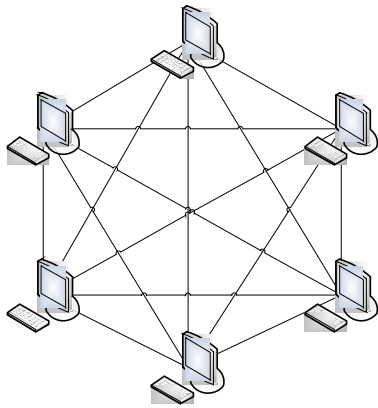


Fig 2. Topology of decentralized P2P network

### B. Decentralized P2P

A decentralized P2P network is “ideal” P2P network because of one-range topology, equal rights and responsibilities of nodes. There are no central server (Fig. 2) and it assist protection from malicious attacks, censorship, scalable limitations.

The main problem of this type of P2P is coordinating. Nodes in decentralized P2P networks interact directly and coordination are maintained by all nodes. There are two approaches for coordination by using different types of logical network topology, difference between them lies in query transferring between nodes:

#### 1) Unstructured

In an unstructured P2P each node is responsible for its own data, and keeps track of a set of neighbors that it may forward queries to. Nodes have not strict mapping between data and nodes. Classical P2P network Gnutella is related to this category: joining node broadcasts ping-message through whole network and waiting for pong-answer from working nodes. Thereby, flooding is typically method there.

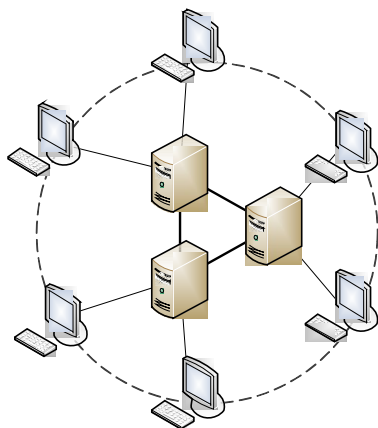


Fig 3. Topology of hybrid P2P network

#### 2) Structured

A structured P2P supports nodes mapping by using pre-defined strategies (in the first place distributed hash tables, DHT). Nodes can interact with another with more or less strict information about them thanks. As a result, a query can be routed to the node who can have desired documents with the high probability. Majority of structured P2P adopt the key-based routing: CAN, Chord, and Pastry.

#### C. Hybrid P2P

The main advantage of centralized P2P systems is that they are providing a quick and reliable resource locating; the main problem is server limitations. On the other hand, decentralized P2P require more time in resource locating. Different researchers propose to combine techniques of both networks in hybrid P2P concept, which is semi-decentralized and uses a fashion of super-nodes instead of one central server (Fig. 3). There are no difficulties with nodes locating, but problems with super-nodes maintaining for building one-range top level of network.

One of the most famous hybrid P2P network is Edutella (2004), German project for reliable exchange of educational materials. The two-level network is maintained by a set of super-nodes, which are composing HyperCuP level. All documents (document, papers, and videos) in the network are described via defined RDF schema and stored on appropriated super-nodes. A query routing is provided by exchanging RDF indices between super-nodes.

### III. MULTI-AGENT APPROACH FOR P2P

The one of the main modern trends of architecture for P2P search engines is Multi-agent systems (MAS) [9], which consists of a number of intelligent agents, each of which operates independently for the benefit of the whole system. This method allows you to create a fully decentralized or semi-decentralized P2P network, in which, the respective agents work independently or with privileged local coordinating agents [1].

Some modern P2P search engines are developed as a “meta-search engines” [7] for parsing and joining search results from popular commercial engines using [4] and these meta-results are often classified according to own rules [2]. However these systems are based on commercial search engines, and for improved their performance and therefore can not operate independently. These systems may improve search results, but do not work autonomously.

Another actual way to solve is decentralized search engine, among which we can mention the experimental YaCy search engine. YaCy was developed at the University of Karlsruhe in 2003 [16]. YaCy is a P2P search engine without main server with indices and where each Linux-server with an installed YaCy separate downloads, indexes the Web and processes user queries to search for documents throw other servers in the YaCy network. YaCy uses distributed hash tables (DHT) for defined, simple and effective allocation

documents between nodes: nodes calculate (key; value) pairs for all documents in the network and then use these pairs for looking for and getting required file by participating in required DHT. DHT is a classical communication mechanism for P2P networks since P2P networks with popular file-sharing protocols like BitTorrent, Gnutella, Napster, etc [10]. However YaCy uses 4 predefined servers with node lists, therefore YaCy is not fully-decentralized P2P.

IV. SEMANTIC P2P NETWORKS

Information-retrieval systems is one of the main problems of P2P networking through problems with generation and distribution indices among nodes. Although P2P is suitable for content-sharing systems thanks to using DHT from some attributes (filename, author and so on), sharing of documents is connected to more complicated and bigger indexed.

YaCy search engine respects the concept of a decentralized P2P system, but its performance is very small [12] of the principal reasons the time cost of the communication between peers by using DHT. This may resolve the problem uses the idea of semantic P2P (SP2P) [8], in which peers are described as meta-models in the Semantic Web according to standard OWL. SP2P systems eliminate problems associated with the use of common ontologies (e.g. maintenance, scalability). However, we have to explicit to provide explicit semantic mappings (i.e. definitions of semantic relationship) between nodes [8].

The use of Semantic Web techniques in Peer-to-Peer traced back to the project SWAP (Semantic Web and Peer-to-Peer, 2002), coordinated by the University of Karlsruhe. During the project, the researchers analysed the potential of Semantic Web Technologies for Peer-to-Peer, prepared by method descriptions and software prototypes. Researcher on the project Steffen Staab updated and published research results in 2006 [11].

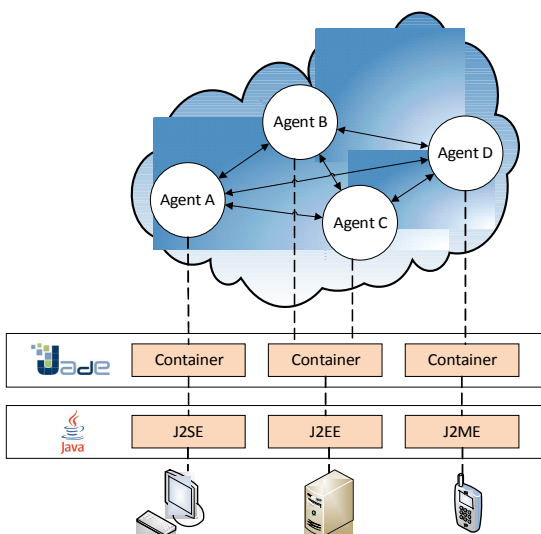


Fig 4. Topology of hybrid P2P network

V. FIPA AND JADE

Open-source JADE framework (Java Agent Development framework) [14] is a free framework for developing Java-based intelligent multi-agent systems and in addition, according to standards from the FIPA (Foundation for Intelligent Physical Agents) [15], a major non-commercial group in the multi-intelligent systems. The FIPA’s membership includes Toshiba Corp., Siemens, Boeing Company, RWTH Aachen University, etc. The widely adopted FIPA standards are the Agent Management and Agent Communication Language (FIPA-ACL) specifications, which already in use as an industry standard.

It is a modern and popular environment, which can be used without restriction or need major interventions and other collaborators in the research. One can certainly believe that the principles of the proposed system will be used not only as a research subject, but in practical applications.

Using JADE for implementation MAS-based P2P applications is a common practice [9] and has obvious advantages:

- Interoperability: JADE is according to FIPA specifications;
- Portability: Java allows to use different platforms and JADE-based implementation can run on J2EE, J2SE, J2ME environment;
- Easy of use: JADE is a set of APIs, which has GUI for a nodes management.

VI. OUR APPROACH

The idea of our project involves the design, testing and implementation of semantic P2P search engine.

The first phases of the project is devoted to a theoretical model of the system, which will have a decentralized architecture and therefore will not use any central node and a set of documents and indices will be placed on any intelligent agent, the amount which will create a multi-agent system (MAS). The theoretical model is based on the ontological reasoning: fully-decentralised P2P network. We build on the progress achieved in this field [13], which expands on the idea of using the semantic model of P2P architectures [8]. Agents will have the same rights and functionality.

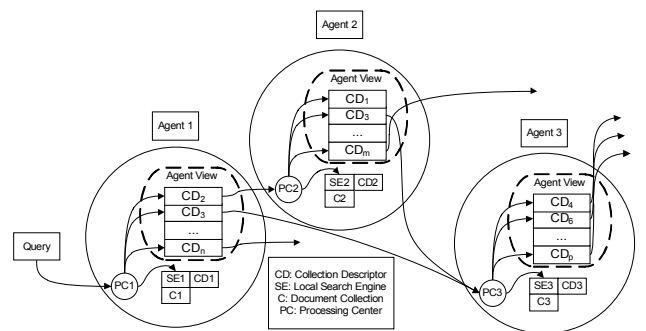


Fig 5. Structure of proposed SP2P network

### A. Node structure

Each intelligent agent will include (Fig. 5):

- A set of documents ("Document Collection") with available information. Document Collection is used by Local Search Engine in searching progress.
- Semantic structure of the interaction of node's Document Collection on neighbour nodes („Collection Descriptor“). Collection Descriptor provides information about neighbours (e.g. IP-addresses) and their content. This component works similar to signature of node: nodes distribute their Collection Descriptors in the network with any reconnection, similar to common P2P-practice (e.g. in Gnutella network) of sending notification ping in time of reconnection to the network. Search engine for searching information around documents on the node („Local Search Engine“). Component looks for relevant documents around Document Collection storage to incoming request.
- Processing center for incoming requests and sending results of searching („Processing Center“). All requests in the network are passing through nodes Processing Centers, which manage sending local requests to the Local Search Engine and coordinate request resending to another nodes because of information about neighbour from their Collection Descriptors.
- Information about other P2P network nodes („Agent-view structure“) is a set of received Collection Descriptors with common methods to parse and store.

### B. Topology

We suppose to use modified Chord (i.e. DHT-based) model of structured decentralized P2P with possibility to distribute semantic indices among nodes. Chord represents a one-dimensional circular fashion of  $2^m$  nodes (Fig. 6), where each node have a ID (i.e. IP-address) and which is arranging from 0 to  $2^m - 1$ . Each node has a predecessor (counter-clockwise node) and successor (the next node in a clockwise direction).

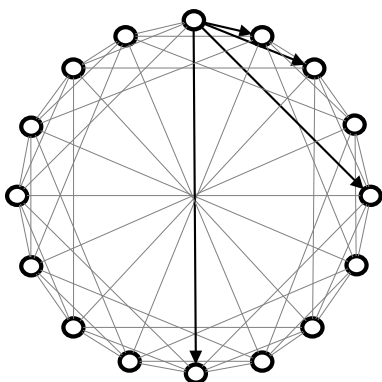


Fig 6. Instance of 16-node Chord network

### C. Semantic mapping

Nodes should communicate with each other using the semantic network map to be created from an "agent-view structure" of agents, allocated on nodes. Semantic P2P is such a P2P, which combines the advantages of structured (in which the nodes are defined) and unstructured (in which network topology is not defined) P2P, and assume that because it does not have the disadvantages of both types, i.e. nodes will have addresses for the fastest routing topology but not defined and the nodes can be disconnected without any problem with the network reliability.

Similar OWL-based ontology for simple agents was developed by Michal Laclavik and his colleagues as AgentOWL project in 2006-2009 [6]. In addition, their project was implemented by JADE framework and we can use their experience in our network.

## VII. FUTURE WORK

On the next phase we will research opportunities to integrate generated semantic indices of nodes into Chord protocol. Finally, we will implement an experimental system using JADE framework, which is suitable for this project because of possibility to use XML (and RDF) for communication between agents and OWL for mapping nodes. We propose to conduct research efficiency of this experimental Semantic P2P network in comparison with existed P2P search engines (e.g. YaCy search engine).

