Proceedings of the International Multiconference on Computer Science and Information Technology

Volume 5 (2010)

Proceedings of the International Multiconference on Computer Science and Information Technology

Volume 5 (2010)

M. Ganzha, M. Paprzycki (editors)

ISSN 1896-7094

Polskie Towarzystwo Informatyczne
Oddział Górnośląski
ul. Raciborska 3
40-074 Katowice

ISBN 978-83-60810-27-9

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

TEXnical editor: Aleksandr Denisiuk

Proceedings of the International Multiconference on Computer Science and Information Technology

October 18–20, 2010. Wisła, Poland

Volume 5 (2010)

Dear Reader, it is our pleasure to present to you Proceedings of the 2010 International Multiconference on Computer Science and Information Technology (IMCSIT), which took place in Wisła, Poland, on October 18–20, 2010. IMCSIT 2010 and was co-located with the XXVI Autumn Meeting of the Polish Information Processing Society (PIPS).

IMCSIT is a result of the evolutionary process. In 2005 a Scientific Session took place during the XXI Autumn Meeting of PIPS and consisted of 27 refereed presentations. After this relative success (we have advertised the Session very late in the year) we have decided to expand and extend it into a full-blown conference but continue cooperation (co-location) with the Autumn Meetings of PIPS. As a result of a steady growth, in 2010, IMCSIT consisted of the following events (and Proceedings are organized into sections that correspond to each of them):

- 5th International Symposium Advances in Artificial Intelligence and Applications (AAIA'10),
- Workshop on Agent Based Computing: from Model to Implementation VII (ABC:MI'10),
- International Workshop on Advances in Business ICT (ABICT'10),
- Computer Aspects of Numerical Algorithms (CANA'10),
- Computational Linguistics—Applications (CLA'10),
- 10th International Multidisciplinary Conference on e-Commerce and e-Government (ECOM&EGOV'10),
- International Symposium on E-Learning—Applications (EL-A'10),
- 6th Workshop on Large Scale Computations on Grids and 1st Workshop on Scalable Computing in Distributed Systems (LaSCoG-SCoDiS'10),
- 2nd International Workshop on Medical Informatics and Engineering (MI&E'10),
- 3rd International Symposium on Multimedia—Applications and Processing (MMAP'10),
- International Workshop on Real Time Software (RTS'10),
- 4th International Workshop on Secure Information Systems (SIS'10),
- International Symposium on Technologies for Social Advancement (T4SA'10),
- Workshop on Ad-Hoc Wireless Networks (WAHOC'10),
- Workshop on Computational Optimization (WCO'10).

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 201 submissions.

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

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

Proceedings of the International Multiconference on Computer Science and Information Technology

Volume 5

October 18 – 20, 2010. Wisła, Poland

TABLE OF CONTENTS

5$^{\text{TH}}$ International Symposium Advances in Artificial Intelligence and Applications:

Workshop on Agent Based Computing: from Model to Implementation VII:

International Workshop on Advances in Business ICT:

Computer Aspects of Numerical Algorithms:

Computational Linguistics–Applications:

10$^{\text{TH}}$ International Multidisciplinary Conference on e-Commerce and e-Government:

International Symposium on E-Learning–Applications:

6TH Workshop on Large Scale Computations on Grids and 1st Workshop on Scalable Computing in Distributed Systems:

2ND International Workshop on Medical Informatics and Engineering:

3RD International Symposium on Multimedia–Applications and Processing:

International Workshop on Real Time Software:

4TH International Workshop on Secure Information Systems:

International Symposium on Technologies for Social Advancement:

Workshop on Ad-Hoc Wireless Networks:

Workshop on Computational Optimization:

5th International Symposium
Advances in Artificial Intelligence and Applications

CELEBRATING 75TH BIRTHDAY OF PROFESSOR LEONARD BOLC

The AAIA'10 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'10 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
- Artificial Immune Systems
- Ant Systems in Applications
- 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

INTERNATIONAL PROGRAMME COMMITTEE

Organizing Committee

ISBN 978-83-60810-27-9
ISSN 1896-7094

A Breast Cancer Classifier based on a Combination of Case-Based Reasoning and Ontology Approach

Essam Amin M.Lotfy Abdrabou
Ph.D Candidate
Faculty of Computer and Information Sciences
Ain Shams University, Abbassia, 11566, Cairo, EGYPT
(+202) 26330636
Email: gm@ceslabs.com

AbdEl-Badeeh M. Salem
Professor
Faculty of Computer and Information Sciences
Ain Shams University, Abbassia, 11566, Cairo, EGYPT
(+202) 26844284
Email: absalem@asunet.shams.edu.eg

***Abstract*—Breast cancer is the second most common form of cancer amongst females and also the fifth most cause of cancer deaths worldwide. In case of this particular type of malignancy, early detection is the best form of cure and hence timely and accurate diagnosis of the tumor is extremely vital. Extensive research has been carried out on automating the critical diagnosis procedure as various machine learning algorithms have been developed to aid physicians in optimizing the decision task effectively. In this research, we present a benign/malignant breast cancer classification model based on a combination of ontology and case-based reasoning to effectively classify breast cancer tumors as either malignant or benign. This classification system makes use of clinical data. Two CBR object-oriented frameworks based on ontology are used *jCOLIBRI* and *myCBR*. A breast cancer diagnostic prototype is built. During prototyping, we examine the use and functionality of the two focused frameworks.**

***Index Terms*—Case-Based Reasoning, Case-Based Reasoning Frameworks, CBR, CBR Frameworks, jCOLIBRI, myCBR, Breast Cancer**

I. Introduction

BREAST cancer classification, diagnosis and prediction techniques have been a widely researched area in the past decade in the world of medical informatics. Several articles have been published which tries to classify breast cancer data sets using various techniques such as fuzzy logic, support vector machines, Bayesian classifiers, decision trees and neural networks. Classification accuracy as high as 98.8% has been achieved using a learning algorithm combining simulated annealing with the perceptron algorithm. Another study involving fuzzy modeling and cooperative co-evolution has gained an accuracy of 98.98% over one of the widely studied Wisconsin breast cancer database [16].

This research applies a new technique in the field of breast cancer classification. It uses a combination of ontology and case-based reasoning by using ontology based object-oriented case-based reasoning frameworks. Two frameworks are examined in building the classifier. One is the open source *jCOLIBRI* [5] system developed by *GAIA* group and provides a framework for building CBR systems based on state-of-the-art software engineering techniques. The other is the novel open source CBR tool *myCBR* [24] developed at the German Research Center for Artificial Intelligence *(DFKI)*. The objective of this classifier is to classify the patient based on his/her electronic record whether he/she is benign or malignant.

This paper is organized in four sections. Section 1 is this introduction. Section 2 gives a theoretical background about breast cancer, ontology, CBR and object-oriented frameworks. Section 3 illustrates the implementation of the breast cancer classifier on the two frameworks. Finally, section 4 discusses and concludes the results

II. Theoritical Background

A. Breast Cancer

Breast cancer is the form of cancer that either originates in the breast or is primarily present in the breast cells. The disease occurs mostly in women but a small population of men is also affected by it. Breast cancer is the most common form of cancer amongst the female population as well as the most common cause of cancer deaths [25]. Early detection of breast cancer saves many thousands of lives each year. Many more could be saved if the patients are offered accurate, timely analysis of their particular type of cancer and the available treatment options. Since the breast tumors whether malignant or benign share structural similarities, it becomes an extremely tedious and time consuming task to manually differentiate them. As seen in Figure 1 there is no visually significant difference between the fine needle biopsy image of the malignant and benign tumor for an untrained eye. Accurate classification is very important as the potency of the cytotoxic drugs administered during the treatment can be life threatening or may develop into another cancer. Laboratory analysis or biopsies of the tumor is a manual, time consuming yet accurate

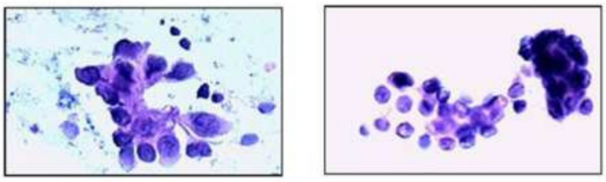


Fig. 1. Fine needle biopsies of breast. Malignant (left) and Benign (right) [25]

system of prediction. It is however prone to human errors, creating a need for an automated system to provide a faster and more reliable method of diagnosis and prediction for the patients.

B. Ontology

Ontology is a formal explicit description of concepts in a domain of discourse (classes (sometimes called concepts)), properties of each concept describing various features and attributes of the concept (slots (sometimes called roles or properties)), and restrictions on slots (facets (sometimes called role restrictions)). Ontology together with a set of individual instances of classes constitutes a knowledge base. In reality, there is a fine line where the ontology ends and the knowledge base begins [8].

C. Case-Based Reasoning

In case-based reasoning (CBR) systems expertise is embodied in a library of past cases, rather than being encoded in classical rules. Each case typically contains a description of the problem, plus a solution and/or the outcome. The knowledge and reasoning process used by an expert to solve the problem is not recorded, but is implicit in the solution. To solve a current problem: the problem is matched against the cases in the case base, and similar cases are retrieved. The retrieved cases are used to suggest a solution that is reused and tested for success. If necessary, the solution is then revised. Finally the current problem and the final solution are retained as part of a new case.

The CBR process can be represented by a schematic cycle, as shown in Figure 2 [1].

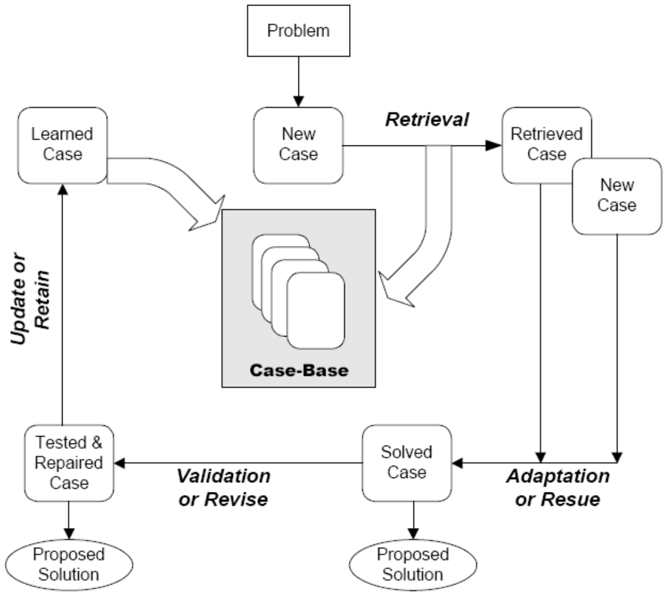


Fig. 2. The CBR Cycle

*R*epresentation: Given a new situation, generate appropriate semantic indices that will allow its classification and categorization. This usually implies a standard indexing vocabulary that the CBR system uses to store historical information and problems. The vocabulary must be rich enough to be expressive, but limited enough to allow efficient recall [2].

*R*etrieval: Given a new, indexed problem, retrieve the best past cases from memory. This requires answering three questions: What constitute an appropriate case? What are the criteria of closeness or similarity between cases? How should cases be indexed? Part of the index must be a description of the problem that the case solved, at some level of abstraction. Part of the case, though, is also the knowledge gained from solving the problem represented by the case. In other words, cases should also be indexed by some elements of their solution [11].

*A*daptation: Modify the old solutions to confirm to the new situation, resulting in a proposed solution. With the exception of trivial situations, the solution recalled will not immediately apply to the new problem, usually because the old and the new problem are slightly different. CBR researchers have developed and used various adaptation techniques [11].

*V*alidation: After the system checks a solution, it must evaluate the results of this check. If the solution is acceptable, based on some domain criteria, the CBR system is done with reasoning. Otherwise, the case must be modified again, and this time the modifications will be guided by the results of the solution's evaluation [11].

*U*pdate: If the solution fails, explain the failure and learn it, to avoid repeating it. If the solution succeeds and warrants retention, incorporate it into the case memory as a successful solution and stop. The CBR system must decide if a successful new solution is sufficiently different from already-known solutions to warrant storage. If it does warrant storage, the system must decide how the new case will be indexed, on which level of abstraction it will be saved, and where it will be put in the case-base organization [11].

Retaining the case is the process of incorporating whatever is useful from the new case into the case library. This involves deciding what information to retain and in what form to retain it; how to index the case for future retrieval; and integrating the new case into the case library.

D. CBR Object-Oriented Frameworks

The concept of object-oriented frameworks has been introduced in the late 80's and has been defined as a set of classes that embodies an abstract design for solutions to a family of related problems, and supports reuses at a larger granularity than classes [9].

The goal of a framework is to capture a set of concepts related to a domain and the way they interact. In addition, a framework is in control of a part of the program activity and calls specific application code by dynamic method binding. A framework can be viewed as an incomplete application where the user only has to specialize some classes to build the complete application [9].