## REFERENCES

- [1] W. Galuba and S. Girdzijauskas: Peer to Peer Overlay Networks: Structure, Routing and Maintenance. *Encyclopedia of Database Systems, Part 16*, Springer, 2009, p. 2056-2061.
- [2] H. Gylfason, O. Khan, and G. Schoenebeck: Chora: Expert-Based P2P Web Search. *Agents and Peer-to-Peer Computing. Lecture Notes in Computer Science, Volume 4461/2008*, Springer, 2008, p. 74-85.
- [3] T. Kathiravelu: Approaches to P2P Internet Application Development. *Proceedings of the Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services (ICAS/ICNS 2005)*, 2005.
- [4] I. Kovalev, M. Rusakov, and M. Tsarev: Search Crossplatform Multiagent System. *Automatic Documentation and Mathematical Linguistics, 2010, Vol. 44, No. 1*, Allerton Press, p. 53-55.
- [5] Iraklis A. Klampanos, J. Barnes, and J. Jose: *Evaluating Peer-to-Peer Networking for Information Retrieval within the Context of Meta-searching*, 2006, p. 530.
- [6] M. Laclavik, M. Babik, Z. Balogh, and L. Hluchy: AgentOWL: Semantic Knowledge Model and Agent Architecture. *Computing and Informatics. Vol. 25, no. 5 (2006)*, p. 419-437. ISSN 1335-9150, Chapters 1, 4, 5.
- [7] J. Lehtikainen, I. Salminen, A. Aaltonen, P. Huuskonen, and J. Kaario: *Personal and Ubiquitous Computing, Volume 10, Number 6*, Longon: Springer, 2006, p. 357-367.
- [8] A. Mawlood-Yunis, M. Weiss, and N. Santoro: From P2P to reliable semantic P2P systems. *Peer-to-Peer Networking and Applications, 2010, Volume 3, Number 4*, Springer, p. 363-381.
- [9] A. Poggi and M. Tomaiuolo: Integrating Peer-to-Peer and Multi-agent Technologies for the Realization of Content Sharing Applications. *Studies in Computational Intelligence, 2011, Volume 324, Information Retrieval and Mining in Distributed Environments*, Springer, p. 93-107

- [10] C. Roncancio, M. del Pilar Villamil, C. Labbé, and P. Serrano-Alvarado: Data Sharing in DHT Based P2P Systems. *Lecture Notes in Computer Science, 2009, Volume 5740, Transactions on Large-Scale Data- and Knowledge-Centered Systems I*, Springer, p. 327-352
- [11] S. Staab and H. Stuckenschmidt: *Semantic Web and Peer-to-Peer*, Springer, 2006.
- [12] G. Weikum: Peer-to-Peer Web Search. *Encyclopedia of Database Systems*, Saarbrücken: Springer Science+Business Media, 2009, p. 2082-2085.
- [13] H. Zhang and V. Lesser: *Toward Peer-to-Peer Based Semantic Search Engines: An Organizational Approach*, Saarbrücken: VDM Verlag, 2008. ISBN 3639084799.
- [14] JADE Software Framework (2009), <http://jade.tilab.com/>
- [15] FIPA Specifications (2000), <http://www.fipa.org>
- [16] YaCy Project (2006), <http://www.yacyweb.de>



# Hybrid Immune-inspired Method for Selecting the Optimal or a Near-Optimal Service Composition

Ioan Salomie, Monica Vlad, Viorica Rozina Chifu, Cristina Bianca Pop  
Department of Computer Science  
Technical University of Cluj-Napoca  
26-28 Baritiu str., Cluj-Napoca, Romania  
Email: {Ioan.Salomie, Viorica.Chifu, Cristina.Pop}@cs.utcluj.ro

**Abstract**—The increasing interest in developing optimization techniques that provide the optimal or a near-optimal solution of a problem in an efficient way has determined researchers to turn their attention towards biology. It has been noticed that biology offers many clues regarding the design of such optimization techniques, since biological systems exhibit self-optimization and self-organization capabilities in a decentralized way without the existence of a central coordinator. In this context we propose a bio-inspired hybrid method that selects the optimal or a near-optimal solution in semantic Web service composition. The proposed method combines principles from immune-inspired, evolutionary, and neural computing to optimize the selection process in terms of execution time and explored search space. We model the search space as an Enhanced Planning Graph structure which encodes all the possible composition solutions for a given user request. To establish whether a solution is optimal, the *QoS* attributes of the services involved in the composition as well as the semantic similarity between them are considered as evaluation criteria. For the evaluation of the proposed selection method we have implemented an experimental prototype and carried out experiments on a set of scenarios from the trip planning domain.

## I. INTRODUCTION

WEB services are software components exposing atomic functionalities over the Internet that are often composed to address complex user requests. As there might be more than one service offering the same functionality the problem of manually composing Web services is unfeasible and automatic strategies need to be considered. The composition process focuses mainly on finding the appropriate services that composed satisfy the user functional requirements without considering the non-functional ones. These non-functional requirements are taken into consideration in the process of selecting the optimal composition of services. The large number of available services providing the same functionality leads to many composition solutions and the choice of the best one according to the functional and non-functional requirements becomes an optimization problem. Such an optimization problem requires appropriate selection strategies which provide the optimal or a near-optimal solution in a short time and without processing the entire search space. Generally, selection strategies fall into the category of exhaustive strategies or approximate ones. In the context of selecting the optimal Web service composition solution the most appropriate from the practical point of view are the

approximate ones. These methods guarantee to find an optimal or a near optimal solution in a time efficient manner. Meta-heuristics have emerged as a promising type of approximate algorithms as they represent algorithmic frameworks defining general-purpose concepts and strategies that can be easily adapted and used to solve various combinatorial optimization problems [3], such as Web service composition.

This paper presents how principles inspired from immune and evolutionary systems can be combined with reinforcement learning to design a hybrid method for selecting the optimal or a near-optimal composition solution. The search space for the hybrid method is represented by an Enhanced Planning Graph (EPG) which encodes all the possible composition solutions for a given user request. In our approach, a user request is described in terms of functional and non-functional requirements. The functional requirements are expressed using ontological concepts that semantically describe the inputs and outputs of the requested composed service. The non-functional requirements represent weights associated to user preferences regarding the relevance of a composition solutions semantic quality and its *QoS* attributes. To identify the optimal composition solution encoded in the EPG, we define a fitness function which uses as evaluation criteria both the *QoS* attributes and the semantic quality of the services involved in composition as opposed to other research approaches which focus only on *QoS*. The proposed selection method has been tested and validated on a set of scenarios from the trip planning domain.

The paper is structured as follows. Section II presents related work. In Section III we overview the formal model for representing the search space of our hybrid selection method. Section IV introduces the hybrid selection method, while its evaluation is discussed and analyzed in Section V. We end our paper with conclusions and future work proposals.

## II. RELATED WORK

This section reviews bio-inspired methods for selecting the optimal service composition solution available in the research literature.

In [1] authors present a genetic-based algorithm for binding concrete services to an abstract composition workflow. A genetic chromosome is mapped to a service composition solution encoded using an integer array representation. Each composition solution is evaluated using a fitness function



that considers only the  $QoS$  attributes as binding criteria. The crossover and mutation operators are used to ensure the evolution of the available service composition solutions towards the optimal or a near-optimal one. When applying the crossover and mutation operators authors are guided only by the dependency constraints between the services, which means that if a concrete service is assigned to an abstract task then all the abstract tasks that depend on the first one must have assigned only services from the same endpoint address [1]. The crossover and mutation points are chosen randomly. To select the service composition solutions that will be part of the next population, a roulette wheel selection operator is used. Authors introduce a re-binding algorithm used at runtime to re-evaluate the  $QoS$  score of a composition solution and if it is below a specified threshold then the services affecting the  $QoS$  score are replaced with other similar services which do not violate the dependency constraints.

A genetic-based algorithm for selecting the optimal or a near-optimal solution in multi-path Web service composition is proposed in [6]. A genetic chromosome is mapped to a service composition solution and each chromosome unit is represented as a triple containing the task context, workflow sign (specifies if the task is part of an AND/OR workflow) and the pointer towards the candidate service [6]. Initially, a set of service composition solutions having different topologies (corresponding to different paths) are randomly generated. Then the actual selection process is performed by iteratively (i) selecting the best solutions based on  $QoS$  attributes, (ii) applying a crossover operator between solutions with different topologies, (iii) and randomly mutating a randomly chosen service composition solution.

A hybrid method combining Particle Swarm Optimization (PSO) [7] with Simulated Annealing is proposed in [4] for selecting the optimal or a near-optimal service composition solution based on  $QoS$  attributes. Authors model service composition using an abstract workflow on which concrete services are mapped. A composition solution is considered as the position of a particle in PSO, while velocity is used to modify a composition solution. To avoid the problem of premature stagnation in a local optimal solution, a Simulated Annealing-based strategy is introduced which produces new composition solutions by randomly perturbing an initial solution.

Another hybrid method based on PSO is presented in [10] which introduces a non-uniform mutation strategy that aims to modify the global best optimal composition solution for ensuring diversity - i.e. the exploration of new areas of the search space. In addition, to improve the convergence speed the authors use an adaptive weight strategy for adjusting the particle velocity. A local best first strategy is used to replace the services having a low  $QoS$  score from a composition solution with others having a higher  $QoS$  score.

In [11], authors apply the Ant Colony Optimization (ACO) meta-heuristic to select the optimal or a near-optimal service composition solution based on  $QoS$  attributes. The service composition problem is modeled as an abstract workflow on

which concrete services are mapped similar to [1] and [4]. The resulting composition graph represents the search space for the proposed ACO-based selection algorithm. To identify the optimal values of the adjustable parameters introduced by the ACO-based selection algorithm, authors employ a genetic-based strategy.

The differences between our approach and the ones presented above are the following: (i) our composition graph is generated dynamically based on a user request and it does not start from a predefined workflow, (ii) our criteria for evaluating the quality of a candidate composition solution includes not only  $QoS$  attributes but also the property of semantic quality between the services involved in the composition solution, (iii) the replacement of a concrete candidate service is not done randomly, but according to  $QoS$  attributes and semantic quality. The chances of randomly choosing a certain service as a replacement are directly proportional to the quality of the solution obtained by using the respective service.

### III. FORMAL MODEL FOR REPRESENTING SEMANTIC WEB SERVICE COMPOSITION

The Web service composition has been modeled using an *Enhanced Planning Graph* (EPG) structure which we described in [8]. The EPG model is obtained by mapping the classical AI planning graph problem [9] to semantic Web service composition and also by enhancing the mapped concepts with the new abstractions of service cluster and parameter cluster. A service cluster groups services which provide the same functionality. The functionalities of the services belonging to the same cluster are annotated only with *is-a* related ontological concepts. For simplicity, we consider that a Web service has only one operation. A parameter cluster groups similar input and output service parameters annotated with *is-a* related ontological concepts. Consequently, the AI planning graph concepts are mapped to the Web service composition concepts as follows:

- An action becomes an ontology concept annotating a service operation.
- A precondition becomes an ontology concept annotating an input parameter of a service operation.
- An effect becomes an ontology concept annotating an output parameter of a service operation.
- The initial state becomes the set of ontology concepts semantically describing the user provided input parameters.
- The goal state becomes the set of ontology concepts semantically describing the user requested output parameters.

The EPG construction is an iterative process which operates at the semantic level by considering the ontology concepts that annotate the service functionality and the input/output parameters. At each step, a new layer  $i$  consisting of a tuple  $(A_i, L_i)$  is added to the graph where  $A_i$  represents a set of service clusters and  $L_i$  is a set of clusters of service input/output parameters. Layer 0 consists of a tuple  $(A_0, L_0)$  where  $A_0$  is an empty set of services (actions) and  $L_0$  contains the input parameters of the user request.

For each layer  $i > 0$ ,  $A_i$  consists of a set of clusters of services for which the input parameters are contained in  $L_{i-1}$ . The services which contribute in each step to the extension of the EPG are provided by a discovery process which finds the appropriate Web services in a repository of services, based on the semantic matching between the services' inputs and the set of parameters of the previous graph layer. The  $L_i$  set is built as a union of the  $L_{i-1}$  set and the set of the outputs of the services in  $A_i$ .

The construction of the EPG ends either when the user requested outputs are contained in the current set of parameters or when the graph reaches a fixed point. Reaching a fixed point means that the sets of services and parameters are the same for the last two consecutive generated layers.

A composition solution encoded in the EPG consists of a set of services, one from each cluster from each EPG layer.