Frameworks allow the reuse of both code and design for a class of problems, giving the ability to non-expert to write complex applications quickly. Frameworks also allow the development of prototypes which could be extended further on by specialization or composition. A framework once understood, it can be applied in a wide range of domain, and can be enhanced by the adding of new components [9].

Using frameworks for development of new applications helps improve software quality. It improves programmers' productivity and quality, performance, and reliability of software. It also enhances extensibility by providing the required methods that allow applications to extend its stable interfaces [20]. Figure 3 clearly shows the difference of the effort required for developing an application from scratch and using a framework [15].

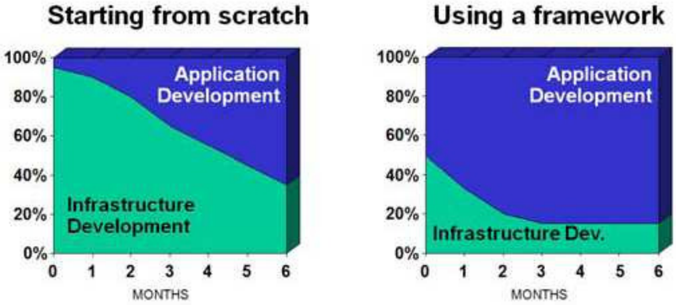


Fig. 3. Development Effort Reduction by using Frameworks

CBR researchers agree that the best way to satisfy the increasing demand of developing CBR application is by development of frameworks. Recently, some efforts within the CBR community have developed CBR frameworks [20]. This paper focuses on two of them *jCOLIBRI* developed by *GAIA* group and *myCBR* developed by *DFKI* group.

III. Experiments

A. Breast Cancer Classifications

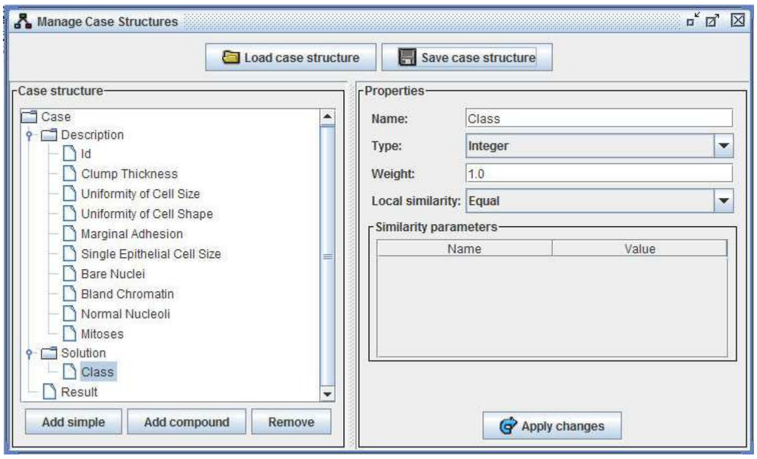
Breast cancer has become the number one cause of cancer deaths amongst women. Once a breast cancer is detected, it can be classified benign (not cancerous tissue) or malignant (cancerous tissue). In this study, the two compared CBR frameworks are tested by developing a CBR application that classifies the condition of the breast cancer tumor whether it is benign or malignant. Wisconsin breast cancer data set was used for building the case-bases. It is obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg [14]. Samples inside the data set arrive periodically as Dr. Wolberg reports his clinical cases. The number of instances inside the dataset is 699 (as of 15 July 1992). Each record contains ten attributes plus the class attribute. Table I shows the attributes and their possible values. 65.5% of the elements belong to the benign class and 34.5% to the malignant class. 16 elements are incomplete (an attribute is missing) and have been excluded from the database.

TABLE I
Wisconsin Breast Cancer Dataset

No.	Attribute	Possible Value
1	Sample code number	id number
2	Clump Thickness	1 – 10
3	Uniformity of Cell Size	1 – 10
4	Uniformity of Cell Shape	1 – 10
5	Marginal Adhesion	1 – 10
6	Single Epithelial Cell Size	1 – 10
7	Bare Nuclei	1 – 10
8	Bland Chromatin	1 – 10
9	Normal Nucleoli	1 – 10
10	Mitoses	1 – 10
11	Class	(2 for benign, 4 for malignant)

B. jCOLIBRI

1) Overview: *jCOLIBRI* is an evolution of the COLIBRI architecture [7], that consisted of a library of problem solving methods (PSMs) for solving the tasks of a knowledge-intensive CBR system along with ontology, CBROnto [8], with common CBR terminology. COLIBRI was prototyped in LISP using LOOM as knowledge representation technology. This prototype served as proof of concept; was very useful but it is not helpful for non-expert users. Then, people at *GAIA* group have started to develop a new complete framework with the name of *jCOLIBRI*. It stands for Cases and Ontology Libraries Integration for Building Reasoning Infrastructures. CBR ontology assumes the same vocabulary provided by any CBR system. In *jCOLIBRI*, ontology is not represented as a new source. All concepts of CBR are mapped into classes and interfaces of framework. Classes that represent the concept of ontology serve as templates where new CBR types should be added. They also provide the tasks and abstract interface of the methods. The design of the *jCOLIBRI* framework comprises a hierarchy of Java classes plus a number of XML files. The framework is organized around the following elements [2]:
Tasks and methods: The tasks supported by the framework and the methods that solve them are all stored in a set of XML files.
Case-base: Different connectors are defined to support several types of case determination, from the file system to a database.
Cases: A number of interfaces and classes are included in the framework to provide an abstract representation of cases that support any type of actual case structure.
Problem solving methods: The actual code that supports the methods included in the framework.
The *jCOLIBRI* comes in two major releases version 1 and version 2. According to the tutorial [19], version 2 is a new implementation that follows a new and clear architecture divided into two layers: one oriented to developers and other oriented to designers. Unfortunately, the only available distribution of version 2 is the one that is oriented to the developers which is out of scope of this paper. *jCOLIBRI* version 1 is the first release of the framework. It includes a complete Graphical



Name: Class

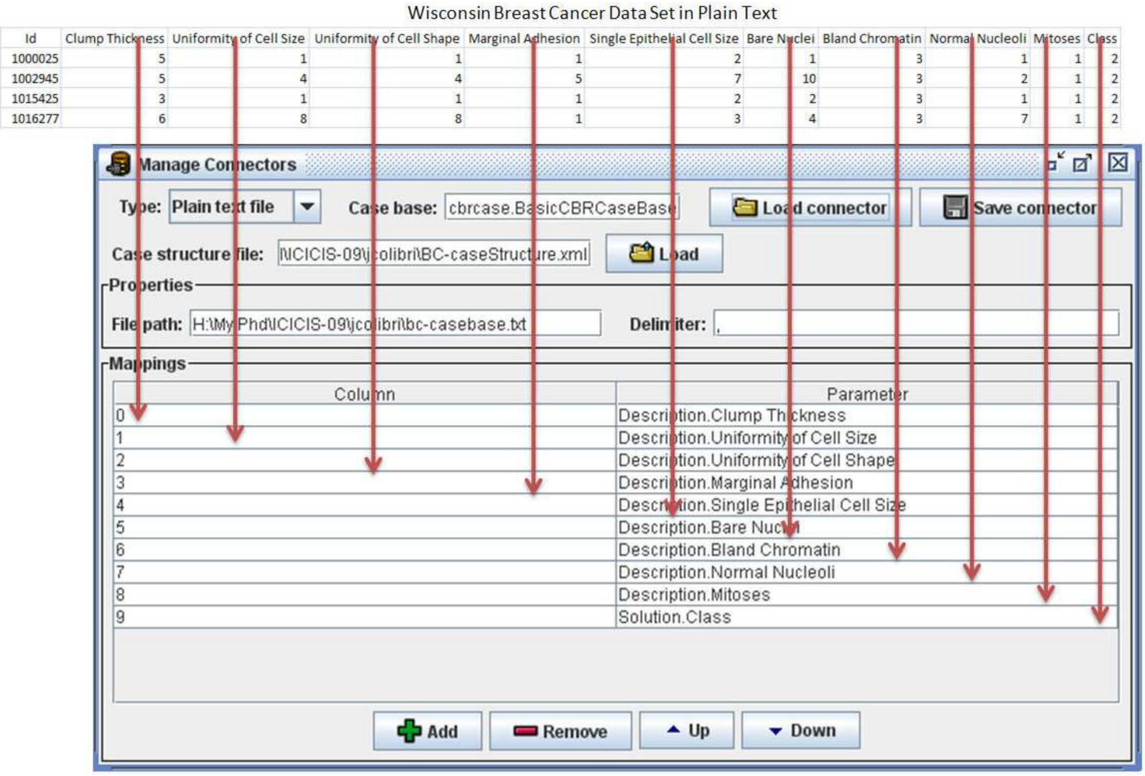
Type: Integer

Weight: 1.0

Local similarity: Equal

Name	Value

(a) Patient Case Definition in *jCOLIBRI*


Wisconsin Breast Cancer Data Set in Plain Text

Id	Clump Thickness	Uniformity of Cell Size	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	Class
1000025	5	1	1	1	2	1	3	1	1	2
1002945	5	4	4	5	7	10	3	2	1	2
1015425	3	1	1	1	2	2	3	1	1	2
1016277	6	8	8	1	3	4	3	7	1	2

Type: Plain text file

Case base: cbrcase.BasicCBRCaseBase

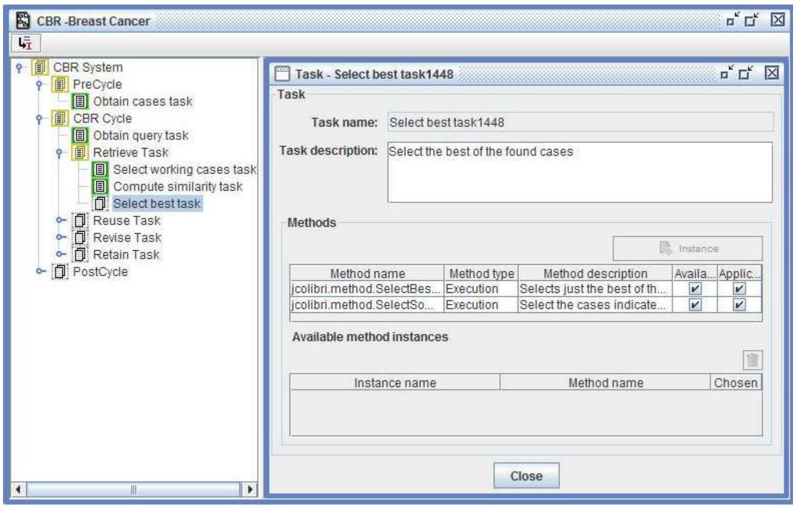
Case structure file: \ICICIS-09\jcolibri\BC-caseStructure.xml

File path: H:\My Phd\ICICIS-09\jcolibri\bc-casebase.txt

Delimiter: ,

Column	Parameter
0	Description.Clump Thickness
1	Description.Uniformity of Cell Size
2	Description.Uniformity of Cell Shape
3	Description.Marginal Adhesion
4	Description.Single Epithelial Cell Size
5	Description.Bare Nuclei
6	Description.Bland Chromatin
7	Description.Normal Nucleoli
8	Description.Mitoses
9	Solution.Class

(b) Managing Connectors in *jCOLIBRI*



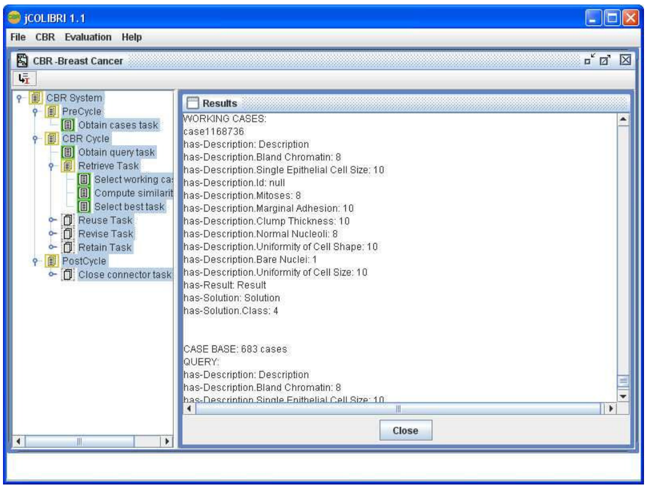
Task name: Select best task1448

Task description: Select the best of the found cases

Method name	Method type	Method description	Availa...	Applic...
jcolibri.method.SelectBes...	Execution	Selects just the best of th...	☑	☑
jcolibri.method.SelectSo...	Execution	Select the cases indicate...	☑	☑

Instance name	Method name	Chosen

(c) Configuration of Tasks in *jCOLIBRI*



```
WORKING CASES:
case1168736
has-Description: Description
has-Description.Bland Chromatin: 8
has-Description.Single Epithelial Cell Size: 10
has-Description.Id: null
has-Description.Mitoses: 8
has-Description.Marginal Adhesion: 10
has-Description.Clump Thickness: 10
has-Description.Normal Nucleoli: 8
has-Description.Uniformity of Cell Shape: 10
has-Description.Bare Nuclei: 1
has-Description.Uniformity of Cell Size: 10
has-Result: Result
has-Solution: Solution
has-Solution.Class: 4


CASE BASE: 683 cases
QUERY:
has-Description: Description
has-Description.Bland Chromatin: 8
has-Description.Single Epithelial Cell Size: 10
```

(d) *jCOLIBRI* Retrieval

Fig. 4. Implementation in *jCOLIBRI*

User Interface (GUI) that guides the user in the design of a CBR system. This version is recommended for non-developer users that want to create CBR systems without programming any code which is exactly the scope in this study. As a result, version 1 is selected to implement the required application. Downloading of the *jCOLIBRI* is an easy task; it can be obtained through the web page of GAIA group. It comes in a compressed distribution that can be easily extracted to have the full package. To run *jCOLIBRI*, there is a ready batch file (we are using MS Windows® platform) that can be invoked directly to run jCOLIBRI. It is required to have JAVA Virtual Machine installed before running the batch file. By invoking this batch file we get the first screen of the framework GUI.

2) Implementation: By the help of the multimedia tutorials provided and the GUI of the *jCOLIBRI*, users can go through five steps to implement and deploy a CBR System. These steps are

- Definition of case structures
- Building the case-base
- Managing similarity measures
- Configuring the behavior of the CBR process
- Testing and deploying the CBR application

Definition of Case Structures: By using *jCOLIBRI* GUI users are able to create the case structure defining simple and compound attributes that describe the cases together with their types, weights, similarity measure -that is chosen from a library of existing similarity functions and parameters. The case structure can be saved or loaded in and from a XML file. Figure 4(a) shows the definition of the patient case parameters.

Building the case-base: *jCOLIBRI* introduces the concept of Connectors which cases persistence is built around. Connectors are objects that know how to access and retrieve cases from the storage media and return those cases to the CBR system in a uniform way. Therefore connectors provide an abstraction mechanism that allows users to load cases from different storage sources in a transparent way [24] [21]. Defined connectors can work with plain text files, XML files, or relational data bases. The graphical interface helps mapping the defined case structure with the tables and columns from the storage scheme. Figure 4(b) shows how the patient case structure is mapped to columns in a text file containing the Wisconsin data set patient records.

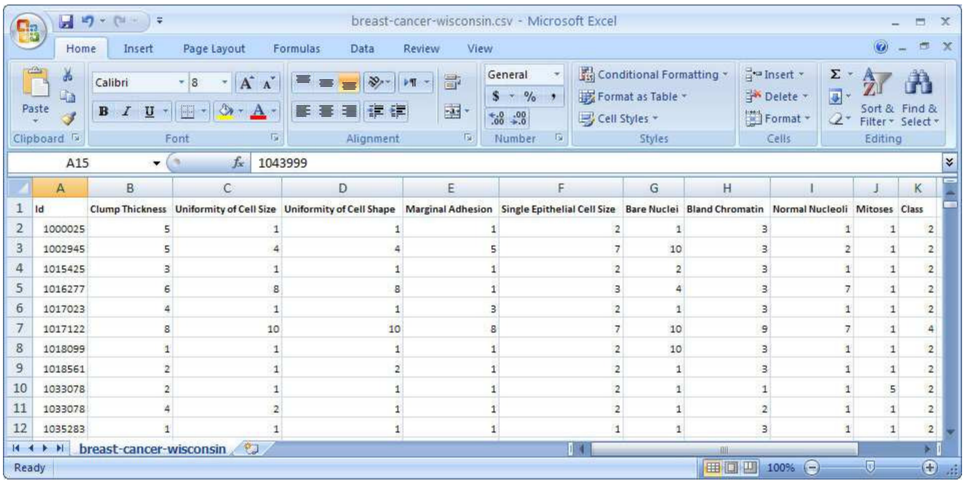
Managing similarity measures: When two cases are compared, the local similarity functions are used to compare simple attribute values. Global similarity functions are linked to compound attributes and are used to gather the similarities of the collected attributes in a unique similarity value. At last, the similarity value of two cases is computed as the similarity of their description concepts. The available similarity measures are listed in a configuration file, and can be managed using the GUI. Since our problem is simple, we leave the default similarity assigned by *jCOLIBRI*.

Configuring the behavior of the CBR process: As introduced, *jCOLIBRI* formalizes the CBR knowledge using CBR ontology (CBROnto), a knowledge level description of the CBR tasks and a library of reusable Problem Solving Methods (PSMs) [21]. Configuration of tasks is done in an interactive approach by choosing from a library of reusable methods one that is suitable to solve the selected task. Constraints of the selected task are being tracked during the configuration process so that only applicable methods in the given context are offered to users. In our comparison we focus only on the retrieval task. Figure 4(c) shows the configured tasks in the breast cancer application.

Testing and deploying the CBR application: The CBR application is finished when all the tasks have been configured. Users can test the system from inside the graphical interface. The first task of the CBR system, (Obtain query task)ÿ obtains the query that is going to be used to retrieve the most similar cases. Figure 4(d) shows the GUI after a query. We tested the 16 records that are excluded from the dataset according to one missing value. Only two missed classifications are obtained. Documentation mentions that it is possible to deploy the developed CBR application by generating a code template with most of the code required to run the developed system as an independent application. We have tried this process but it is completely failed.

C. *myCBR*

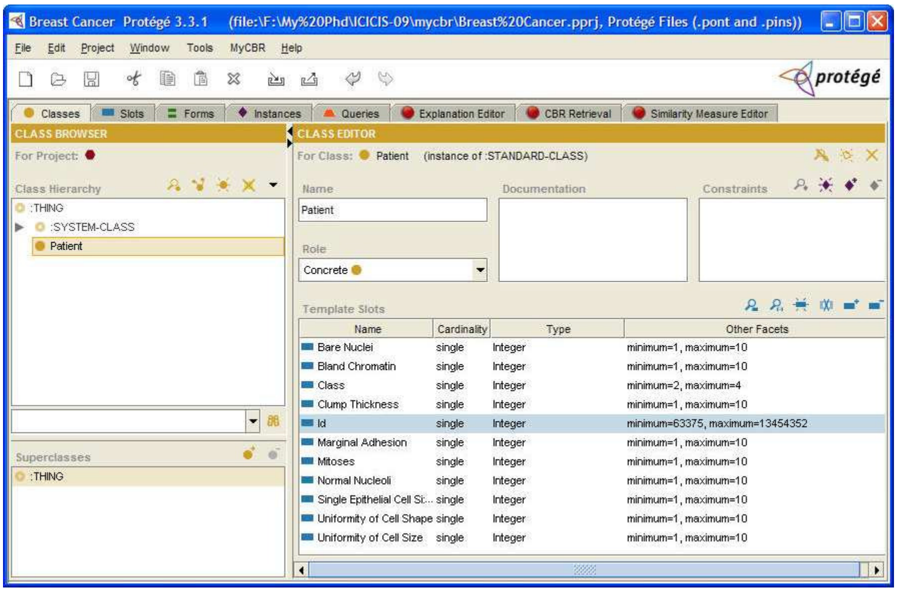
1) Overview: myCBR is an open-source plug-in for the open-source ontology editor Protégé [6]. Protégé is based on Java, is extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development [4]. Protégé [4] allows defining classes and attributes in an object-oriented way. Furthermore, it manages instances of these classes, which *myCBR* interprets as cases [22]. So the handling of vocabulary and case base is already provided by Protégé. The *myCBR* plug-in provides several editors to define similarity measures for an ontology and a retrieval interface for testing [24]. As the main goal of *myCBR* is to minimize the effort for building CBR applications that require knowledge-intensive similarity measures, *myCBR*



A15 f_x 1043999

	A	B	C	D	E	F	G	H	I	J	K
1	Id	Clump Thickness	Uniformity of Cell Size	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	Class
2	1000025	5	1	1	1	2	1	3	1	1	2
3	1002945	5	4	4	5	7	10	3	2	1	2
4	1015425	3	1	1	1	2	2	3	1	1	2
5	1016277	6	8	8	1	3	4	3	7	1	2
6	1017023	4	1	1	3	2	1	3	1	1	2
7	1017122	8	10	10	8	7	10	9	7	1	4
8	1018099	1	1	1	1	2	10	3	1	1	2
9	1018561	2	1	2	1	2	1	3	1	1	2
10	1033078	2	1	1	1	2	1	1	1	5	2
11	1033078	4	2	1	1	2	1	2	1	1	2
12	1035283	1	1	1	1	1	1	3	1	1	2

(a) Wisconsin Dataset in a CSV File



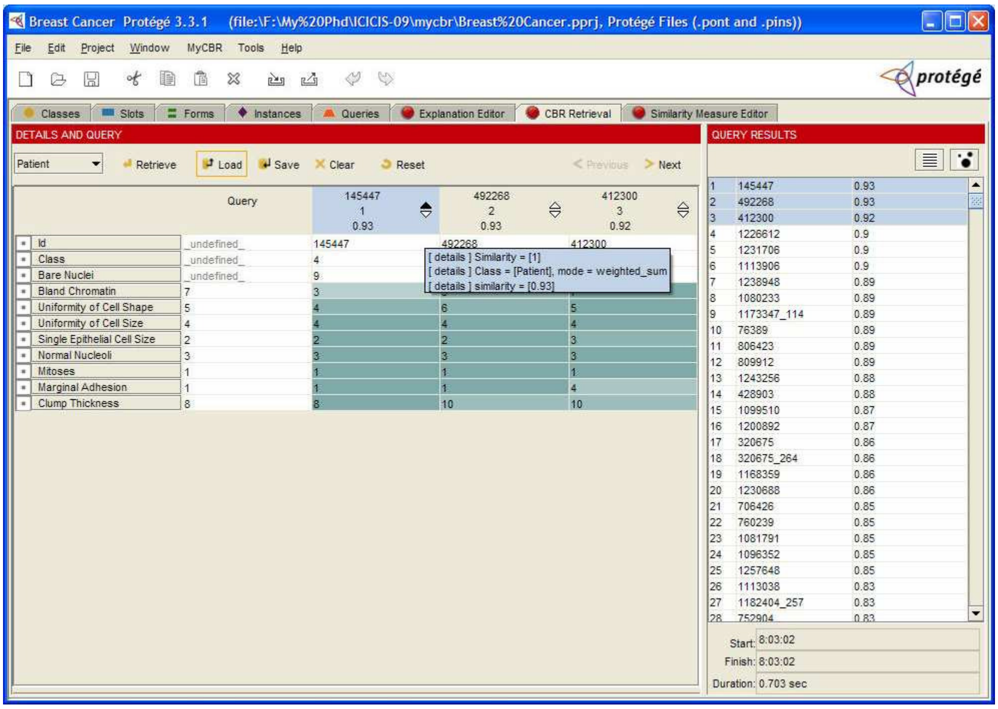
For Class: Patient (instance of :STANDARD-CLASS)

Name: Patient

Role: Concrete

Name	Cardinality	Type	Other Facets
Bare Nuclei	single	Integer	minimum=1, maximum=10
Bland Chromatin	single	Integer	minimum=1, maximum=10
Class	single	Integer	minimum=2, maximum=4
Clump Thickness	single	Integer	minimum=1, maximum=10
Id	single	Integer	minimum=63375, maximum=13454352
Marginal Adhesion	single	Integer	minimum=1, maximum=10
Mitoses	single	Integer	minimum=1, maximum=10
Normal Nucleoli	single	Integer	minimum=1, maximum=10
Single Epithelial Cell Si:...	single	Integer	minimum=1, maximum=10
Uniformity of Cell Shape	single	Integer	minimum=1, maximum=10
Uniformity of Cell Size	single	Integer	minimum=1, maximum=10