#### IV. THE IMMUNE-INSPIRED HYBRID METHOD

This section presents how immune-inspired principles can be applied to the problem of selecting the optimal service composition solution as well as a hybrid selection algorithm. This algorithm adapts and enhances a version of the CLON-ALG algorithm [2] (proposed for general purpose optimization problems) by combining immune-inspired principles with evolutionary computing and reinforcement learning to ensure that the optimal or a near-optimal composition solution is obtained in a short time and without processing the entire search space.

##### A. Applying Immune-inspired Principles to Select the Optimal Composition Solution

Clonal selection is one of the most important processes of the immune system. It is triggered when a B-cell has high affinity to an invading pathogen (antigen presenting cell), and as a result, the B-cell is stimulated to clone itself. Through cloning, a number of identical copies of B-cells, proportional to the affinity value, are obtained. The number of copies is proportional to the affinity value. The clones are involved in an affinity maturation process which helps in improving their specificity to the invading pathogen by means of somatic hypermutation. The degree of mutation is inverse proportional to the affinity value, meaning that the clones having high affinity do not need to be mutated as much as the ones with low affinity. The affinity matured clones pass through new selection processes aiming at (i) keeping the clones having high affinity to the pathogen, and (ii) eliminating the clones with low affinity. The selected clones are then differentiated into memory cells and effector cells.

Generally, the biological clonal selection process is mapped to the Web service composition problem as follows: (i) a B-cell (or antibody) is represented by a service composition solution, (ii) a pathogen (or antigen) is represented by a function  $f$  that evaluates the set of composition solutions in order to find the optimal one in terms of  $QoS$  attributes, and (iii) the affinity between an antibody and an antigen is represented by the value of the function  $f$  for a composition solution [5][12].

In our approach, the pathogen (or antigen) is represented by the following multi-criteria function  $QF$  which evaluates a composition solution,  $sol$ , in terms of  $QoS$  and semantic quality:

$$QF(sol) = \frac{w_{QoS} * QoS(sol) + w_{Sem} * Sem(sol)}{(w_{QoS} + w_{Sem}) * |sol|} \quad (1)$$

where:

- $QoS(sol)$  [8] is the  $QoS$  score of the composition solution  $sol$ .
- $Sem(sol)$  [8] is the semantic quality score of the composition solution  $sol$ .
- $w_{QoS}$  and  $w_{Sem}$  are the weights corresponding to user preference related to the relevance of  $QoS$  and semantic quality.

In our case, the objective of applying the clonal selection principle is to obtain the optimal or a near-optimal solution which maximizes the  $QF$  function. Thus, the affinity between the antibody and antigen is considered to be the value of the  $QF$  function for a solution  $sol$ .

In our approach, we represent a composition solution by using a discrete type of genetic representation. Consequently, each Web service has associated an identification number in the format  $n_l n_c n_s$ , where  $n_l$  is the service's layer number in EPG,  $n_c$  is the number of the cluster to which the service belongs, and  $n_s$  is the service's identification number within the cluster. A solution will be a set of service identification numbers.

We model the processes of cloning and somatic hypermutation by keeping the current solution set and adding new different solutions as a result of a mutation process over the older ones. Customized to our problem, cloning implies duplicating for a number of times each solution in the solution set, while somatic hypermutation implies replacing some of the services part of a solution with other services from the same cluster, according to specific criteria. Thus the affinity maturation process is considered to enlarge the search space and to ensure diversity but in a controlled way such that only the mutated clones that are relevant to the problem to be solved are kept. Consequently, if the somatic hypermutated clone is not better in terms of the  $QF$  function than the parent solution (the original solution) it will be ignored (i.e. will die), otherwise it will be added to the solutions set (i.e. will survive).

##### B. The Immune-inspired Hybrid Algorithm

Our bio-inspired hybrid selection algorithm (see Algorithm 1) determines the optimal composition solution,  $sol_{opt}$ , according to the fitness function  $QF$  (see Formula 1) by processing the EPG. Besides the EPG graph, the algorithm takes as inputs the following adjustable parameters: (i) the number  $n$  of solutions that can be selected for the somatic hypermutation and affinity maturation process, (ii) the number  $m$  of the worst solutions that will be replaced by new randomly generated solutions, (iii) the number  $\beta$  that is used to decide how many clones should be generated for each solution in an iteration, (iv) the number  $noS$  of allowed stagnations, and (v) the restart

iteration number,  $r$ . The algorithm returns the optimal or a near-optimal composition solution.

---

**Algorithm 1:** Hybrid\_Selection
 

---

```

1 Input:  $EPG, n, m, \beta, noS, r$ 
2 Output:  $sol_{opt}$ 
3 Comments:  $M_L$  - learning memory,  $M$  - solution
  frequency memory,  $it_c$  - current iteration,  $f_{opt}$  -  $sol_{opt}$ 
  appearance frequency.
4 begin
5    $Sol = \text{Initialize\_Solution\_Set}(EPG)$ 
6    $sol_{opt} = \text{Max\_QF}(Sol)$ 
7    $M = \emptyset, f_{opt} = 1, it_c = 1$ 
8    $M = \text{Update\_Frequency\_Memory}(Sol, M)$ 
9   while ( $\text{!Stop\_Cond}(it_c, f_{opt}, noS, sol_{opt})$ ) do
10     $topN = \text{Calculate\_TopN}(Sol, n)$ 
11     $SelectedSols = \text{Select\_Solutions}(Sol, topN)$ 
12     $n_c = \text{Calculate\_Number\_of\_Clones}(\beta, topN)$ 
13     $Sol^* = \emptyset$ 
14    foreach  $sol$  in  $SelectedSols$  do
15       $Sol^* = Sol^* \cup \text{Generate\_Clones}(sol, n_c)$ 
16      foreach  $cl$  in  $Sol^*$  do
17         $cl = \text{Somatic\_HyperMut}(cl, M, M_L)$ 
18        if ( $sol \neq cl$  and  $QF(sol) < QF(cl)$ ) then
19           $Sol = Sol \cup \{cl\}$ 
20           $M_L = \text{Reward}(sol, cl, M_L)$ 
21          if ( $QF(sol_{opt}) \leq QF(cl)$ )
22            then  $\text{Replace\_Optim}(sol_{opt}, cl)$ 
23            else  $M_L = \text{Penalize}(sol, cl, M_L)$ 
24          end if
25        end for
26      end for
27       $lastR = \text{Calculate\_LastR}(m, topN, Sol)$ 
28       $Sol = \text{Sort}(Sol)$ 
29       $\text{Replace\_Solutions}(Sol, lastR, M)$ 
30      if ( $\text{Restart\_Condition}(it_c, r)$ ) then  $\text{Restart}(Sol, M_L)$ 
31       $f_{opt} = f_{opt} + 1, it_c = it_c + 1$ 
32    end while
33    return  $sol_{opt}$ 
34 end

```

---

The algorithm starts with an initialization stage (lines 5-8) in which two composition solutions,  $sol_1$  and  $sol_2$ , are randomly generated by processing the EPG. These solutions are added to the set of solutions  $Sol$  and the one that has the highest  $QF$  value is declared the current optimal solution,  $sol_{opt}$ . This solution will be considered as the model of the set and will not take part in any other processes. The aim of the other solutions in the set  $Sol$  is to become similar to the model. In addition, the appearance frequency of these two solutions is initialized with 1 and stored in a frequency memory. The frequency memory is used to record the appearance frequency of every generated composition solution.

Next, the following main steps are repeated until the stopping condition is satisfied (line 9), namely whether the number

of stagnations is equal to  $noS$ :

1) *Select and clone the topN best solutions:* The number  $topN$  of solutions that will be selected in order to be cloned and affinity matured is computed (line 10) according to the following formula:

$$topN = \begin{cases} |Sol| - 1, & \text{if } |Sol| - 1 < n \\ n, & \text{otherwise} \end{cases} \quad (2)$$

Afterwards, the first  $topN$  best solutions in the set  $Sol$  are selected (line 11) for further processing.

The number  $n_c$  of clones that will be generated for each selected solution is calculated (line 12) according to the following formula adapted from [2]:

$$n_c = \text{Round}(\beta * topN) \quad (3)$$

where  $\beta$  is a problem specific parameter which influences the selection algorithm's convergence speed towards the optimal solution. In our approach we have considered that  $\beta \in [0, 1]$  and its optimal value is determined experimentally.

Afterwards, for each selected solution, a number of  $n_c$  clones are generated (line 15).

2) *Somatic Hypermutation:* Each clone is passed through a somatic hypermutation process (line 17) in which a combined genetic operator is applied between the clone and the current optimal solution. We introduced a combined genetic operator in the selection method to ensure diversity by allowing the exploration of other possible solutions encoded in the EPG. This operator is applied between two composition solutions, one being the current optimal solution  $sol_{opt}$ , and the other one being the currently processed solution  $sol_i$ , and consists of the following processing steps: (1) the two solutions are compared service by service and as a result, a new solution is obtained which keeps the services that are common to both  $sol_{opt}$  and  $sol_i$  on one hand and keeps the services from  $sol_i$  which do not appear in  $sol_{opt}$  on the other hand, (2) the new solution is subject to a mutation process in which the services that belong only to  $sol_i$  are replaced with other services from the same cluster. In the mutation step it is important to constantly improve the current local optimum solution. This can be efficiently achieved by taking into account the result of previous decisions. For example, if once we have replaced a service  $s_1$  by another service  $s_2$  and obtained a better solution then it is obvious that by using this replacement in the future, better solutions will be obtained. This is the motivation that lead us to choose reinforcement learning as an important component of the bio-inspired hybrid model. In our case, we have adopted an active type of reinforcement learning.

In what follows, we present the reinforcement learning strategy applied for selecting the optimal Web service composition solution. First, we have defined a special data structure, called the *learning memory*,  $M_L$ , defined as follows:

$$M_L = \{m_l | m_l = ((s_i, s_j), reward)\} \quad (4)$$

where  $s_i$  is the service that will be replaced by  $s_j$  and  $reward$  is a numerical value used for recording awards and penalties

for the tuple  $(s_i, s_j)$ . Suppose  $sol$  is the current processed solution. After the affinity maturation process completes, some feedback information needs to be sent. If the clone  $sol_c$  of  $sol$  survives after the affinity maturation process, which means that the clone's affinity value is higher than the parent's affinity value, then the learning process starts and some rewards are given. Consequently, the structure of the mutated clone  $sol_c$  is compared against the structure of its parent  $sol$  at the cluster level to identify the  $sol$ 's services that need to be analyzed as follows: if a service  $s_j$  belonging to cluster  $C$  in  $sol_c$  is different than its adjacent service  $s_i$  from  $sol$ , and the service  $s_j$  positively affects the quality of  $sol$  compared to  $s_i$ , then it will be learnt that by replacing service  $s_i$  with service  $s_j$  a better solution is obtained and  $s_j$  will receive a reward. In other words, the pair  $(s_i, s_j)$  is added to  $M_L$  if it does not already exist and the reward associated to it will be 1, otherwise it will not be added to  $M_L$  and the associated reward will be incremented by 1. If  $sol_c$  will not survive after the affinity maturation process, then the learning process also starts, but this time some penalties will be considered. The solutions  $sol$  and  $sol_c$  are compared as was described above. If the pair  $(s_i, s_j)$  is stored in  $M_L$  then the reward associated to it will be decremented by 1. In our hybrid method, the learning memory  $M_L$  will be used in the somatic hypermutation process every time  $s_i$  needs to be mutated. In this process, a pair  $(s_i, s_j)$  is searched in  $M_L$  and if it is found then  $s_j$  will take the place of  $s_i$ . Otherwise  $s_j$  will be chosen from the  $s_i$ 's cluster, providing that  $sol_c$  will have a higher value of the affinity than  $sol$ .

3) *Update Sol and Replace lastR Worst Solutions from Sol*: If the new clone,  $cl$ , is different from its parent  $sol$  and  $cl$ 's  $QF$  value is better than  $sol$ 's  $QF$  value, then  $cl$  is added to the set of solutions and rewards are given (line 20) to those pairs of services that lead to a solution that is closer to the optimal one. Also, if  $cl$ 's  $QF$  value is greater than  $sol_{opt}$ 's  $QF$  value, then  $sol_{opt}$  is replaced with  $cl$ . In case the new clone is not better than its parent, then some penalties are applied.