(b) Patient Case Data Representation in *myCBR*



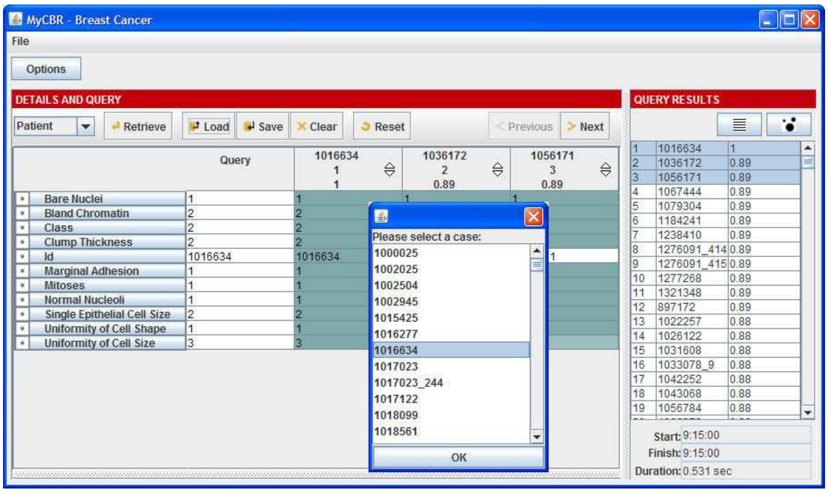
	Query	145447 1 0.93	492268 2 0.93	412300 3 0.92
Id	_undefined_	145447	492268	412300
Class	_undefined_	4	[illegible]	[illegible]
Bare Nuclei	_undefined_	9	[illegible]	[illegible]
Bland Chromatin	7	3	[illegible]	[illegible]
Uniformity of Cell Shape	5	4	6	5
Uniformity of Cell Size	4	4	4	4
Single Epithelial Cell Size	2	2	2	3
Normal Nucleoli	3	3	3	3
Mitoses	1	1	1	1
Marginal Adhesion	1	1	1	4
Clump Thickness	8	8	10	10

[details] Similarity = [1]
[details] Class = [Patient], mode = weighted_sum
[details] similarity = [0.93]

1	145447	0.93
2	492268	0.93
3	412300	0.92
4	1226612	0.9
5	1231706	0.9
6	1113906	0.9
7	1238948	0.89
8	1080233	0.89
9	1173347_114	0.89
10	76389	0.89
11	806423	0.89
12	809912	0.89
13	1243256	0.88
14	428903	0.88
15	1099510	0.87
16	1200892	0.87
17	320675	0.86
18	320675_264	0.86
19	1168359	0.86
20	1230688	0.86
21	706426	0.85
22	760239	0.85
23	1081791	0.85
24	1096352	0.85
25	1257648	0.85
26	1113038	0.83
27	1182404_257	0.83
28	752904	0.83

Start: 8:03:02
Finish: 8:03:02
Duration: 0.703 sec

(c) Retrieval of a Case Query with a Missing Attribute Value



	Query	1016634 1 1	1036172 2 0.89	1056171 3 0.89
Bare Nuclei	1	1	1	1
Bland Chromatin	2	2		
Class	2	2		
Clump Thickness	2	2		
Id	1016634	1016634		1
Marginal Adhesion	1	1		
Mitoses	1	1		
Normal Nucleoli	1	1		
Single Epithelial Cell Size	2	2		
Uniformity of Cell Shape	1	1		
Uniformity of Cell Size	3	3		

Please select a case:
1000025
1002025
1002504
1002945
1015425
1016277
1016634
1017023
1017023_244
1017122
1018099
1018561

1	1016634	1
2	1036172	0.89
3	1056171	0.89
4	1067444	0.89
5	1079304	0.89
6	1184241	0.89
7	1238410	0.89
8	1276091_414	0.89
9	1276091_415	0.89
10	1277268	0.89
11	1321348	0.89
12	897172	0.89
13	1022257	0.88
14	1026122	0.88
15	1031608	0.88
16	1033078_9	0.88
17	1042252	0.88
18	1043068	0.88
19	1056784	0.88

Start: 9:15:00
Finish: 9:15:00
Duration: 0.531 sec

(d) Breast Cancer as a Stand-Alone Application

Fig. 5. Implementation in *myCBR*

provides comfortable GUIs for modeling various kinds of attribute specific similarity measures and for evaluating the resulting retrieval quality. In order to reduce also the effort of the preceding step of defining an appropriate case representation, it includes tools for generating the case representation automatically from existing raw data [22]. The novice as well as the expert knowledge engineer are supported during the development of a myCBR project through intelligent support approaches and advanced GUI functionality [22]. Downloading *myCBR* requires two steps of downloading. The first is to download *myCBR* plug-in files; this can be done directly through *myCBR* web page. The second step is to download the Protégé ontology editor; this can be done through the Protégé web page. Downloading Protégé is not an easy task. Users need to do some readings on the site to be able to select the suitable version to download. Since *myCBR* is a plug-in inside Protégé, users need to install Protégé first. It is required to have JAVA Virtual Machine installed before proceeding in installation, or users may choose to download the version that includes the JAVA. To install the *myCBR* plug-in for Protégé, users need to copy the *myCBR* plug-ins into Protégé's plug-ins directory. Then to start Protégé and create new projects, users need to enable the myCBR plug-ins from the configuration menu of Protégé. After installing and activating the myCBR plug-in, the user interface of Protégé is extended with additional tabs to access the *myCBR* modules. After developing a CBR application using the Protégé plug-in, *myCBR* can also be used as a stand-alone Java module, to be integrated in arbitrary applications, for example, JSP5-based web applications. In this application phase, the retrieval engines of *myCBR* just read the XML files of the created project generated using the plug-in interface and perform the similarity-based retrieval [24]. For Protégé manuals and tutorial, users may consult the documentation section of the Protégé web site for available documentation. Among other things, users may find the Protégé User's Guide, a "getting started" tutorial, and information on ontology development. The manual for *myCBR* is available on its web page as HTML version or a PDF version. The manual covers installation and different usage issues. No multimedia tutorials are available for the usage of *myCBR*.

2) Implementation: Four steps are required to develop a CBR application:

- Generation of case representations
- Modeling similarity measures
- Testing of retrieval functionality
- Implementation of a stand-alone application

Generation of case representations: One powerful feature provided by *myCBR* is the easiness of the case representation by CSV data import module [24]. Users have the choice to import data instances in an existing Protégé class or to create a new class that is suitable for their raw data. Figure 5(a) shows how Wisconsin dataset is arranged in a CSV file. *myCBR* allows also slots to be added manually using Protégé. Figure 5(b) shows myCBR screen after importing the dataset into a new class Patient which will be used as query and case values for retrieval step.

Modeling of similarity measure: *myCBR* follows the local-global approach which divides the similarity definition into a set of local similarity measures for each attribute, a set of attribute weights, and a global similarity measure for calculating the final similarity value. This means, for an attribute-value based case representation consisting of n attributes, the similarity between a query q and a case c may be calculated as follows

$$Sim(q,c) = \sum_{i=1}^{N} w_i \times Sim_i(q_i, c_i) \qquad (1)$$

Here, sim_i and w_i denote the local similarity measure and the weight of attribute i, and Sim represents the global similarity measure [24]. The dataset used in this experiment is simple so we leave the similarity measure definition as the default of myCBR. We only change the weight values of the Id and Class slots from one to zero. However, users may consult myCBR tutorial for more options in defining local and global similarity measure.

Testing of retrieval functionality: *myCBR* includes an easy to use GUI for performing retrievals and for analyzing the corresponding results. By providing similarity highlighting and explanation functionality, *myCBR* supports the efficient analysis of the outcome of the similarity computation. We tested the 16 records that are excluded from the dataset according to one missing value. Only two missed classifications are obtained. Figure 5(c) shows one query of these records after retrieving the most similar cases. Another alternative of performing case retrieval is to use a query from cases. This is also tested and gives a similar result as shown in Figure 5(d).

Implementation of stand-alone application: *myCBR* can also be used as a stand-alone Java module, to be integrated in arbitrary applications. In this application phase, the retrieval engines of *myCBR* just read the XML files of the created project generated using the plug-in interface and perform the similarity-based retrieval. Figure 5(d) shows the breast cancer stand-alone application.

IV. Discussion and Conclusion

In this paper, we examined two object-oriented ontology based CBR frameworks *jCOLIBRI* developed by *GAIA* group and myCBR developed by *DFKI* group. A breast cancer classifier is built by using the two selected frameworks.

During the implantation of the breast cancer diagnostic application using jCOLIBRI we found that *jCOLIBRI* is user-friendly and efficient to develop a quick application. The classifier was successful in classification of the selected data set. During the implantation of the breast cancer classifier using *myCBR* we noticed that *myCBR* is a really a tool for rapid prototyping of a new CBR application. In seconds, users may have a running standalone CBR application by using the CSV importing feature. *myCBR* is intelligent enough to build the case structure and the case base by parsing the provided CSV file. *myCBR* avoids reinventing the wheel by making the development of a new CBR application done inside Protégé. The classifier was successful in classification of the selected data set.

In conclusion, two CBR frameworks are very useful to develop CBR base breast cancer classifier that can play a very important role to help for early detecting the disease and hence right medications can be used to save lives.

References

[1] A. Aamodt and E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variation and System Approaches," *AICOM,* vol. 7, no. 1, 1994, pp. 39–58.

[2] J. J. Bello-Tomás, J. A. Gonzĺaez-Calero and B. Dïáz-Agudo, "JCOLIBRI: An Object-Oriented Framework for Building CBR Systems," *in Advances in Case-Based Reasoning, Lecture Notes in Computer Science,* Springer Berlin/ Heidelberg, vol. 3155, 2004, pp. 32–46.

[3] S. Bogaerts and D. Leake, "Increasing AI Project Effectiveness with Reusable Code Frameworks: A Case Study Using IUCBRF," *in Proceedings of the 18th International Florida Artificial Intelligence Research Society Conference,* Menlo Park, CA: AAAI Press, 2005.

[4] S. Bogaerts and D. Leake, "A Framework for Rapid and Modular Case-Based Reasoning System Development," *Technical Report,* TR 617, Computer Science Department, Indiana University, Bloomington, IN, 2005.

[5] B. Dïáz-Agudo, P. A. Gonzïález-Calero, J. Recio-Garcïá and A. Sanchez-Ruiz, "Building CBR systems with jCOLIBRI," *Journal of Science of Computer Programming,* vol. 69, no 1–3, 2007, pp. 68–75.

[6] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubezy, H. Eriksson, N. F. Noy and S. W. Tu, "The evolution of Protege an environment for knowledge-based systems development," *Int. J. Hum.-Comput. Stud,* vol. 58(1), 2003, pp. 89–123.

[7] J. A. Gonzïález-Calero and B. Dïáz-Agudo, "An architecture for knowledge intensive CBR systems," in E. Blanzieri and L. Portinale, editors, *Advances in Case-Based Reasoning (EWCBR–00),* Springer-Verlag, Berlin Heidelberg New York.

[8] J. A. Gonzïález-Calero and B. Dïáz-Agudo, "CBROnto: a task/method ontology for CBR," in S. Haller and G. Simmons, editors, *Procs. of the 15th International FLAIRS–02 Conference (Special Track on CBR, 101–106).* AAAI Press.

[9] M. Jaczynski and B. Trousse, "An Object-Oriented Framework for the Design and the Implementation of Case-Based Reasoners," *in Proceedings of the 6th German Workshop on Case-Based Reasoning,* Berlin, 1998.

[10] R. Johnson and B. Foote, "Designing reusable classes," *Journal of Object-Oriented Programming,* vol. 1(5), 1988, pp. 22–35.

[11] J. L. Kolodner, *Case-Based Reasoning*, 1993, Morgan Kaufmann Publishers, California.

[12] D. Leake, *Case Based Reasoning. Experiences, Lessons and Future Directions,* AAAI Press, MIT Press, USA, 1997.

[13] M. Manago, R. Bergmann, N. Conruyt, R. Traph ner, J. Pasley, J. Le Renard, F. Maurer, S. Wes, K. D. Althoff and S. Dumont, "CASUEL: a common case representation language," *ESPRIT project 6322,* 1994. Task 1.1, Deliverable D1.

[14] O. L. Mangasarian and W. H. Wolberg, "Cancer diagnosis via linear programming," *SIAM News,* vol. 23, no. 5, 1990, pp. 1–18.

[15] A. Mulder, "Developing a Reusable Application Framework," *Chariot Solutions,* http://www.chariotsolutions.com/javalab/presentations.jsp, 2003.

[16] C. A. Pena-Rayes and M. Sipper, *Applying Fuzzy CoCo to Breast Cancer Diagnosis,* IEEE, 2000, pp. 1168-1175.

[17] J. A. Recio-Garcïá, B. Dïáz-Agudo and P. A. Gonzïález-Calero, "Prototyping recommender systems in jCOLIBRI," *in Proceedings of the 2008 ACM Conference on Recommender Systems (Lausanne, Switzerland, October 23 - 25, 2008),* RecSys' 08, ACM, New York, NY, pp. 243-250.

[18] J. A. Recio-Garcïá, B. Dïáz-Agudo and P. A. Gonzïález-Calero, *jCOLIBRI2 Tutorial,* 2008. Group of Artificial Intelligence Application (GAIA). University Complutense of Madrid. Document Version 1.2.

[19] J. A. Recio-Garcïá, D. Bridge, B. Dïáz-Agudo and P. A. Gonzïález-Calero, "CBR for CBR: A Case-Based Template Recommender System," in K. D. Althoff and R. Bergmann, editors, *Advances in Case-Based Reasoning,* 9th European Conference, ECCBR 2008 (in press), LNCS. Springer.

[20] J. A. Recio-Garcïá, B. Dïáz-Agudo, , A. Sïánchez and P. A. Gonzïález-Calero, "Lessons learnt in the development of a CBR framework," in M. Petridis, editor, *Proceedings of the 11th UK Workshop on Case Based Reasoning,* CMS Press, University of Greenwich, 2006, pp. 60–71.

[21] J. A. Recio-Garcïá, A. Sïánchez, B. Dïáz-Agudo and P. A. Gonzïález-Calero, "jCOLIBRI 1.0 in a nutshell. A software tool for designing CBR systems," in M Petridis, editor, *Proccedings of the 10th UK Workshop on Case Based Reasoning,* CMS Press, University of Greenwich, 2005, pp. 20–28.

[22] T. R. Roth-Berghofer and D. Bahls *Explanation Capabilities of the Open Source Case-Based Reasoning Tool myCBR,* 2008.

[23] S. Schulz, "CBR-Works: A state-of-the-art shell for case-based application building," in Melis, E., ed., *Proceedings of the 7th German Workshop on Case-Based Reasoning,* GWCBR'99, Wurzburg, Germany, University of Wurzburg, pp. 166–175.

[24] A. Stahl and T. R. Roth-Berghofer, "Rapid prototyping of CBR applications with the open source tool myCBR," in R. Bergmann and K. D. Altho, eds., *Advances in Case-Based Reasoning,* 2008, Springer Verlag.

[25] M. Sewak, P. Vaidya, C. C. Chan and Z. H. Duan, *SVM Approach to Breast Cancer Classification,* IMSCCS, vol. 2, 2007, pp. 32–37.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 11–17

ISBN 978-83-60810-27-9
ISSN 1896-7094

Using data mining for assessing diagnosis of breast cancer

Dr. Medhat Mohamed Ahmed Abdelaal
Statistics and Mathematics Department,
Faculty of Commerce, Ain Shams University.
medhatal@hotmail.com

Muhamed Wael Farouq
Statistics and Mathematics Department,
Faculty of Commerce, Ain Shams University.
m.wael.farouq@gmail.com

Prof. Dr. Hala Abou Sena
Faculty of Medicine, Ain Shams University.

Prof. Dr. Abdel-Badeeh Mohamed Salem
Faculty of Computer Science, Ain Shams University.

Abstract—**The capability of the classification SVM, Tree Boost and Tree Forest in analyzing the DDSM dataset was investigated for the extraction of the mammographic mass features along with age that discriminates true and false cases. In the present study, SVM technique shows promising results for increasing diagnostic accuracy of classifying the cases witnessed by the largest area under the ROC curve (area under empirical ROC curve =0.79768 and area under binomial ROC curve =0.85323) comparable to empirical ROC and binomial ROC of 0.57575 and 0.58548 for tree forest while least empirical ROC and binomial ROC of 0.53452 and 0.53882 was accounted by tree boost. These results are confirmed by SVM average gain of 1.7323, tree forest average gain of 1.5576 and tree boost average gain of 1.5718.**

***Keywords*- Breast Cancer, Classification Support Vector Machine (SVM), Decision Tree, Receiver Operating Characteristic Curve (ROC), Tree Boost, Tree Forest, Gain.**

I. Introduction

CANCER can develop when cells in a part of the body begin to grow out of control. These extra cells form a mass of tissue, called a growth or tumor. Tumors can be benign or malignant. The scientific discipline whose goal is the classification of objects into a number of categories or classes can be called Pattern Recognition. Objects can be images, signal waveforms or any type of measurement that needs to be classified [11].

The importance of the study is that; the breast cancer refers to life-threatening malignancies that develop in one or both breasts and is the most common form of cancer among women in developed countries. According to American Cancer Society one in eight women will develop breast cancer during their lifetime.

The problem with Breast Cancer Diagnosis is that despite radiographic breast imaging and screening has allowed for more accurate diagnosis of breast cancer, 10% to 30% of malignant cases are not detected for various reasons. There are two errors typical in examining mammograms. They are False Positives (FP) and False Negatives (FN).

The Computer-Aided Diagnosis (CAD) can reduce both the FP and the FN diagnosis rates. CAD is an application of pattern recognition aiming at assisting doctors in making diagnostic decisions. The final diagnosis is made by the doctor. Our aim is to utilize a pattern recognition system in order to assist radiologist with a "Second" opinion by concluding the mammographic mass features that most indicates malignancy.

II. Data Set

This part of the study includes shedding light on the case study used and the collected data description.

An image may be defined as a two-dimensional function, $f(x, y)$, where x and y are spatial (plane) coordinates, and the amplitude of f at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a digital image. The field of digital image processing refers to processing digital images by means of a digital computer. Note that a digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as picture elements, image elements and pixels. Pixel is the term most widely used to denote the elements of a digital image [11].

Computers cannot handle continuous images but only arrays of digital numbers. Thus it is required to represent images as two-dimensional arrays of points. A point on the 2-D grid is called a pixel or pel. Both words are abbreviations of the word picture element. A pixel represents the irradiance at the corresponding grid position. In the simplest case, the pixels are located on a rectangular grid; the position of the pixel is given in the common notation for matrices. The first index, m, denotes the position of the row, the second, n, the position of the column [4].

The principal goal of the image segmentation process is to partition an image into regions of interest that are homogeneous with respect to one or more homogeneity criteria(s) or features. Segmentation is an important tool in medical image processing, and it has been useful in many applications. A segmentation algorithm, in a mammographic context, is an algorithm used to detect something, usually the whole breast or a specific kind of abnormalities, like micro-calcifications or masses [8].

A wide variety of segmentation techniques have been proposed. However, there is no one standard segmentation technique that can produce satisfactory results for all imaging applications. The definition of the goal of segmentation varies according to the goal of the study and the type of image data. Different assumptions about the nature of the analyzed images leads to the use of different algorithms [7].

This section provides the general mammographic image information explaining the mammographic abnormalities then introduces techniques, operations and statistics that form the tools for the development of comprehensive analysis/ diagnosis algorithms.

The digital database for screening mammography (DDSM) is a resource for use by the mammographic image analysis research community. The primary purpose of the database is to facilitate sound research in the development of computer algorithms to aid in screening. Secondary purposes of the database may include the development of algorithms to aid in the diagnosis and the development of teaching or training aids. The database contains approximately 2,500 studies [15].

The most effective method of early detection of the breast cancer is mammograms, certain characteristics in the mammograms determines whether cancer exists or not, breast cancer often presents as a mass with or without presence calcifications.

The location, size, shape, density, and margins of the mass are useful for the radiologist in evaluating the likelihood of cancer. Most benign masses are well circumscribed, compact, and roughly circular or elliptical. Malignant lesions usually have a blurred boundary, an irregular appearance, and sometimes are surrounded by a radiating pattern of linear spicules. Masses are categorized by their shape, density, and margins.

The mass shape is described with a four-point assessment: round, oval, lobular and irregular. The mass margins modify the boundaries. For example the overall shape may be round, but close inspection may reveal scalloping along the border, which may indicate a degree of irregularity or a lobular characteristic. The margins are rated with a 5-point system: circumscribed (well-defined or sharply-defined) margins, microlobulated margins, obscured margins, indistinct margins and spiculated margins as shown in Fig. 1.

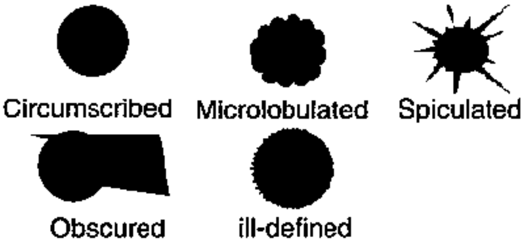


Fig. 1 Mass descriptors for margin

The intensity or the x-ray attenuation of the mass tissue region is described as density. The density here is the relative density, i.e. higher, lower or similar relative to the surrounding tissue. The density is rated on 4-point system:

High density, equal density, low density (lower attenuation, but not fat containing) and fat containing – radiolucent [10].

Numerous statistics can be developed from digital images which aid in describing and analyzing images. This section provides an introduction to a selected group of image statistics. The selected group represents those used in this research; however, they are far from exhaustive. In fact, a significant amount of researches continues developing new statistics for describing and analyzing images. This research is primarily interested in the application of existing statistics. .

(1) Age

(2) Variance

The variance of an image is a measure of the variation of pixel intensities in the image.

(3) Area

The area is defined as the total number of pixels belonging to the object.

(4) X-centroid, Y-centroid

The coordinates of the geometrical center of the object defined with respect to the image origin.

$$x_{centroid} = \sum_{i \in object} i \div A \qquad (1)$$

$$y_{centroid} = \sum_{j \in object} j \div A \qquad (2)$$

where i and j are image pixel coordinates and A is the area of the segment or region of interest (ROI) [9].

(5) Compactness

Compactness is another measure of the object's roundness and is calculated as:

$$compactness = p^2 \div 4\pi \mathrm{A} \qquad (3)$$

where P is the object perimeter. Compactness gives the minimal value 1 for circles [12].

(6) Circularity

Area and perimeter are two parameters which describe the size of an object. In order to compare objects which are observed from different distances, it is important to use shape parameters which do not depend on the size of the object on the image plane. The circularity c is defined as:

$$c = p^2 \div A \qquad (4)$$

The circularity is a dimensionless number with a minimum value of 4π ≈ 12.57 for circles. The circularity is 16 for a square and 12√3 ≈ 20.8 for an equilateral triangle. Generally, it shows large values for elongated objects.

(7) Eccentricity

This is a measure similar to the circularity but with a better defined range. The parameter is extracted from the second-order moments as:

$$\varepsilon = \frac{\left(m_{2,0} - m_{0,2}\right)^2 + 4\mathrm{m}_{1,1}^2}{\left(m_{2,0} + m_{0,2}\right)^2} \qquad (5)$$

The eccentricity ranges from 0 to 1, it is zero for circular object and one for line shaped object [3].

III. Statistical Techniques

One of the most useful applications of statistical analysis is the development of a model to explain the relationship between the variables; many types of models have been developed, including classification support vector machines, tree boost and tree forest. This part will focus on the deployed analytical techniques.

SUPPORT VECTOR MACHINES

Methods for analyzing and modeling data can be divided into groups; supervised learning and unsupervised learning. Supervised learning requires input data that has both independent variables and a dependent (target) variable whose value is to be estimated. By various means, the process learns how to model predict the value of some variable, then supervised learning is recommended approach.

Unsupervised learning does not identify a dependent variable (target), but rather treats all of the variables equally. In this case, the goal is not to predict the value of a variable but rather to look for patterns, groupings or other ways to characterize the data that may lead to understanding of the way the data relate. Cluster analysis, correlation, factor analysis (principle components analysis) and statistical measures are examples of unsupervised learning.

One of the most useful applications of statistical analysis is the development of a model to represent and explain the relationship between variables. One of the best state-of-the-art modeling methods including support vector machines (SVM).

In the manner of speaking of SVM literature, a predictor variable is called an attribute, and a transformed attribute that is used to define the hyper plane is called a feature. The task of choosing the most suitable representation is known as feature selection. A set of features that describes one case (i.e., a row of predictor values) is called a vector. So the goal of SVM modeling is to find the optimal hyper plane that separates clusters of vector in such a way that cases with one category of the target variable are on one side of the plane and cases with the other category are on the other side of the plane. The vectors near the hyper plane are support vectors.

To illustrate classification SVM, let us assume that there is a linear relationship with N observations. The independent variable is x_i and the dependent variable is y_i given that i=1, 2, 3……N. The goal of classification SVM is to produce a linear function which can make the best fit of the dependent variable y_i. The linear function can be expressed as follows:

$$y = f(x) = \langle b \cdot x \rangle + a \qquad (6)$$

Where a and b are the classification parameters and is the dot product of b and x.

The dot product of two vectors is defined as:

$$b \cdot x = \sum_{i=1}^{N} b_i x_i = b_1 x_1 + b_2 x_2 + \dots . b_N x_N$$

The optimum classification function can be found by minimizing the following function:

$$M(b, S) = 0.5\|b\|^2 + C\sum_{i=1}^{N}\left(S_i^- + S_i^+\right) \qquad (7)$$

The constraint can be as follows:

$$\begin{aligned} & y_i - \langle b, x_i \rangle - a \leq \varepsilon + S_i \\ & \langle b, x_i \rangle + a - y_i \leq \varepsilon + S^*_i \qquad (8) \\ & S_i, S_i^* \geqslant 0 \end{aligned}$$

Where C is a constant greater than zero, and can be used to determine the trade off the smoothness of f and the amount up to which deviations larger than ε are accepted. and S_i^+ are two slack variables, which can be used to represent the upper and the lower constraints of the classification function. is norm of b. the norm can be as follows:

$$\|b\| = \sqrt{b_1^2 + b_2^2 + \dots b_N^2}$$

To find the optimum solution of M function, loss function must be determined. Loss function is a function shows the maximum allowed deviation of the predicted values from the observed one. The most recommended loss functions are four. These four functions are: Huber, ε-insensitive, quadratic and Laplace [13]. Fig. 2 shows the difference between the four types of loss functions.