After processing each solution selected for affinity maturation, the number  $lastR$  of worst solutions, in terms of the  $QF$  function, is computed (line 27) according to the formula:

$$lastR = \begin{cases} val_1, & \text{if } topN < n \\ val_2, & \text{otherwise} \end{cases} \quad (5)$$

In Formula 5,  $val_1$  and  $val_2$  are computed with Formulas 6 and 7, respectively:

$$val_1 = \begin{cases} 1, & \text{if } \frac{m*n}{topN} < 0.5 \\ Round(\frac{m*n}{topN}), & \text{otherwise} \end{cases} \quad (6)$$

$$val_1 = \begin{cases} |Sol| - 1, & \text{if } |Sol| - 1 \leq m \\ m, & \text{otherwise} \end{cases} \quad (7)$$

where  $n$  is the number of solutions that can be selected for the somatic hypermutation and affinity maturation process, and  $m$  is the number of the worst solutions that will be replaced by new randomly generated solutions.

Consequently, the set  $Sol$  is sorted (line 28) and the last  $lastR$  worst solutions in the set  $Sol$  will be replaced by other randomly generated solutions (line 29).

Formulas 5, 6 and 7 have been validated experimentally.

4) *Verify Restart Condition*: If the restart condition is true (line 30), namely if  $it_c$  is a multiple of  $r$ , then the algorithm is restarted (line 30). Consequently, the set of solutions  $Sol$  will be initialized with the first two solutions in the current set  $Sol$  and the learning memory content will be deleted.

The restart strategy works as follows. Let's consider a predefined parameter  $t$  called the restart iteration. Whenever the current iteration of the selection algorithm is a multiple of  $t$ , the algorithm is restarted in this way:

- The first two composition solutions of the initial set will not be randomly generated (as when starting the selection algorithm); they will be the current local optimal solution and the solution that follows after it (in terms of affinity values - see Formula 1).
- The learning memory  $M_L$  is deleted.

## V. EXPERIMENTAL RESULTS

We have evaluated the proposed selection method on a set of scenarios from the trip planning domain. The set of services used in our experiments was developed in-house and annotated according to the SAWSDL specification. A user request is specified by a set of ontological concepts semantically describing the provided inputs and requested outputs as well as by weights indicating the relevance of  $QoS$  related to the property of semantic quality. Services are semantically described with concepts annotating their functionality, the input and the output parameters. In what follows we present the experimental results obtained for the user request in Table I which aims to make the travel arrangements for a trip to Szczecin, Poland. The EPG for this user request is organized on 3 layers consisting of 51 services grouped in 11 clusters.

A challenge we faced while testing the proposed hybrid method was setting the optimal values of the following parameters:  $n$  - the number of composition solutions selected

TABLE I  
USER REQUEST FOR PLANNING A HOLIDAY

User inputs	Requested outputs	QoS weights	Semantic quality weight
SourceCity	AccommodationInvoice	Total QoS = 0.55	0.35
DestinationCity	FlightInvoice	Availability = 0.30	
StartDate	CarInvoice	Reliability = 0.30	
EndDate		Cost = 0.15	
HotelType		ResponseTime = 0.25	
NumberOfPersons			
NumberOfRooms			
CarType			
ActivityType			

to be cloned,  $m$  - the number of worst composition solutions to be replaced with randomly generated solutions,  $noS$  - the number of stagnations in a local optimal solution,  $r$  - restart iteration number,  $\beta$  - mutation rate.

The methodology used for establishing the optimal configuration (i.e. a 5-tuple containing a value for each adjustable parameter) of the adjustable parameters consists of three steps. In the first step, an exhaustive search is performed to identify the score of the optimal composition solution which is 6.8 for the considered scenario. This score is further used to identify the most appropriate configuration of the adjustable parameters which ensures that the optimal or a near-optimal composition solution is obtained without processing the entire search space. In the second step, we chose the following initial configuration of the adjustable parameters based on some suppositions:  $\beta = 0.25$ ,  $r = 11$ ,  $n = m = 4$  and  $noS = 33$ . In the third step, the initial configuration of the adjustable parameters is fine-tuned iteratively to identify the optimal configuration. For the considered scenario we have chosen 43 configurations and for each one we have performed 100 runs of the selection algorithm analyzing the following aspects: the global optimal solution frequency, the number of iterations, the number of generated distinct solutions, the execution time measured in seconds, and the standard deviation. In Table II we present a fragment of the average experimental values obtained for each configuration, where: (i)  $f_{opt_{av}}$  is the average global optimal solution frequency, (ii)  $noIt_{av}$  is the average number of iterations, (iii)  $noSG_{av}$  is the average number of generated distinct solutions, (iv)  $tEx_{av}$  is the average execution time measured in seconds, and (v)  $stD_{av}$  is the standard deviation.

By analyzing the experimental results from Table II we reached the conclusion that the optimal configuration of the adjustable parameters is  $n = 7$ ,  $m = 8$ ,  $\beta = 0.5$ ,  $r = 6$ ,  $noS = 24$ . We chose these values because they ensure a tradeoff between obtaining the optimal solution and a very low execution time.

In the final stage of the hybrid algorithm's evaluation we performed 2100 simulations on the scenario presented in Table I and by considering the optimal configuration of the adjustable parameters. In our experiments we focused on the following issues: the average number of processed solutions, the average percent of explored search space, the average simulation time, the average number of cases in which the optimal solution has been obtained, and the standard deviation of the score of the best solution returned by the algorithm related to the score of the global optimal composition solution for the considered scenario. By analyzing the experimental results we conclude that:

- The hybrid selection algorithm explores approximately 0.001% of the search space (i.e. approximately 205 distinct composition solutions explored out of 13996800).
- The average execution time is about 27 seconds.
- The hybrid selection algorithm returns the global optimal solution in 95% of the simulations (i.e. 2015 simulations

TABLE II  
TESTS SUMMARY

No.	$n$	$m$	$\beta$	$r$	$noS$	$f_{opt_{av}}$	$noIt_{av}$	$noSG_{av}$	$tEx_{av}$	$stD_{av}$
1	4	4	0.25	11	33	0.95	54	207	19	0.042
4	4	4	0.5	11	44	0.96	70	271	48	0.031
3	4	4	0.25	11	44	0.96	68	261	32	0.028
7	6	6	0.25	11	44	0.97	66	351	36	0.027
8	6	6	0.25	11	33	0.91	54	287	28	0.043
9	6	6	0.5	11	33	0.89	54	286	34	0.065
11	8	8	0.25	22	11	0.23	18	134	9	0.175
12	8	8	0.25	22	22	0.55	38	272	18	0.128
13	8	8	0.5	11	22	0.87	41	266	30	0.056
15	8	8	0.5	11	44	0.91	66	439	50	0.052
17	8	8	0.5	11	33	0.96	53	352	39	0.035
18	6	4	0.5	11	33	0.96	56	205	39	0.031
19	6	4	0.5	11	22	0.8	42	156	29	0.079
21	6	4	0.75	11	33	0.93	55	202	52	0.039
23	6	4	0.75	11	11	0.51	23	82	23	0.155
24	8	6	0.75	11	11	0.57	23	114	23	0.131
26	6	5	0.75	11	33	0.93	56	245	51	0.049
27	6	5	0.75	11	22	0.79	39	174	37	0.078
28	6	5	0.75	11	11	0.6	24	106	25	0.121
30	6	5	0.75	6	12	0.86	25	97	24	0.076
34	7	8	0.5	11	33	0.93	51	335	37	0.038
35	7	8	0.5	11	22	0.85	38	245	27	0.083
37	7	8	0.5	6	18	0.93	33	173	23	0.038
41	5	6	0.75	11	33	0.87	53	281	38	0.060
42	4	6	0.75	11	44	0.93	70	380	37	0.040
43	7	8	0.5	6	24	0.95	39	204	28	0.033

out of 2100 simulations).

- The average standard deviation on all 2100 simulations of the hybrid selection algorithm is 0.0344.

## VI. CONCLUSIONS AND FUTURE WORK

This paper proposed a hybrid method for selecting the optimal or a near-optimal solution in semantic Web service composition. The method was designed according to principles from the clonal selection process which occurs in biological immune systems. To increase the algorithm's convergence speed towards the optimal solution as well as to avoid the problem of stagnation in a local optimum we introduce a combined genetic operator which generates new better composition solutions by querying a learning memory. The learning memory stores a history of service replacements as well as a score which indicates which replacement can lead to a new better service composition solution.

The hybrid method has been tested on a set of scenarios from the trip planning domain and its performance has been evaluated according to the following criteria: the number of processed solutions, the percent of explored search space, the simulation time, the number of cases in which the optimal solution has been obtained, and the standard deviation.

As future work we intend to comparatively analyze the hybrid method with other selection methods using the same test scenarios.

#### REFERENCES

- [1] G. Canfora, M. Di Penta, R. Esposito, M. L. Villani, *A Framework for QoS-Aware Binding and Re-Binding of Composite Web Services*, Journal of Systems and Software, Volume 81, Issue 10, pp. 1754-1769, 2008.
- [2] L. Castro, F. von Zuben, *Learning and Optimization using the Clonal Selection Principle*, IEEE Transactions on Evolutionary Computation, Volume 6, Number 3, pp. 239-251, 2002.
- [3] M. Dorigo, M. Birattari, Thomas Stutzle, *Ant Colony Optimization - Artificial Ants as a Computational Intelligence Technique*, IEEE Computational Intelligence Magazine, Volume 1, Number 4, pp. 28-39, 2006.
- [4] X. Fan, X. Fang, *On Optimal Decision for QoS-Aware Composite Service Selection*, Information Technology Journal, Volume 9, Issue 6, pp. 1207-1211, 2010.
- [5] G. Yan, N. Jun, Z. Bin, Y. Lei, G. Qiang, D. Yu, *Immune Algorithm for Selecting Optimum Services in Web Service Composition*, Wuhan University Journal of Natural Sciences, Volume 11, Number 1, pp. 221-225, 2006.
- [6] H. Jiang, X. Yang, K. Yin, S. Zhang, J. A. Cristoforo *Multi-path QoS-aware Web Service Composition using Variable Length Chromosome Genetic Algorithm*, Information Technology Journal, Volume 10, Issue 1, pp. 113-119, 2011.
- [7] J. Kennedy, R. C. Eberhart, *Particle Swarm Optimization*, Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ. pp. 1942-1948, 1995.
- [8] C. B. Pop, V. R. Chifu, I. Salomie, M. Dinsoreanu, *Immune-Inspired Method for Selecting the Optimal Solution in Web Service Composition*, LNCS, Volume 6162/2010, pp. 1-17, 2010.
- [9] S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, Upper Saddle River, NJ, Prentice Hall/Pearson Education, ISBN: 0137903952, 2003.
- [10] W. Wang, Q. Sun, X. Zhao, F. Yang, *An improved Particle Swarm Optimization Algorithm for QoS-aware Web Service Selection in Service Oriented Communication*, International Journal of Computational Intelligence Systems, Volume 3, Supplement 1, pp. 18 - 30, 2010.
- [11] Z. Yang, C. Shang, Q. Liu, C. Zhao, *A Dynamic Web Services Composition Algorithm Based on the Combination of Ant Colony Algorithm and Genetic Algorithm*, Journal of Computational Information Systems, Volume 6, Issue 8, pp. 2617-2622, 2010.
- [12] J. Xu, S. Reiff-Marganiec, *Towards Heuristic Web Services Composition Using Immune Algorithm*, Proceedings of the International Conference on Web Services, Beijing, China, pp. 238-245, 2008.