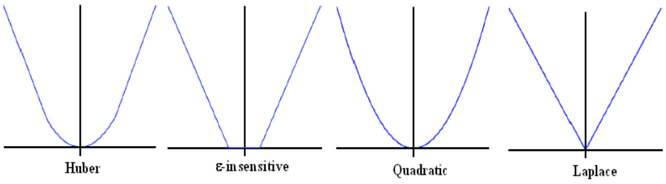


Fig. 2 The four types of loss functions

The first loss function is Huber loss function: it is a robust loss function that has optimal properties when the underlying distribution of the data is unknown. The second one is ε-insensitive loss function; it is an approximation to Huber's loss function but it can reduce sensitivity to the outliers. The third one is quadratic loss function: it corresponds to the predictable least squares error measure. The fourth one is Laplace loss function: it is less sensitive to outliers than the quadratic loss function.

In this paper, the ε-insensitive loss function was selected. For classification SVM model which depends on ε-insensitive loss function: the difference between the estimated values and the observed values of the dependent variable y_i can be calculated. If the difference is less than ε, the classification function is considered to be most popular and accurate [13]. The ε-insensitive loss function can be expressed as follows:

$$|S|_{\varepsilon} = \begin{cases} o & if\ |S| < \varepsilon \\ |S| - \varepsilon & otherwise \end{cases} \qquad (9)$$

Most modeling techniques are trying to find the best fit between the observed and predicted values, however, SVM'S ε-insensitive loss function focuses on optimizing a bound around the classification function thus making it stronger against the outliers.

To find the solution of equation 7, Lagrange optimization must be used, the solution to this optimization problem can be expressed as follows: Minimizes the target function as follows:

$$\varepsilon \sum_{i=1}^{N} \left(\alpha_i + \alpha_i^*\right) + 0.5 \sum_{i=1,j=1}^{N} \left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right)\left(x_i \cdot x_j\right) + \sum_{i=1}^{N} y_i\left(\alpha_i - \alpha_i^*\right) \qquad (10)$$

The constraints can be as follows:

$$\sum_{i=1}^{N} \left(\alpha_i - \alpha_i^*\right) = 0{,}0 \leq \alpha_i,\ \alpha_i^* \leq C \qquad (11)$$

SVM models are built around a kernel function that transforms the input data into an n-dimensional space where a hyper plane can be constructed to partition the data.

There are four kernel functions, linear, polynomial, radial basis function (RBF) and sigmoid (S-shaped). There is no way in advance to know which kernel function will be best for an application.

The RBF kernel non-linearity maps samples into a higher dimensional space, so it can handle nonlinear relationships between target categories and predictor attributes; a linear basis function cannot do this. Furthermore, the linear kernel is a special case of the RBF. A sigmoid kernel behaves the same as a RBF kernel for certain parameters. The RBF function has fewer parameters to tune than polynomial kernel, and the RBF kernel has less numerical difficulties.

This is the case of linear classification, but to illustrate the classification case of non-linearity, the data must be firstly linearized by mapping it into a higher dimensional space, called "feature space" by using kernel functions, that linear classification functions can be applied. The most recommended kernel function is the RBF Kernels [6]. The RBF kernel is defined as:

Given that is a kernel parameter.

Insert kernel function in the previous model (10), this model can be adjusted as follows:

The same constraints in (8) can be used, minimize:

$$\varepsilon \sum_{i=1}^{N} \left(\alpha_i + \alpha_i^{¿}\right) + 0.5 \sum_{i=1,j=1}^{N} \left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right) K\left(x_i \cdot x_j\right) + \sum_{i=1}^{N} y_i\left(\alpha_i - \alpha_i^*\right) \qquad (12)$$

DECISION TREES

A decision tree is a logical model represented as a binary (two-way split) tree that shows how the value of a target variable can be predicted by using the values of a set of predictor variables. There are many techniques for decision trees. In this paper the authors selected two techniques which are Boosting Trees and Tree Forest.

A decision tree can be used to predict the values of the target variable based on values of the predictor variables.

Each node represents a set of records (rows) from the original dataset. Nodes that do not have child nodes are called "terminal" or "leaf" nodes. The topmost node called the "root" node. Unlike a real tree, decision trees are drawn with their root at the top. The root node represents all of the rows in the dataset.

A decision tree is constructed by a binary split that divides the rows in a node into two groups (child nodes). The same procedure is then used to split the child groups. This process is called "recursive partitioning".

DECISION BOOSTING TREES

"Boosting" is a technique for improving the accuracy of a predictive function by applying the function repeatedly in a series and combining the output of each function with weighting so that the total error of the prediction is minimized. In many cases, the predictive accuracy of such a series greatly exceeds the accuracy of the base function used alone.

The TreeBoost algorithm is optimized for improving the accuracy of models built on decision trees. Research has shown that models built using TreeBoost are among the most accurate of any known modeling technique.

The TreeBoost algorithm is functionally similar to Decision Tree Forests because it creates a tree ensemble, and it uses randomization during the tree creations. However, a random forest builds the trees in parallel and they "vote" on the prediction; whereas TreeBoost creates a series of trees, and the prediction receives incremental improvement by each tree in the series.

Mathematically, a TreeBoost model can be described as:

$$\mathrm{PT} = F_0 + \mathrm{B}_1 T_1(X) + \mathrm{B}_2 T_2(X) + . + B_M T_M(X) \qquad (13)$$

Where *PT* is the predicted target, *F*$_0$ is the starting value for the series (the median target value for a regression model), *X* is a vector of "pseudo-residual" values remaining at this point in the series, $T_1(X)$, $T_2(X)$ are trees fitted to the pseudo-residuals and B_1, B_2, etc. are coefficients of the tree node predicted values that are computed by the TreeBoost algorithm.

The first tree is fitted to the data. The residuals from the first tree are then fed into the second tree which attempts to reduce the error. This process is repeated through a chain of successive trees. The final predicted value is formed by adding the weighted contribution of each tree.

Usually, the individual trees are fairly small, but the full TreeBoost additive series may consist of hundreds of these small trees. TreeBoost models often have a degree of accuracy that cannot be obtained using a large, single-tree model. TreeBoost models are often equal to or superior to any other predictive functions including neural networks. TreeBoost models can handle hundreds or thousands of potential predictor variables. Irrelevant predictor variables are identified automatically and do not affect the predictive model. TreeBoost uses the Huber M-regression loss function which makes it highly resistant to outliers and misclassified cases. TreeBoost procedures are invariant under all (strictly) monotone transformations of the predictor variables. So transformations such as (a*x+b), log(x) or exp(x) do not affect the model. Hence, there is no need for input transformations. The sophisticated and accurate method of surrogate splitters is used for handling missing predictor values. The stochastic element in the TreeBoost algorithm makes it highly resistant to over fitting. Cross-validation and random-row-sampling methods can be used to evaluate the generalization of a TreeBoost model and guard against over fitting. TreeBoost can be applied to regression models and k-class classification problems. TreeBoost can handle both continuous and categorical predictor and target variables.

DECISION TREE FOREST

A Decision Tree Forest is an ensemble of decision trees whose predictions are combined to make the overall prediction for the forest. A decision tree forest is similar to a TreeBoost model in the sense that a large number of trees are grown. However, TreeBoost generates a series of trees with the output of one tree going into the next tree in the series. In contrast, a decision tree forest grows a number of independent trees in parallel, and they do not interact until after all of them have been built.

Both TreeBoost and decision tree forests produce high accuracy models. Decision tree forests use the out of bag data rows for validation of the model. This provides an independent test without requiring a separate data set or holding back rows from the tree construction. The sophisticated and accurate method of surrogate splitters is used for handling missing predictor values. The stochastic element in the decision tree forest algorithm makes it highly resistant to over fitting. Decision tree forests can be applied to regression and classification models.

The primary disadvantage of decision tree forests is that the model is complex and cannot be visualized like a single tree. It is more of a “black box” like a neural network. Because of this, it is advisable to create both a single-tree and a decision tree forest model. The single-tree model can be studied to get an intuitive understanding of how the predictor variables relate, and the decision tree forest model can be used to score the data and generate highly accurate predictions.

IV. Analysis and Results

The measures in Table I were used for comparison purposes. These measures can be used as an indicator of the mean difference between the measured and estimated values.

Table I.
The Results of Applying Classification SVM and Decision Trees

Measures	Classification SVM		Forest Tree		Boosting Tree	
	Training	Validation	Training	Validation	Training	Validation
CV	0.796	0.922	0.852	0.958	0.922	0.988
NMSE	0.479	0.644	0.550	0.695	0.643	0.738
MSE	0.118	0.158	0.135	0.170	0.158	0.181
MAE	0.331	0.346	145.125	169.396	155.113	181.055
MAPE	32.870	37.414	37.675	39.542	47.142	53.658
Var%	36%	30%	42%	36%	69%	58%

Root mean squared error (RMSE) is a quadratic scoring rule which measures the average magnitude of the error. RMSE is the difference between predictions and corresponding observed values are each squared and then averaged over the sample. Finally, the square root of the average is taken. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. This means that RMSE is most useful when large errors are particularly undesirable.

The normalized root mean squared error (NMSE) is the RMSE divided by the range of observed values; the value is often expressed as a percentage, where lower values indicate less residual variance.

Mean absolute error (MAE) measures the average magnitude of the errors in a set of predictions, without considering their direction. The MAE is the average over the verification sample of the absolute values of the differences between predictions and the corresponding observation. The MAE is a linear score which means that all the individual differences are weighted equally in the average.

Proportion of variance (Var%) explained by model variables; this is the best single measure of how well the predicted values match the actual values. If the predicted values exactly match the actual values, then the model would explain 100% of the variance.

According to the previously discussed measures of C.V, NMSE, MSE, MAE and MAPE, the classification SVM was the optimum technique declared by the lowest values of 0.796, 0.479, 0.118, 0.331 and 32.870 respectively for the training set among techniques applied and validated with values of 0.922, 0.644, 0.158, 0.346 and 37.414 respectively.

To determine the importance of the independent variables of the suggested model; the misclassification rate for the model using the actual data values for all predictors must be calculated. Then for each predictor, it randomly rearranges the values of the predictor and computes the misclassification rate for the model using the rearranged values. The difference between the misclassification rate with the correctly ordered values and the misclassification rate for the rearranged values is used as the measure of importance of the predictor. Table II shows the variables importance for both techniques.

Table II.
The Variables Importance of Classification SVM and DEcision Tree

Variable	SVM	Forest Tree	Boosting Tree
Age	100	100	100
Variance	22.462	49.36	36.11
Circularity	11.044	45.758	34.152
Compactness	11.018	50.444	31.032
Eccentricity	8.72	35.175	17.044
Centroid X	7.66	40.385	23.212
Centroid Y	3.48	36.636	17.332
Area	1.429	36.503	18.607

The importance score for the most important predictor is scaled to a value of 100. Other predictors will have lower scores. Variance, Circularity, Compactness, Eccentricity, Centroid X, Centroid Y and Area are the less important variables for both SVM and Boosting tree, their importance ranges from 1.429 to 36.11 which is less than 40, then these variables could be ignored in the analysis [14] while Age, Compactness, Circularity and Centroid X possess the highest importance for the Forest Tree with importance of 100, 50.444, 49.36, 45.758 and 40.385 respectively.

The lift and gain is a useful tool for measuring the value of a predictive model. The basic idea of lift and gain is to sort the predicted target values in decreasing order of purity on some target category and then compare the proportion of cases with the category in each bin with the overall proportion. The lift and gain values show how much improvement the model provides in picking out the best of the cases. A gain chart displays cumulative percent of the target value on the vertical axis and cumulative percent of population on the horizontal axis. Cumulative gain is the ratio of the expected outcome using the model to prioritize the prospects divided by the expected outcome of randomization. The straight, diagonal line shows the expected return if no model is used for the population. The curved line shows the expected return using the model. The shaded area between the lines shows the improvement (gain) from the model. The gain of 1.00 means we are not doing any selective targeting.

Figures 3, 4and 5 exploits that by applying boosting tree we get 1.5718 times better outcome than the expected return of no model or randomization comparable to an average gain of 1.5576 for forest trees while the highest average gain is due to applying SVM of 1.7323.