# Dynamic Consolidation Methodology for Optimizing the Energy Consumption in Large Virtualized Service Centers

Tudor Cioara, Ionut Anghel, Ioan Salomie,  
Daniel Moldovan, Georgiana Copil  
Technical University of Cluj-Napoca  
Cluj-Napoca, Romania  
{tudor.cioara, ionut.anghel, ioan.salomie,  
daniel.moldovan, georgiana.copil}@cs.utcluj.ro

Pierluigi Plebani  
Politecnico di Milano  
Milano, Italy  
plebani@elet.polimi.it

□ *Abstract*—In this paper we approach the high energy consumption problem of large virtualized service centers by proposing a dynamic server consolidation methodology for optimizing the service center IT computing resources usage. The consolidation methodology is based on logically structuring the service center servers hierarchical clusters, consolidation decisions being taken in each cluster using a reinforcement learning based algorithm. The methodology defines two ways of consolidation decisions propagation across the hierarchy: bottom-up propagation for the dynamic power management actions and top-down propagation for the consolidation actions. The consolidation decision time complexity analysis shows that the methodology usage in large service centers improves the decision time with a factor proportional with the ratio between the service center total number of servers and the logical clusters' number of servers.

*Keywords*—large service centers, dynamic server consolidation, reinforcement learning, energy consumption, hierarchical clusters.

## I. INTRODUCTION

OVER the last years the energy efficiency management of service centers has emerged as one of the most critical environmental challenges to be dealt with. A U.S. Environmental Protection Agency report to Congress [1] shows that in just five years, the electricity consumed by service centers and their additional infrastructure will double and the trend is expected to accelerate driven by their shift towards cloud computing. One of the major sources of the service centers huge amount of consumed energy is the inefficient utilization of IT computing resources. The IT computing resources in today's service centers are under-used, usually operating below the optimal loads for energy efficiency. According to [2] in a service center about 30% of servers consume energy without doing any actual work. In large service centers with thousands of servers, the computing resources average utilization ratio is between 5 and 10 percent providing a huge opportunity for organizations to reduce the service center energy consumption. A state of the art technique for providing an

optimal energy performance trade-off in service centers is resource consolidation using virtualization.

In this paper the dynamic server consolidation of large service centers is approached by using a hierarchy structure to logically organize the service center in clusters. Each logical cluster is being managed by its own instance of a reinforcement learning based consolidation algorithm presented in one of our previous papers [14]. The a reinforcement learning based consolidation algorithm was proven to be effective for increasing the service centers energy efficiency but its main drawback is the fact that the decision time complexity increases with the number of servers and become unsatisfactory for medium and large service centers. We have showed that reinforcement learning consolidation solution complexity for a large number of servers decrease when the service center is organized according to our hierarchical structure. The consolidation decision time improves with a factor proportional with the ratio between the service center total number of servers and the logical clusters' number of servers.

The rest of the paper is structured as follows: Section II presents the state of the art for dynamic consolidation, Section III presents an overview of the reinforcement learning based consolidation algorithm underlying the consolidation decision time problem for large service centers, Section IV introduces the proposed consolidation methodology based on logically organizing the service center in hierarchical clusters, Section V presents an analysis of the consolidation methodology decision time complexity, while Section VI concludes the paper.

## II. RELATED WORK

Resource consolidation (or server consolidation) in service centers aims at combining the virtualized workloads that are executed on different machines (servers) for obtaining an optimal number of computing resources usage [3]. As shown in [4], the greatest challenge for consolidation methods is deciding which workloads should be combined on a common physical server since resource usage, performance and energy consumption are not additive.

A service center may be classified taking into account the number of servers as [5]: (i) small service centers having a number of servers between 101 and 500, (ii) medium service

□ This work has been done in the context of the EU FP7 GAMES project (<http://www.green-datacenters.eu/>).

centers having between 501 and 5000 servers and (iii) large service centers with a number of servers usually over 5000. Many state of the art solutions regarding service center servers consolidation approach the energy consumption optimization through resource allocation or consolidation.

A thermal aware workload scheduling and consolidation solution aiming to reduce the power consumption and temperatures in data centers was proposed in [6]. The simulation results show that the algorithm can significantly reduce the energy consumption with some degree of performance loss. In [7] a novel technique for controlling the service centers servers CPU allocation and consolidation based on first order Kalman filter is presented. In [8] the server consolidation problem is approached for small service centers as a constraint satisfaction problem. The authors also propose a heuristic for approaching the server consolidation in large service centers. In [9], the authors propose an algorithm for consolidating virtual machines in large service centers based on a simple gossip protocol. To enable energy efficient consolidation, the inter-relationships between energy consumption, resource utilization, and performance of consolidated workloads must be considered [10]. In [11] the authors reveal that energy performance trade-offs for consolidation and optimal operating points exist. A bio-inspired workload consolidation algorithm for service centers based on defining some autonomous scouting entities is defined in [12]. The entities try to find the suitable server for migrating a virtual machine (worker entity).

Optimal computing resources allocation techniques for server clusters based on reinforcement learning are proposed in [16]. Learning techniques are also used to trade-off between computing resources power consumption and performance during the allocation process [17]. In [13] a consolidation methodology that uses machine learning to deal with uncertain information is discussed. Previous server behavior data is used to predict and estimate the current power consumption and also to improve the scheduling and consolidation decisions.

The presented state of the art approaches fail to consider the scalability problem when varying the service center dimension (number of servers) and applying different consolidation algorithms.

### III. REINFORCEMENT LEARNING BASED DYNAMIC SERVER CONSOLIDATION

In a previous published paper [14] we have approached the problem of dynamic server consolidation in virtualized service centers by proposing the development of an energy aware run-time consolidation algorithm based on reinforcement learning. To make this paper self-contained in Section A we present a short overview of the reinforcement learning based consolidation algorithm. More details can be found in [14]. Also the reinforcement learning consolidation decision time problem statement for large service centers is described in Section B.

#### A. Consolidation Algorithm Overview

The reinforcement learning consolidation algorithm has three main phases: (i) representing the service center energy related context data in a programmatic manner, (ii) calculating the service center greenness level and (iii) deciding on the consolidation actions that must be executed to bring the service center in an energy efficient state.

##### 1) Context data representation

To represent the energy related context data in a programmatic manner we have defined an ontology based context model: the EACM (Energy Aware Context Model) model [15]. The energy related context data is represented in the EACM model using three main concepts: Context Resources, Context Actions and Context Policies.

Context Resources define the physical or virtual entities that generate and / or process energy related context data. For a service center we have identified three sub-types of Context Resources: Facility Resources, Computing Resources and Application Resources. Facility Resources are physical entities which capture the service center ambient data (sensors) and enforce the design time defined environmental conditions (actuators). Computing Resources are physical entities that consume energy as a result of executing workload. The main Computing Resource of a service center considered in our representation is the server. Application Resources are the software entities executed on the service center computing resources as incoming workload. An activity is modeled through its processor, memory and hard disk computing resources requests.

Context Actions define the set of design time enabled adaptation actions that may be executed at run time to enforce the service center energy efficiency goals. We have identified three sub-types of adaptation actions: Facility Adaptation Actions (e.g. adjust the room temperature or start the air conditioner), IT Computing Adaptation Actions and Application Adaptation Actions (e.g. application redesign for energy efficiency). We have defined two main sub-classes of IT Computing Adaptation Actions: resource consolidation actions (Deploy Activity, Migrate Activity) and dynamic power management actions (Wake-up server, Turn-off server).

Context Policies define the service center energy efficiency goals through a design time established set of Green and Key Performance Indicators (GPIs/KPIs). We have defined three sub-classes of GPIs/KPIs: (1) environmental, imposing restrictions about the service centre ambient conditions (e.g. the temperature in the service center must be under 21°C), (2) IT Computing, describing the energy/performance characteristics of the service centre computing resources (e.g. the server CPU is efficiently used for a load between 60%-80%) and (3) Application, specifying the rules (QoS requests) imposed by the business application for execution (e.g. for optimal execution time the application needs 1Gb of allocated physical memory).

## 2) Service center greenness level

To calculate the service center greenness level we have defined the concept of service center context situation entropy ( $E_S$ ) [15]. The entropy is a metric which establishes the service center context situation degree of complying with the design time defined set of GPIs/KPIs. The GPIs/KPIs are represented in SWRL (Semantic Web Rule Language) reasoning rules and automatically evaluated against the EACM model instance ontology implementation. The entropy value of a service center context situation ( $S$ ) is calculated using the following relation:

$$E_S = \sum_i w_{p_i} \sum_j w_{r_{ij}} * v_{r_{ij}} \quad (1)$$

where: (i)  $w_{p_i}$  is the weight of GPIs/KPIs policy  $i$ , and represents the importance of the policy in the service centre context, (ii)  $w_{r_{ij}}$  is the weight of the service centre context resource  $j$  in the GPIs/KPIs policy  $i$  and reflects the context resource importance for that policy and (iii)  $v_{r_{ij}}$  is the deviation between the value recorded by the context resource  $j$  and the accepted value defined by policy  $i$  (if  $x$  is the accepted value of the context resource  $j$  defined by the GPI/KPI policy  $i$  and the actual value recorded by the resource  $j$  is  $r_{ij}$ , then  $v_{r_{ij}} = r_{ij} - x$ ).

The entropy value is used to trigger the consolidation process as follows: if the current service center context situation entropy value is above a predefined threshold, the service center greenness level is acceptable and consolidation is not required, otherwise the reinforcement learning consolidation process is started.

## 3) Consolidation actions selection

To decide on the consolidation actions that have to be executed if the service center is not in an energy efficient state a reinforcement learning based solution is used (see Fig. 2 for the algorithm pseudo-code).

The consolidation process starts from the current service center context situation, simulates the execution of all available consolidation (Deploy or Migrate activity) or dynamic power management actions (Turn-on or Turn-off server) based on a reward / penalty approach and builds a decision tree (see Fig. 1).

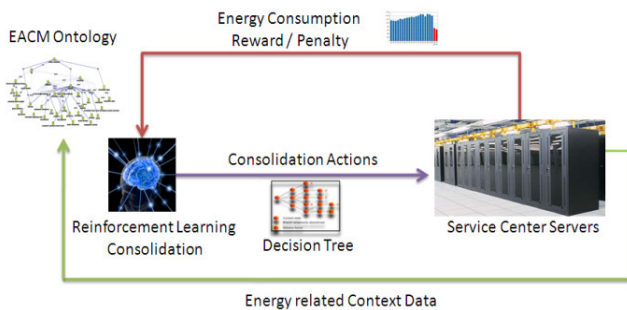


Fig. 1 The reinforcement learning consolidation decision process

A decision tree node stores: (i) the EACM instance describing the service center energy related context situation, (ii) the list of actions that were simulated to generate that EACM instance, (iii) the EACM instance calculated entropy value (relation 1) and (iv) the reward value calculated for the list of actions simulated so far. The reward for executing an action in a certain situation is calculated as follows:

$$R_{S+1} = R_S + \gamma * (E_{S+1} - E_S - ActionCost) \quad (2)$$

where: (i)  $R_{S+1}$  represents the reward for the newly generated (current) tree leaf node  $S+1$ , (ii)  $R_S$  represents the reward of the current leaf node parent, (iii)  $E_{S+1}$  and  $E_S$  represent the calculated entropy values for the EACM instance stored in the leaf node  $S+1$  and its parent  $S$ , while (iv)  $ActionCost$  represents an associated design time consolidation and dynamic power management action cost value.

```

1  Input: pQueue – a priority queue containing the current step reinforcement learning
2         tree leaf nodes sorted by their rewards
3         highestRewardNode – the reinforcement learning tree node with the highest reward
4  Output: TreeNode – the reinforcement learning tree node containing the sequence of
5         consolidation actions that should be executed to bring the service center
6         in a energy efficiency state and its associated reward
7
8  TreeNode reinforcementLearningConsolidation (PriorityQueue <TreeNode> pQueue,
9         TreeNode highestRewardNode)
10 begin
11   TreeNode currentLeaf = pop (pQueue)
12   if (currentLeaf == NULL) then return highestRewardNode
13   if (getEntropy(currentLeaf) < TE) then return currentLeaf
14   if ((highestRewardNode == NULL)
15       or (getReward(currentLeaf) > getReward (highestRewardNode))) then
16     highestRewardNode = currentLeaf
17   else
18     brokenGPI_KPI_Policies = getBrokenGPI_KPI_Policies (currentLeaf)
19     Action = NULL
20     foreach policy in brokenGPI_KPI_Policies
21       if (getSubject(policy) instanceof ApplicationActivity) then
22         activityInstance = getSubject(policy)
23         foreach server in sortServersByDistanceToActivity(currentLeaf, activityInstance)
24           if (hasResourcesFor(server, activityInstance) and notRunning(activityInstance)) then
25             set Action to (DEPLOY activityInstance on server)
26           foreach server in getTurnedOffServers (currentLeaf)
27             if (hasResourcesFor(server, activityInstance)) then
28               set Action to (TURNON server)
29       if (getSubject(policy) instanceof ComputingResource) then
30         serverInstance = getSubject(policy)
31         foreach activity in getRunningActivities(serverInstance)
32           foreach server in sortServersByDistanceToActivity(currentLeaf, activity)
33             if (hasResources(server, activity)) then
34               set Action to (MIGRATE activity from serverInstance to server)
35           if (getRunningActivities(serverInstance) == NULL) then
36             set Action to (TURNOFF server)
37       TreeNode nextLeaf = genNextLeaf (Action)
38       addNewLeafNode(pQueue, nextLeaf)
39   return highestRewardNode (pQueue, highestRewardNode)
40 end

```

Fig. 2 The reinforcement learning consolidation algorithm

A tree path between two nodes  $Node_0$  and  $Node_n$  defines the sequence of actions that executed starting from  $Node_0$  service center context situation generates the new service center context situation stored by node  $Node_n$ . The maximum reward path in the tree represents the sequence of actions that must be executed for consolidating the service center servers.

### B. Consolidation Decision Time Problem

The dynamic reinforcement learning process takes consolidation decisions in reasonable time frames (less than 85 seconds for 50 virtual activities to be consolidated) for small service centers (100 servers). When dealing with medium and large service centers (no. servers higher than 500), the consolidation decision time grows exponentially (see Fig. 3).

To evaluate the consolidation process decision time we have simulated a service center with varying numbers of physical servers on which the workload virtualized tasks must be deployed. To ease the estimations, we have considered homogenous service centers with identical server hardware resources configuration (1 CPU with 8 cores x 3000 MHz and 6000 MB Memory). The workload tasks that need to be deployed are also homogenous (a task's hardware request is: 8 CPU Cores with 500 MHz frequency and a 900 MB amount of memory).

Fig. 3 chart shows the results of the consolidation process decision time evaluation. The decision time grows exponentially with the service center number of servers and the workload number of tasks that have to be deployed. Analyzing the results it can be seen that for a service center with around 1000 servers the consolidation decision process time, involving the deployment of 200 tasks, is over 2500 seconds (about 40 minutes). In case of dynamic consolidation which usually involves extremely dynamic workload, this time is unsatisfactory.



Fig. 3 The reinforcement learning consolidation decision time

Taking into account the above presented time results for medium and large service centers, in this paper we propose a methodology for reducing the decision time by logically organizing the service center servers using a hierarchical clusters structure. Each structure element contains a cluster of service center computational resources (servers or other clusters) managed by its one instance of the reinforcement learning consolidation algorithm.

### IV. METHODOLOGY FOR DYNAMIC SERVER CONSOLIDATION IN LARGE SERVICE CENTERS

To solve the reinforcement learning consolidation decision time problem in large service centers, we propose a consolidation methodology based on logical clustering the service center in a hierarchical manner and associating to each cluster a specific reinforcement learning algorithm instance (see Fig. 4).

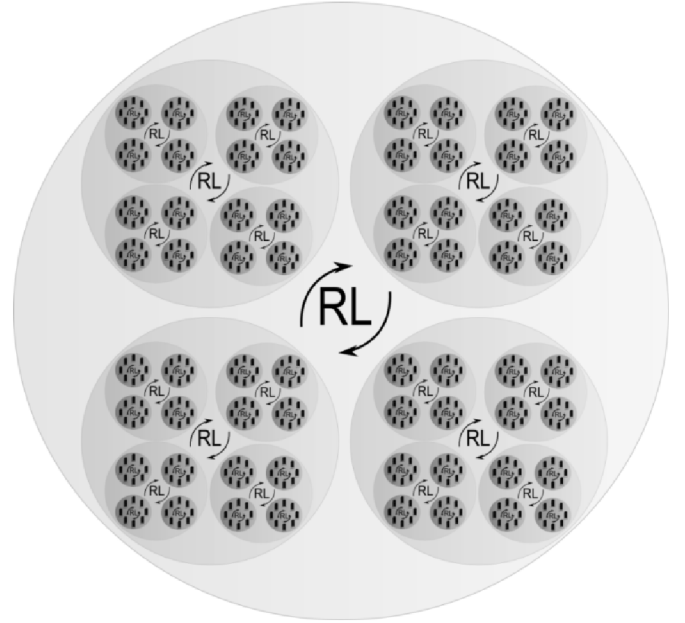


Fig. 4 Logical structuring of the service center servers using hierarchical clusters

#### A. Service Center Hierarchical Clusters Structure

The bottom layer of the hierarchical structure (level 0) is composed of service center physical servers. The server is the basic service center computational resource atomic granule for which the consolidation actions are considered. On the next hierarchical layer (level 1), the bottom layer servers are grouped into logical clusters, each cluster being managed by its own new instance of the reinforcement learning consolidation algorithm.

**Definition 1.** A logical cluster is a level 1 cluster if and only if it groups two or more physical servers (level 0 computational resources) that will be managed by the same instance of the reinforcement learning consolidation algorithm.

$$\begin{aligned}
 (c^{L1}[S_k, RL] \text{ is a level 1 cluster}) \leftrightarrow \\
 (\forall s_k \in S_k, s_k \text{ is a server}) \text{ and } (2 \leq ||S_k|| \leq MAX) \text{ and} \\
 (\exists RL(S_k) \text{ such that } RL \text{ manages } S_k)
 \end{aligned} \quad (3)$$

Level 1 clusters are recursively grouped into higher layer logical clusters on level 2, each obtained cluster being managed by its one reinforcement learning algorithm instance.

**Definition 2.** A logical cluster is a level 2 cluster if and only if it groups two or more level 1 logical clusters that will



be managed by the same instance of the reinforcement learning consolidation algorithm (see Fig. 4).

$$\begin{aligned}
 &(c^{L^2}[C_k^{L^1}, RL] \text{ is a level 2 cluster}) \leftrightarrow \\
 &(\forall (c_k^{L^1} \in C_k^{L^1}, c_k^{L^1} \text{ is a level 1 cluster}) \text{ and} \\
 &(2 \leq |C_k^{L^1}| \leq MAX) \text{ and} \\
 &(\exists RL(C_k^{L^1}) \text{ such that RL manages } C_k^{L^1})) \quad (4)
 \end{aligned}$$

To generalize, we can state that a level n cluster ( $0 < n < \text{TopMostLevel} - \text{level 0}$  and the top most level of the hierarchical structure does not fit in this definition) can be defined as follows:

$$\begin{aligned}
 &(c^{Ln}[C_k^{Ln-1}, RL] \text{ is a level n cluster}) \leftrightarrow \\
 &(\forall (c_k^{Ln-1} \in C_k^{Ln-1}, c_k^{Ln-1} \text{ is a level n - 1 cluster}) \text{ and} \\
 &(2 \leq |C_k^{Ln-1}| \leq MAX) \text{ and} \\
 &(\exists RL(C_k^{Ln-1}) \text{ such that RL manages } C_k^{Ln-1})) \quad (5)
 \end{aligned}$$

A logical cluster is a level n cluster if and only if it groups level n-1 clusters that will be managed by the same instance of the reinforcement learning consolidation algorithm.

**Definition 3.** A logical cluster is the top most cluster of the hierarchical structure (also called meta cluster) if and only if it logically groups all the clusters defined on the layer below it. On the topmost level of the hierarchy it must exist a single meta cluster.

*B. Consolidation Decision Propagation in the Hierarchy*

The reinforcement learning consolidation algorithm instances decisions are propagated across the service center logical hierarchical structure in two manners: (i) top-down, for decisions implying the execution of consolidation actions (deploy or migrate activity) and (ii) bottom-up for decisions implying the execution of dynamic power management actions (turn-on and turn-off server).

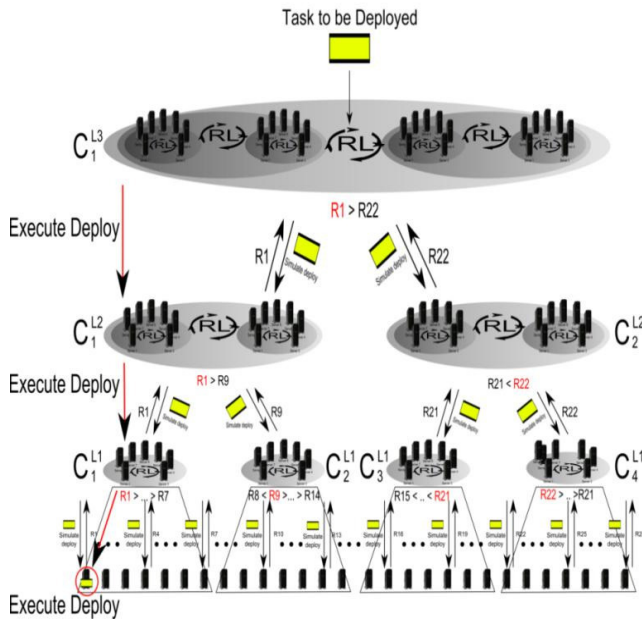


Fig. 5 The deploy action propagation example

The *deploy* activity decision is taken only by the meta cluster which receives the workload that the service center must execute (see Fig. 5). The decision and its associated activity is propagated to all the reinforcement learning algorithm instances controlling the inferior layer logical clusters. Each algorithm instance will simulate the activity deployment on the resources that it controls, calculates the associated reward and propagates the decision to the logical cluster algorithm instances below it. This propagation process continues recursively until the bottom layer is reached. The activity will be deployed on the server which is the leaf of the hierarchical structure path with the maximum reward.

The *migrate* activity decision (from one cluster to another) can only be taken by the reinforcement learning algorithm controlling both logical clusters (see Fig. 6). The migrate decision is also propagated in top-down manner as follows (see Fig. 6): (i) in the hierarchical structure sub-tree having as root the source logical cluster, a *destroy* activity action is propagated and (ii) in the hierarchical structure sub-tree having as root the destination logical cluster a *deploy* activity action is propagated. The destroy activity propagation is similar with the pattern described for deploying a task.

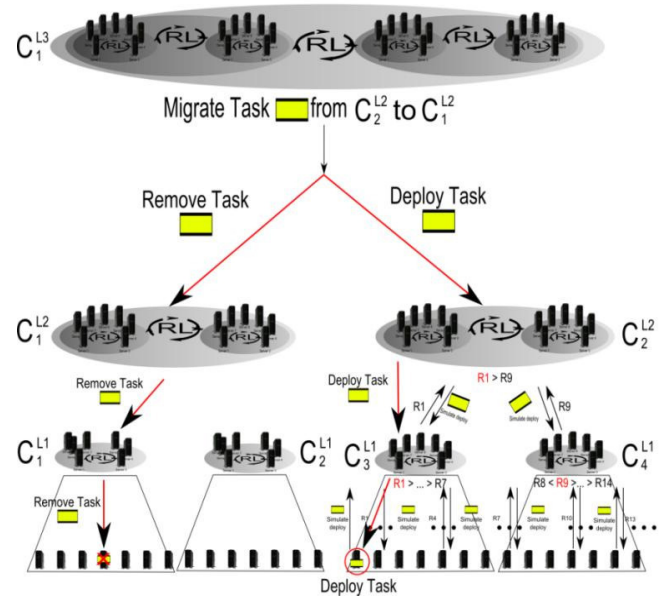


Fig. 6 The migrate activity action propagation example

The *turn-on* and *off* server / cluster actions are propagated across the hierarchical structure in a bottom-up manner. The decision is taken locally by the reinforcement learning consolidation algorithm that controls the computing resources that are turned-on or off. The decision is then signaled to the reinforcement learning algorithm structure controlling the upper level logical cluster which in turn investigates the possibility of turning-on / off the entire cluster containing the inferior level resources which were turned-on or off.

## V. CONSOLIDATION TIME COMPLEXITY ANALYSIS

In this chapter the consolidation decision time complexity is being estimated for a large service center with  $N$  servers ( $N > 5000$ ) that needs to accommodate  $M$  virtualized activities in an energy efficient manner. Two different cases are considered: (i) the service center has no logical organization and (ii) the service center is logical organized using the proposed hierarchical clusters structure described in Section IV.