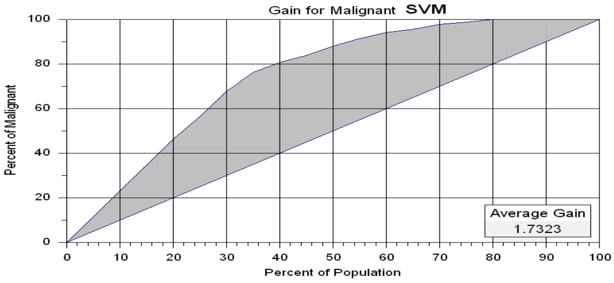


Fig. 3 The gain chart for classification SVM

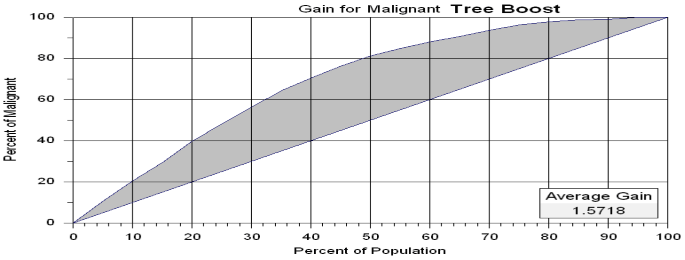


Fig. 4 The gain chart for boosting tree

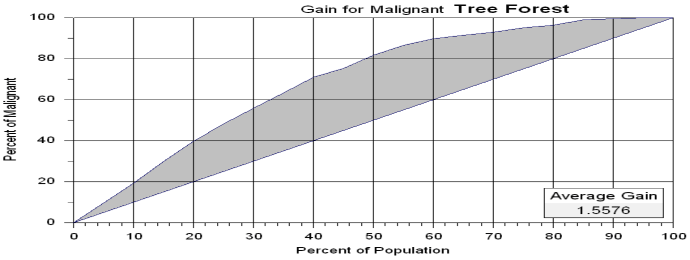


Fig. 5 The gain chart for forest tree

A receiver operating characteristic curve (ROC) curve is a graphical representation of the tradeoff between the false negative and false positive rates for every possible cutoff. Equivalently, the ROC curve is the representation of the tradeoffs between sensitivity and specificity. The area under the curve is a measure of test accuracy. It shows the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity). The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test. The slope of the tangent line at a cut point gives the likelihood ratio for that value of the test.

The accuracy of the test depends on how well the test separates the group being tested into those with and without the disease in question. Accuracy is measured by the area under the ROC curve. An area of 1 represents a perfect test; an area of 0.5 represents a worthless test.

The empirical estimate of area under the ROC curve (±SE) for the SVM was (0.79768±0.02762) while the binomial estimate of area under the ROC curve was (0.85323±0.02537). Empirical estimate of area under the ROC curve of the boosting tree was (0.53452±0.03474) while the binomial estimate of area under the ROC curve was (0.53882±0.03900). Empirical estimate of area under the ROC curve of the forest tree was (0.57575±0.03439) while the binomial estimate of area under the ROC curve was (0.58548±0.03843).As discussed above and illustrated in fig. 6, SVM possess the highest area under the curve among the discussed techniques.

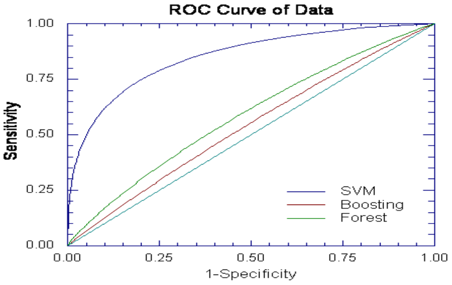


Fig. 6 The ROC for SVM, Forest Trees and Boosting Trees.

IV. Conclusion and Future Work

The results of applying the three classification techniques for extraction of the most important mammographic mass features showed promising and superior results for classification SVM over decision trees witnessed by minimal error measures ans maximum average gain.

Other risk factors could be used to aid the analysis such as genetic risk factors, previous breast radiation, and previous abnormal breast biopsy. Other proposed mammographic mass features includes: center of gravity, sphericity, inertia-shape, mean radius and max radius. Automated detection and classification of other types of mammographic lesions as micro calcifications and distorted architecture. The use of automated detection and segmentation techniques not only in mammographic images but also in MRI.

References

[1] Abdelaal, Medhat Mohamed , Muhamed Wael Farouq ,Hala Abou Sena, Abdel-Badeeh Mohamed Salem . *Using pattern recognition approach for providing second opinion of breast cancer diagnosis*. IEEE International Conference on Informatics and Systems, INFOS 2010.Cairo.Egypt.

[2] Abdelaal, Medhat Mohamed. *Application of regression support vector machine to estimate the fisheries parameters in Lake Nasser*. International Society for Business and Industrial Statistics Conference ISBIS-2010. Portoroz. Slovenia.

[3] Bernd Jahne, 2004, “Practical handbook on image processing for scientific and technical applications”, Florida, CRC press.

[4] Bernd Jahne. “Digital image processing”, Verlag Berlin, Heidelberg, Springer, 2002.

[5] Chang, C. and C. Lin, "A library for support vector machines". 2005, http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[6] Cristianini, N. and J. Shawe-Taylor, "An introduction to support vector machines and other kernel-based learning methods". New York, NY: Cambridge University Press, 2000.

[7] Jadwign Rogowska. “Overview and fundamentals of medical image segmentation”, Handbook of Medical Imaging, San Diego, Academic Press, 2000

[8] John Terry. “Computer assisted screening of digital mammogram images”, The Department of Computer Science, University of Southern Mississippi, 2003

[9] Mohamed Sameti. 1998, “Detection of soft tissue abnormalities in mammographic images for early diagnosis of breast cancer”

[10] Monika Shinde, 2003, “Computer aided diagnosis in digital mammography”, Department of Computer Science and Engineering, College of Engineering, University of South Florida.

[11] Rafael C. Gonzalez, Richard E. Woods. “Digital image processing”, New Jersey, Prentice Hall, 2002.

[12] Rangaraj M. Rangayyan. Biomedical image analysis, Florida, CRC Press, 2005.

[13] Smola, A.J. and A. Scholkopf. "A tutorial on support vector regression". NeuroCOLT2 Technical Report NC2-TR- 1998-030.

[14] Thomas, D. R. Zhu, P. Decady, Y. J. *Point estimates and confidence intervals for variable importance in multiple linear regression*. Journal of Educational and Behavioral Statistics, 2007, Vol 32; Num B 1, pages 61-91.

[15] University of South Florida, Digital Mammography Home Page. http:marathon.csee.usf.edu/Mammography/Database.html.2009.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Advanced scale-space, invariant, low detailed feature recognition from images - car brand recognition

Štefan Badura
University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovakia
Email: stefan.badura@fri.uniza.sk

Stanislav Foltán
University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovakia
Email: stanislav.foltan@fri.uniza.sk

***Abstract*—This paper presents analysis of a model for car brand recognition. The used method is an invariant keypoint detector - descriptor. An input for the method is a set of images obtained from the real environment. The task of car classification according its brand is not a trivial task. Our work would be a part of an intelligent traffic system where we try to collect some statistics about various cars passing a given area. It is difficult to recognize objects when they are in different scales, rotated or if they are low contrasted or when it is necessary to take into count high level of details. In our work we present a system for car brand recognition. We use scale space invariant keypoint detector and descriptor (SURF – Speeded-up Robust Features) for this purpose.**

I. Introduction

THIS paper presents a model for car brand recognition. Not many works have been published for such problem yet. We discuss a model for scale, space invariant car brand detection and recognition. The goal of our work is to provide information about car type approaching monitored region. For example car brands as “Ford”, “Kia”, “Volkswagen” etc. are considered. Recently, some promising approaches to detect foreground objects from images have been published: SIFT - Lowe (Scale Invariant Feature Transform) [1] and SURF - Bay et al. (Speeded Up Robust Features) [2]. Both methods detect interest points, called features, and also propose a method of creating an invariant descriptor for these features. The created descriptor is used as vector, uniquely identifying the found interest points. It has to be distinctive and robust for various scale-space deformations. Vectors can be used for matching of detected interest points even under a variety of disturbing conditions like scale changes, rotation, changes in illumination and viewpoints or image noise. The invariance is the most important ability of these keypoint detectors. The car brand can be in various representation in the image. An example is shown in the Fig.1.

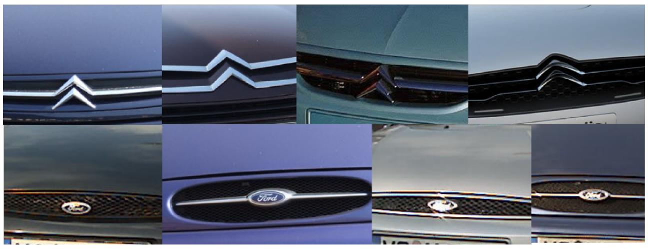


Fig 1: A series of images to show various car brand invariances for the same car brand.

The purpose of this paper is to review and investigate a method referred to as SURF. The SURF detector-descriptor method for our problem is analyzed. We use the OpenSURF library which we integrated into our software demo. In [3] we proposed a possibility to use the SIFT method for the same problem. It showed some positive results but average percentage of successful classification was not to high. The number of different car brands was 7 and the final percentage moved around 60%. Some car brands were classified with better results than others. As far as we know, no other works exist for car brand recognition problem. We focus on scale-space invariant detectors and descriptors. These seem to be a good compromise to other methods. The SIFT and SURF in various problem are used where patterns or objects from database in given scene (unknown image) are searched. Results from other types of common work are motivation for our problem. In [4] authors measure visual similarities between different visual entities with SIFT. In [5] the SIFT is used for face detection. In [6] the SURF is used for foreground detection. Comparative work done by Mikolajczyk and Schmid in [7] is a good reference for other types of interest points detectors and descriptors. The authors of SURF [2] claims SURF to be a superior to SIFT in terms of runtime execution while it is still providing good results with regards to feature point quality. The goal of this paper is not to compare SIFT against SURF or otherwise, but the goal is to analyze the possibility of using the SURF for car brand recognition from images taken in a natural outdoor environment.

In the second part the SURF is discussed in more details. First the algorithm is analyzed from a theoretical standpoint, to provide a brief overview of how it works and what was the motivation of using it. In the third part a brief overview of preprocessing procedures for region of interest specifying is discussed. The 4th proposes experiments for the problem of car brand recognition. The paper is concluded in 5th part.

II. Speeded-Up Robust Features (SURF)

The SURF is scale and rotation invariant interest point detector and descriptor. It is designed to overcome relatively low computation time which SIFT has, by providing the

978-83-60810-27-9/09/$25.00 © 2010 IEEE

same functionality. Much of the performance increase in SURF can be referred to the use of an special image representation known as the “Integral Image" [2]. The SURF is three step process:

- Interest points detection and localization.
- Feature vector construction – descriptor (interest point description).
- Descriptors matching.

A. Interest Point Detection and Localization

The SURF uses Fast – Hessian features detector and it is based on determinant *D(H)* of Hessian matrix *H(f)*. The Hessian matrix is the matrix of partial derivatives of two dimensional function *f(i,j)* (see eq. 1) [8][9].

$$H(f)=\begin{bmatrix} \frac{d^2 f}{di^2} & \frac{d^2 f}{didj} \\ \frac{d^2 f}{didj} & \frac{d^2 f}{dj^2} \end{bmatrix} \qquad (1)$$

The determinant of this matrix *(H)*, known as the discriminant, is calculated by:

$$D(H)=\frac{d^2 f}{di^2}\cdot\frac{d^2 f}{dj^2}-\left(\frac{d^2 f}{didj}\right)^2 \qquad (2)$$

Equations (1) and (2) are defined for continuous function f, but we work with images, so it means discrete space. Then *f(i,j)* as discrete function is considered which represents image. Point at *(i,j)* coordinates is a pixel intensity for the image. Approximated derivatives are computed by convolution with appropriate kernel (*H'* is approximation of *H*). For determinant approximation *(D')* formula is used:

$$D'(H')=D_{ii}.D_{jj}-(0.9\mathrm{D}_{ij})^2 \qquad (3)$$

Determinant computation is necessary for interest point detection (which are maxims in determinant matrix). In order to be able to detect interest points in scale space the notion should be introduced. A scale-space is a continuous function which can be used to find an extrema across all possible scales [10]. In SIFT scale-space as an image pyramid is used where the input image is iteratively convolved with Gaussian kernel in reduced size and so repeatedly sub--sampled. In SURF we do not need to resize images. Instead the convolution mask is re-sampled to ensure size invariance. This allows for multiple layers of the scale-space pyramid to be processed simultaneously and negates the need to subsample the image hence providing performance increase [8]. The idea is shown in Fig. 2. For appropriate and accurate interest point selection some other steps are processed. Firstly a treshold is applied, secondly non maximum supression is done and next step is interpolation. Fo more details see [2],[8].