In the first case, the reinforcement learning algorithm considers all the service center servers and virtual tasks when a consolidation decision needs to be taken. As it can be noticed from Section III, the reinforcement learning consolidation algorithm constructs a decision tree and searches for the best sequence of consolidation actions to be taken, in a certain situation, using a depth first search algorithm. The algorithm complexity is usually expressed as  $B^D$  where  $B$  represents the decision tree branching factor while  $D$  is the depth factor. The learning tree branching factor and the depth factor are equal to the number consolidation / dynamic power management actions that the reinforcement learning algorithm can consider at each step.

In the worst case scenario this number is given by the sum of (see relation 6): (i) the number of possible “turn-on server” actions (in the worst case scenario is equal with the number of turned-off servers), (ii) the number of possible “turn-off server” actions (in the worst case scenario is equal with the number of turned-on servers), (iii) the number of possible “deploy activity” actions (the number of undeployed tasks multiplied with the number of available servers for the worst case scenario), and (iv) the number of possible “migrate activity” actions (the number of already deployed activities multiplied with the number of up and running servers in the worst case scenario).

$$B = D = nrTurnedOffServers + nrTurnedOnServers + nrUndeployedActivities * nrTurnedOnServers + nrDeployedActivities * nrTurnedOnServers \quad (6)$$

By grouping and factoring relation 6 elements, the branching and depth factors can be also calculated using the following relation

$$B = D = N + M * nrTurnedOnServers \leq N + M * N \quad (7)$$

where  $N$  is the service center total number of servers while  $M$  is the total number of service center virtualized activities considered for consolidation.

Therefore the consolidation time decision complexity for a service center with  $N$  servers that needs to accommodate  $M$  virtualized tasks in the worst case scenario is:

$$((N + M * N)^{(N+M*N)}) = O((M * N)^{(M*N)}) \quad (8)$$

In the second case, when the service center is logically organized by using the proposed hierarchical cluster

structure, there will be multiple reinforcement learning consolidation algorithms that are executed on logical clusters with a smaller number of computational resources. For simplicity, we consider that the hierarchical structure logical clusters are uniformly created with the same number of computational resources  $c$  and the service center total number of servers  $N$  can be expressed as  $c^{kmax}$  (where  $kmax$  is to hierarchy total number of layers). In this case at each hierarchical structure level, there will be a number of  $N/c^k$  clusters where  $N$  is the service center total number of servers and  $k$  is the level number. Using the hierarchical structure meta cluster definition which states that on the top-most level of the hierarchy a single cluster may exist, we can compute the maximum number of hierarchical levels as:

$$\frac{N}{c^{kmax}} = 1 \rightarrow kmax = [\log_c N] \quad (9)$$

The reinforcement learning algorithm complexity for a cluster is  $O(Mc)^{(Mc)}$  where  $c$  is the number of computational resources from a cluster and  $M$  is the number of activities considered in the consolidation decisions. But since the consolidation decisions taken by a reinforcement learning algorithm instance are propagated in the hierarchical structure sub-tree under it, we can state that  $M=I$  for all the hierarchical reinforcement learning algorithm instances except the one taking the actual consolidation decision. The overall consolidation decision time complexity is:

$$O(M * c)^{(M*c)} + (M - 1) * kmax * O(c^c) = O(M * c)^{(M*c)} \quad (10)$$

Considering relations 8 and 10, we can state that using the proposed methodology the consolidation decision time complexity remains exponential but grows with a much slower rate:

$$O((M * N)^{(M*N)}) > O(M * c)^{(M*c)} \text{ because } c \ll N \quad (11)$$

The consolidation decision time when the service center logical hierarchical structuring is used, significantly improves when the ratio between the number of computation resources from clusters ( $c$ ) and the service center total number of servers ( $N$ ) decreases. If the difference between  $c$  and  $N$  is small there are few logical clusters created and the algorithm complexity remains the same:

$$\lim_{c \rightarrow N} O(M * c)^{(M*c)} = O((M * N)^{(M*N)}) \quad (12)$$

## VI. CONCLUSIONS

In this paper a server consolidation methodology for large service centers based on logically clustering the service center in a hierarchical manner is proposed. Each logical cluster is being managed by its own instance of a reinforcement learning based consolidation algorithm.

By analyzing the consolidation solution complexity, it can be seen that using the proposed methodology the consolidation decision time complexity remain exponential but its growing rate is much slower. Our methodology consolidation decision time rate of improvement is proportional with the ration between the service center total number servers and the logical clusters number of servers.

For future work we intend to implement and test the proposed methodology for a simulated large service center with the goal of assessing its energy efficiency. We will take into account the overhead and energy penalty induced by powering on/off or migrating a task in the methodology clusters and we will calculate the Deployed Hardware Utilization Ratio (DH-UR) for the testing service center.

#### REFERENCES

- [1] U. S. Environmental Protection Agency, ENERGY STAR Program, *Report to Congress on Server and Data Center Energy Efficiency*, Public Law 109-431, 2007.
- [2] M. Uddin, A. Rahman, "Server Consolidation: An Approach to Make Data Centers Energy Efficient & Green", *International Journal of Scientific & Engineering Research*, Volume 1, Issue 1, 2010.
- [3] N. E. Jerger, D. Vantrease and M. Lipasti, "An Evaluation of Server Consolidation Workloads for Multi-Core Designs", *Proceedings of the IEEE International Symposium on Workload Characterization*, 2007.
- [4] A. Verma, G. Dasgupta, T. Kumar Nayak, et al., "Server workload analysis for power minimization using consolidation", In *Proceedings of the USENIX Annual technical conference*, 2009.
- [5] Material Stock in German Data Centres, available at <http://www.uba-green-it.de>, 2010.
- [6] L. Wang, G. Laszewski, J. Dayaly, et al., "Towards Thermal Aware Workload Scheduling in a Data Center", In *Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms, and Networks*, pp. 116-122, ISBN: 978-0-7695-3908-9, 2009.
- [7] E. Kalyvianaki and T. Charalambous, "On Dynamic Resource Provisioning for Consolidated Servers in Virtualized Data Centers", *Proceedings of the 8th Int. Workshop on Performability Modeling of Computer and Communication Systems (PMCCS-8)*, 2007.
- [8] B. Speitkamp and M. Bichler, "A Mathematical Programming Approach for Server Consolidation Problems in Virtualized Data Centers", *IEEE Transactions on services computing*, Vol. 3 (4), 2010.
- [9] M. Marzolla, O. Babaoglu, and F. Panzieri, "Server Consolidation in Clouds through Gossiping", *Technical Report. University of Bologna, Department of Computer Science*, 2011.
- [10] J. Torres, D. Carrera, et al., "Tailoring Resources: The Energy Efficient Consolidation Strategy Goes Beyond Virtualization", In *Proceedings of the International Conference on Autonomic Computing*, pp. 197 - 198, ISBN: 978-0-7695-3175-5, 2008.
- [11] S. Srikantaiah, A. Kansal and F. Zhao, "Energy Aware Consolidation for Cloud Computing". *Technical Report. Microsoft Research*, 2009.
- [12] D. Barbagallo, E. Di Nitto, D. Dubois, and R. Mirandola, "A bio-inspired algorithm for energy optimization in a self-organizing data center". In *Proceedings of the First international conference on Self-organizing architectures*, SOAR'09, pp. 127-151, 2010.
- [13] J. Berral, I. Goiri, R. Nou, et al., "Towards energy-aware scheduling in data centers using machine learning", In *Proceedings of the Int'l Conf. on Energy-Efficient Computing and Networking*, 2010.
- [14] T. Cioara, I. Anghel, I. Salomie, G. Copil, D. Moldovan and B. Pernici, "A context aware self-adapting algorithm for managing the energy efficiency of IT service centres", *Ubiquitous Computing and Communication Journal*, Special Issue of 9th RoEduNet International Conference, Volume 6 No. 1, ISSN Online 1992-8424 , 2011.
- [15] I. Salomie, T. Cioara, I. Anghel, G. Copil, D. Moldovan, and P. Plebani, "An Energy Aware Context Model for Green IT Service Centers", In *Post-proceedings of the first international workshop on services, energy, & ecosystem (SEE-ICSOC 2010)*, Lecture Notes in Computer Science, Volume 6568/2011, 169-180, DOI: 10.1007/978-3-642-19394-1\_18, 2011.
- [16] G. Tesauro, N. K. Jong, R. Das, M. N. Bennani, "On the use of hybrid reinforcement learning for autonomic resource allocation", *Cluster Computing* 10(3): 287-299, 2007.
- [17] J. O. Kephart, H. Chan, R. Das, D. W. Levine, G. Tesauro, et al., "Coordinating Multiple Autonomic Managers to Achieve Specified Power-Performance Tradeoffs", ICAC 2007.





# Author Index

- A**  
Adduci, Michele ..... 743  
Aerts, Diederik ..... 221  
Afonso, Joao ..... 777  
Ahmed, Farag ..... 3  
Ahrndt, Sebastian ..... 305  
Al-Zokari, Yasmin I. .... 783  
Alberola, Auxiliadora Carlos ..... 925  
Alcazar, Anna Esparcia ..... 925  
Aleksić, Slavica ..... 825  
Alqarni, Mohammed ..... 11  
Alshayji, Sameera ..... 179  
Alt, Rainer ..... 521  
Alyani, Neek ..... 497  
Amundsen, JÅžrn ..... 431  
Anfinogenov, Sergey ..... 685  
Anghel, Ionut ..... 1005  
Arabi, Yassen ..... 11  
Astilean, Adina ..... 763  
Augustyniak, Piotr ..... 401  
Aversa, Rocco ..... 973
- B**  
Baars, Arthur ..... 917  
Badica, Amelia ..... 277  
Badica, Costin ..... 277, 597  
Bagnato, Alessandra ..... 925  
Baier, Daniel ..... 261  
Bajorek, Marcin ..... 371  
Bakrawy, Lamiaa M. El ..... 19  
Bala, Piotr ..... 723  
Baranski, Tomasz ..... 231  
Barateiro, José ..... 791  
Barnaghi, Payam ..... 949  
Bartkowiak, Anna ..... 25  
Battad, Bryan Temprado ..... 887  
Bauer, Martin ..... 949  
Becker, Jörg ..... 545  
Becker, Michael ..... 505  
Belo, Orlando ..... 109  
Bernon, Carole ..... 635  
Bianchi, Valentina ..... 375  
Biały, Tomasz ..... 355  
Bica, Mihai ..... 941  
Bjekovic, Marija ..... 513  
Borbinha, José ..... 791  
Bottcher, Martin ..... 505  
Branki, Cherif ..... 623  
Brezovan, Marius ..... 699  
Buczek, Bartłomiej ..... 77  
Bujnowski, Adam ..... 381, 393  
Bylina, Beata ..... 423, 459  
Bylina, Jarosław ..... 423, 459
- C**  
Cabezuelo, Antonio Sarasa ..... 855, 887  
Caccia, Michele ..... 743  
Calderon, Luis ..... 675  
Carver, Norman ..... 581  
Ceken, Kagan ..... 165  
Cerezo, Daniel Rodriguez ..... 855  
Chifu, Emil Stefan ..... 933  
Chifu, Viorica Rozina ..... 997  
Chodarev, Sergej ..... 891  
Chomiak-Orsa, Iwona ..... 281  
Chwatal, Andreas ..... 239  
Ciampolini, Paolo ..... 375  
Cirrincione, Maurizio ..... 619  
Clifton, David ..... 125  
Clifton, Lei ..... 125  
Coleša, Adrian ..... 941  
Copil, Georgiana ..... 1005  
Cossentino, Massimo ..... 611, 619  
Cybula, Piotr ..... 841  
Czachor, Marek ..... 221
- Č**  
Čeliković, Milan ..... 825
- D**  
Dale, Robert ..... 201  
De, Suparna ..... 949  
Derksen, Christian ..... 623  
Dikenelli, Oguz ..... 635  
Dinkelaker, Tom ..... 809  
Djukic, Verislav ..... 817  
Drag, Pawel ..... 477  
Dussault, Jean-Pierre ..... 247
- E**  
Eguchi, Akihiro ..... 631  
El-Bendary, Nashwa ..... 153  
Eleftherakis, George ..... 561  
Eleyat, Mujahed ..... 431  
Elhassan, Osama ..... 363  
Elkadhi, Nahla Elzant ..... 179  
Erradi, Mohammed ..... 809  
Evreinov, Grigori ..... 691  
Evreinova, Tatiana V. .... 691
- F**  
Fijałkowski, Damian ..... 287  
Fischbach, Michael ..... 521  
Fister, Iztok ..... 801  
Fister, Iztok Jr. .... 801  
Flouri, Tomas ..... 899  
Folgado, Salvador I. .... 925  
Fortino, Giancarlo ..... 569  
Fortis, Teodor-Florin ..... 973

<b>G</b> aldon, Asuncion Santamaria .....	729	Kehris, Evangelos .....	561
Ganea, Eugen .....	699	Khedr, Mohammed .....	11
Gannat, Gebriel .....	255	Khorasani, Elham S. ....	33
Ganzha, Maria .....	439, 443, 589	Kim, Tai-hoon .....	19
Garrido, David Garrido .....	729	Kiselev, Andrey .....	577
Gatsou, Chrysoula .....	705	Klingner, Stephan .....	505
Gaudin, Benoit .....	957	Klyuev, Vitaly .....	195
Gepner, Paweł .....	443	Klügl, Franziska .....	643, 651
Ghali, Neveen I. ....	19	Kobos, Mateusz .....	291
Ghali, Neven .....	157	Kobusiński, Jacek .....	355
Giurca, Adrian .....	261	Kocejko, Tomasz .....	405
Gnyba, Marcin .....	387	Koczkodaj, Waldemar W. ....	11
Gomez, Hector .....	675	Kollar, Jan .....	891
González, Igor .....	985	Korczak, Jerzy .....	41, 69
Greco, Salvatore .....	103	Korfiatis, Georgios .....	879
Grossi, Ferdinando .....	375	Korneev, Victor .....	577
Gryncewicz, Wiesława .....	281	Korzhik, Valery .....	685
Grzenda, Maciej .....	291	Kot, Jacek .....	417
Guerrieri, Antonio .....	569	Kotorowicz, Stanisław .....	485
Gurcan, Onder .....	635	Kreutzová, Michaela .....	895
Guzman, Liliana .....	783	Kruczkowski, Michał .....	387
Gómez-Martínez, Elena .....	985	Kubicki, Sylvain .....	513
<b>H</b> agen, Hans .....	783	Kulikowski, Juliusz .....	47
Haralambous, Yannis .....	195	<b>L</b> akatoš, Dominik .....	895
Hassanien, Aboul Ella .....	153, 157	Lakhotia, Kiran .....	917
Hassanien, Aboul ella .....	19	Leszczyńska, Maja .....	281
Haupt, Tomasz .....	965	Leszek, J. ....	11
Helfert, Markus .....	317	Letia, Ioan Alfred .....	933
Hessler, Axel .....	305	Letia, Tiberiu .....	763
Hinchey, Mike .....	957	Levashenko, Vitaly .....	169
Holgado-Terriza, Juan A. ....	529	Lewandowska, Magdalena .....	405
Homayounfar, Payam .....	133	Lirkov, Ivan .....	443
Houle, Daniel .....	33	Livnat, Yarden .....	783
Hrnčič, Dejan .....	801	Lodato, Carmelo .....	611, 619
Hussain, Amir .....	659	Lopes, Salvatore .....	619
<b>I</b> liopoulos, Costas .....	899	Louloudi, Athanasia .....	651
Ivančević, Vladimir .....	825	Luckner, Marcin .....	291
<b>J</b> anicki, Artur .....	711	Luetzenberger, Marco .....	305
Janoušek, Jan .....	871, 899, 903	Luković, Ivan .....	817, 825
Jaszuk, Marek .....	187	<b>Ł</b> uszczzyk, Walter .....	41
Jaworek, Joanna .....	401	<b>M</b> aamar, Zakaria .....	363
Jelinek, Ivan .....	991	Maciaszek, Leszek A. ....	329
Jestadt, Thomas .....	541	Maciejewski, Henryk .....	55
Junges, Robert .....	643	Maida, Martina .....	297
Jędrzejewska-Szczerska, Małgorzata .....	387	Maier, Konradin .....	297
<b>K</b> aczmarek, Mariusz .....	393	Malsbender, Andrea .....	545
Kaczmarek, Krzysztof .....	291	Marin, Beatriz .....	925
Kakiashvili, Tamar .....	11	Maroszy, Zoltan .....	763
Kashfi, Hajar .....	347	Martino, Beniamino .....	973
Kańtoch, Eliaz .....	401	Mastroeni, Loretta .....	537
Kefalas, Petros .....	561	Masuch, Nils .....	305
		Mazur, Paweł .....	201

Małecki, Maciej .....	355	Pataki, Norbert .....	911
Meissner, Stefan .....	949	Patel, Purvag .....	581
Melichar, Borivoj .....	871, 899, 903	Patyk-Łońska, Agnieszka .....	213, 221
Meresta, Anna .....	417	Pavon, Juan .....	675
Mernik, Marjan .....	801	Petrisor, Constantin .....	737
Miarka, Rostislav .....	63	Petech-Pilichowski, Tomasz .....	321
Michalak, Krzysztof .....	69	Pfitzinger, Bernd .....	541
Mihaescu, Cristian .....	717	Pianini, Danilo .....	667
Mihaescu, Cristian Marian .....	737	Pietrikova, Emilia .....	891
Mindruta, Cristina .....	981	Pissis, Solon .....	899
Mocanu, Mihai .....	737	Piątek, Łukasz .....	147
Moderhak, Mariusz .....	411	Plattfaut, Ralf .....	545
Moderhak, Mateusz .....	411	Plebani, Pierluigi .....	1005
Moldovan, Daniel .....	1005	Plicka, Martin .....	903
Montagna, Sara .....	667	Pliszka, Zbigniew .....	91
Morales-Luna, Guillermo .....	685	Polberg, Sylwia .....	589
Moscato, Francesco .....	973	Polinski, Artur .....	417
Mozgovoy, Maxim .....	209	Politis, Anastasios .....	705
Mračka, Igor .....	451	Pop, Cristina Bianca .....	997
Mucherino, Antonio .....	269	Popovic, Aleksandar .....	817
Munari, Ilaria De .....	375	Porkoláb, Zoltán .....	911
Munteanu, Victor .....	973	Porubän, Jaroslav .....	895
Munteanu, Victor Ion .....	981	Poteras, Cosmin Marian .....	737
Murillo, José Oliver .....	925	Przelaskowski, A. .....	11
Murua, Ane .....	985	Pucci, Marcello .....	619
Myszkowski, Paweł .....	77	Purvag, Patel .....	33
		Puschmann, Thomas .....	521
<b>N</b>		Puzio, Leszek .....	187
Naldi, Maurizio .....	537	Pytel, Krzysztof .....	97
Natvig, Lasse .....	431	Pyzara, Anna .....	459
Neacsu, Gabriela .....	763	PöppelbuÅš, Jens .....	545
Negru, Viorel .....	981		
Nguyen, Filip .....	313	<b>R</b>	
Nguyen, Hung .....	631	Rafajłowicz, Wojciech .....	471
Niazi, Muaz .....	659	Rahimi, Shahram .....	33, 581
Niehaves, Björn .....	545	Raidl, Günther .....	239
Nowak, Jędrzej .....	371, 405	Raisamo, Roope .....	691
Nürnbergger, Andreas .....	3	Rajasekar, Pallikonda .....	363
		Ramamoorthy, Lavanya .....	337
<b>O</b>		Redondo, Carmen Lastres .....	729
Oberlechner, Karin .....	239	Refice, Mario .....	743
Obwegeser, Nikolaus .....	297	Revett, Kenneth .....	153
Ohzeki, Kazuo .....	767	Reyes, Flavio .....	675
Ortbach, Kevin .....	545	Ribino, Patrizia .....	611
Ostrowski, Lukasz .....	317	Rodriguez, Jose Luis Sierra .....	855, 887
Owoc, Mieczysław .....	133	Rodríguez-Valenzuela, Sandra .....	529
Owsiany, Grzegorz .....	147	Romanczuk, Urszula .....	485
		Rudomilov, Ilya .....	991
<b>P</b>		Ruminski, Jacek .....	393, 405
Paliński, Arkadiusz .....	381	Rutkowski, K. .....	11
Pancerz, Krzysztof .....	141, 147		
Pandiyan, Murugavell .....	363	<b>S</b>	
Panka, Maciej .....	723	Sabo, Miroslav .....	895
Papakyriakou, Michalis .....	833, 879	Sadolewski, Jan .....	849
Papaspyrou, Nikolaos .....	833, 879	Saka, Osman .....	165
Paprzycki, Marcin .....	439, 443, 589	Salama, Mostafa .....	153
Paradowski, Mariusz .....	83	Salomie, Ioan .....	997, 1005
Pascual, Leticia Carnero .....	729		

Salvatore, Lopes .....	611	Unland, Rainer .....	623
Sanchez, Belen Rios .....	729	Unold, Olgierd .....	91
Sandru, Calin .....	981	Urban, Joseph .....	337
Sansores, Candelaria .....	675	Ustimenko, Vasył .....	485
Savino, Michelina .....	743	<b>V</b> azhenin, Alexander .....	491
Scafes, Mihnea .....	597	Vazhenin, Dmitry .....	491
Schmidt, Rainer .....	553	Vazou, Niki .....	833
Schmitt, Ingo .....	261	Veiga, Pedro .....	777
Schneider, Daniel .....	783	Viroli, Mirko .....	667
Sedukhin, Stanislav .....	439	Vitale, Gianpaolo .....	619
Semenov, Dmitry .....	577	Vlad, Monica .....	997
Shabanov, Boris .....	577	Voigt, Matthias .....	545
Shirzad, Sara .....	497	Vos, Tanja E. J. ....	917, 925
Shvartsman, Inna .....	605	<b>W</b> alczak, Andrzej .....	187
Siddique, Qasim .....	659	Wang, Zidong .....	179
Sienkiewicz, Lukasz D. ....	329	Wanzeller, Cristina .....	109
Slivnik, Bostjan .....	863	Warchol, Jan .....	141
Słowinski, Roman .....	103	Wassermann, Lubomir .....	891
Somora, Peter .....	451	Watkinson, Peter .....	125
Spahiu, Cosmin Stoica .....	751	Wawer, Dariusz .....	711
Stanescu, Liana .....	755	Wegener, Joachim .....	917
Stefaniak, Krzysztof .....	355	Wtorek, Adam .....	381
Stix, Volker .....	297	Wtorek, Jerzy .....	411
Stpiczynski, Przemyslaw .....	465	<b>X</b> iao, Kai .....	157
Styczeń, Krystyn .....	471, 477	<b>Y</b> ildirim, Pinar .....	165
Subburaj, Vinitha Hannah .....	337	Yutaka, Hirakawa .....	767
Subieta, Kazimierz .....	841	<b>Z</b> aitseva, Elena .....	169
Sukhija, Nitin .....	965	Zatoka, Radosław .....	287
Szczęch, Izabela .....	103	Zeckzer, Dirk .....	783
Szkoła, Jarosław .....	141	Zevgolis, Dimitrios .....	705
Szostek, Grazyna .....	187	Zgódka, Marek .....	341
<b>Ś</b> luzek, Andrzej .....	83	Zhuk, Igor .....	965
<b>Š</b> krabálek, Jaroslav .....	313	Zimroz, Radosław .....	25
<b>T</b> ake, Hiroyuki .....	767	<b>Ž</b> atuchin, Dmitrij .....	117
Tarassenko, Lionel .....	125	<b>Ž</b> áček, Martin .....	63
Taveter, Kuldar .....	605	Žáčik, Tibor .....	451
Telegin, Pavel .....	577		
Thöni, Corinna .....	239		
Travnicek, Jan .....	871		
Troníček, Zdeněk .....	907		
Truyen, Bart .....	411		
Tudor, Cioara .....	1005		
Tudoroiu, Elena-Roxana .....	763		
Tudoroiu, Nicolae .....	763		