B. Interest Point Descriptor

The SURF descriptor describes how the pixel intensities are distributed within a scale dependent neighborhood of each interest point detected by the Fast-Hessian [8]. The process of descriptor extraction consists of two parts. At first a reproducible orientation to each interest point is assigned. Secondly around each interest point a square window in appropriate scale is constructed. In this square a 64 value vector is computed which is the descriptor for given interest point. For more details see the Fig. 3. The SIFT extracts 128 values for one interest point compared to the SURF.

C. Descriptors matching

If descriptors of database items and new query are known, they are matched each other. We use simple principle and its modification. For each item in database find all keypoints. For each keypoint from the query image find the keypoint from database with the smallest distance (MSE). Two keypoints are considered as the same if the error of the best match is at least a factor of α smaller than the distance to the next closest descriptor. If not take another. Then count the number of descriptors that successfully matched.

III. Image preprocessing

In this part a brief introduction to preprocessing is discussed. The main purpose is to reduce image to a size which contains all the necessary information but the image remains as small as possible. The car brand represents just a small part compared to the whole image. Reducing an input image to the region around car brand has positive effect namely in two aspects: Lower number of detected interest points leads to more accurate computation (less error rate) and it improves the execution time. From the Fig. 4 is clear that we use license plate for image reducing into region of interest. The license plate is found and according its position a car brand is searched. We suppose that the car brand is above the license plate. It is not a rule but almost all cars satisfy this condition. In next part experiments are discussed.

IV. Experiments

In this part, we study whether SURF features are suitable and robust enough for our problem. Since we have chosen a typical pattern classification approach for solving the problem of car brand recognition, it is necessary to have a suffi-

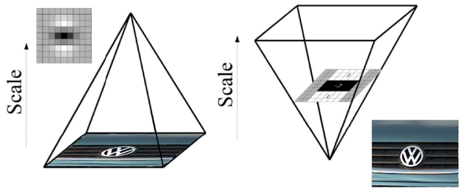


Fig 2: Pyramid. On the left - traditional approach (SIFT), on the right approach used in SURF.

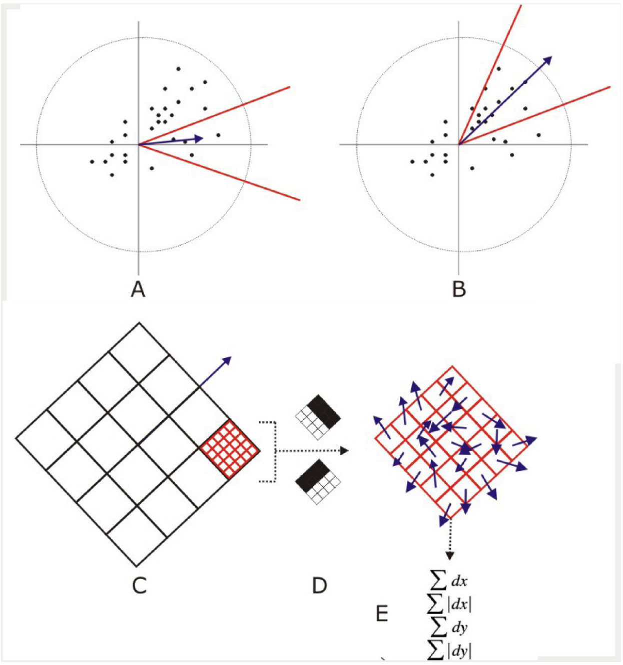


Fig 4: The SURF - descriptor. A, B represent orientation assignment. C is square around interest point with orientation. D and E show 64 value descriptor computation.

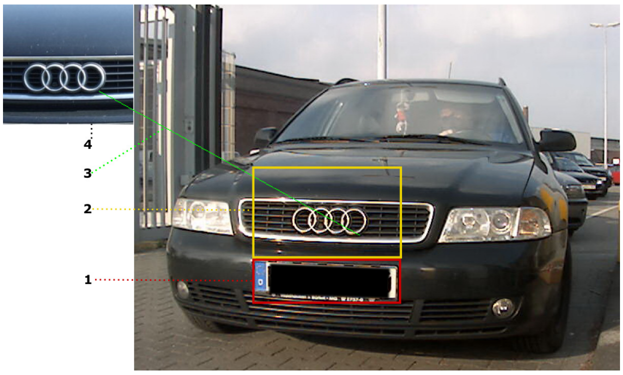


Fig 3: Process of initialization region of interest. At first license plate is found (1). Then above the license plate region of interest which holds car brand is set (2). Next the matching process is initialized (3,4) after descriptors are extracted.

Table I

pospis

Car brand	Num. Of samples
Audi	21
BMW	22
Citroen	8
Ford	16
Kia	12
Mercedes	16
Opel	24
Peugeot	14
Renault	19
Skoda	8
Volkswagen	29

cient data set. The data corpus of altogether 189 different samples was collected in the entrance to a car park. Table 1 shows number of samples for each brand. Our database consisted of 11 different car brands. Original image size was 640x480. When the region of interest was reduced the size moved around 200x200 pixels.

For given set of samples we tested the SURF method. Before experiments a template database was created. The database contained one sample for each brand. We chose the image which seemed to be optically the "best" (without noise, rotation, well visible etc.) for each class (see Fig. 5). From samples the SURF descriptor was computed and saved to file. Detected interest points which did not affect the car brand were deleted (see Fig 6.). The criteria used to evaluate

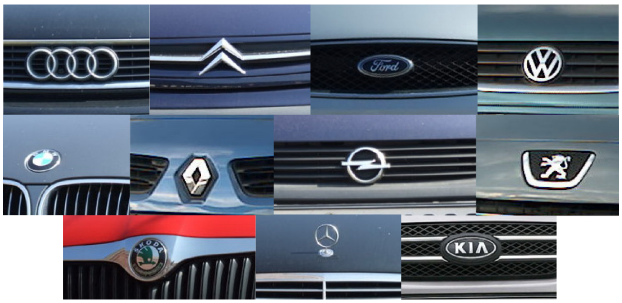


Fig 5: Samples in database. Each image represents one class. These samples are considered as templates for matching.

Table II

Experimental results.

Car brand	Result (α=0.6)	Result (α = 0.8)
Audi	0.52	0.9
BMW	0.14	0.09
Citroen	0.63	0.5
Ford	0	0
Kia	0.08	0.33
Mercedes	0.06	0
Opel	0.71	0.88
Peugeot	0.43	0.29
Renault	0.21	0.42
Skoda	0.25	0.38
Volkswagen	0.66	0.66

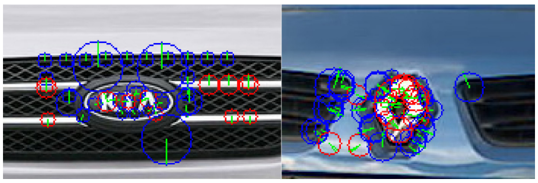


Fig 6: Detected interest points. Interest points which do not affect the car brand are deleted. Most of them describe the mask.

the performance of the detector is number of correctly matched samples from the database to samples from available testing dataset (new queries). We created tests based on average error and tests based on number of descriptor matches for certain class. The second approach provided better results but it was unfair because not all samples had the same number of detected keypoints. Summary is shown in table 2. We modified also the α constant (distance between the best and the next closest descriptor). Better results were achieved with α=0.8 (default values was 0.6) but this fact is clear because we are not so strict to coincidence. Descriptor which fitted the most produces smaller error (the same descriptors produce error equals to 0).

An average result for all samples moved around 45%. Table 1 shows interesting results. Some car brand are recognized with high percentage but the other ones are recognized poorly. The reason could be for example a small database of samples. For some car brands just low number of interest points was detected (see Fig. 7) and therefore they are recognized with insufficient result. The SURF method is quite strong tool from various aspects. Similar (the same) objects were found with almost 100%. Objects in different scales were also recognized with high percentage. Fig. 8 shows some results where the best matched keypoint as an example is depicted. Images on the left are samples from the database and images on the right are new queries.

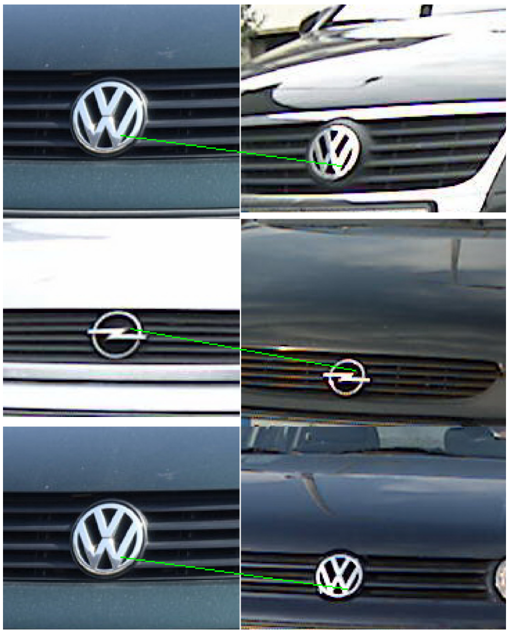


Fig 7: An example of positive matching. The same interest points were detected in different images.

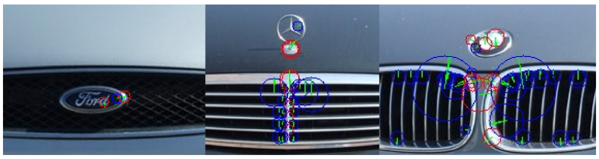


Fig 9: An example of detected interest points for some car brands. In all image low number of keypoints is detected. It is also shown that the mask is quite informative (or noisy) for some car brands (bmw and mercedes). If we would consider the mask, result could improved but the mask is not standardized.

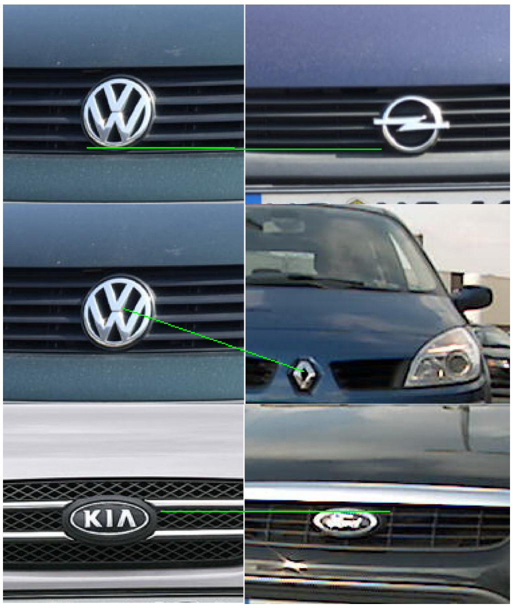


Fig 8: An example of negative matching. Even if the images are not matched, some similarities can be found for highlighted interest point.

V. Conclusion

In this work, we investigated the usage of SURF descriptors for car brand recognition. The final percentage was not very high. The main purpose was to analyze the possibility of this method for our problem and from this standpoint we obtained some valuable results. As the future work a new strategy for creating database is necessary to build. It would be also appropriate to analyze descriptor values and that especially for car brands with small amount of detected interest points. For some brands, which were recognized very poorly, different invariant key point detector should be tested (MSER – maximal stable external regions).

References

[1] D. G. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints.* In International Journal of Computer Vision, 60, 2, 2004, pp. 91–110.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. *Surf: Speeded up robust features.* Computer Vision and Image Understanding (CVIU), 2008, pp. 346–359,

[3] S. Foltán, Š. Badura. *Robust Car Brand REcognition From Camera Image.* In MENDEL - International Conference on Soft Computing, 2010.

[4] E. Shechtman, M. Irani, *Matching Local Self-Similarities across Images and Videos,* Computer Vision and Pattern Recognition, CVPR '07. IEEE Conference, 2007.

[5] Mohamed Aly. *Face Recognition using SIFT Features*, CNS186 Term Project Winter, 2006.

[6] S. Badura, A. Lieskovsky. *Intelligent traffic system: cooperation of MANET and image processing,* In First International Conference on Integrated Intelligent Computing IEEE, 2010.

[7] K. Mikolajczyk and C. Schmid. *A performance evaluation of local descriptors.* IEEE Transactions on Pattern Analysis & Machine Intelligence, pp: 1615–1630, 2005.

[8] C. H. Evans, *Notes on the OpenSURF Library,* CSTR-09-001, University of Bristol. January 2009.

[9] J. Bauer, N. Sunderhauf, P. Protzel. *Comparing Several Implementations of Two Recently Published Feature Detectors,* In Proc. of the International Conference on Intelligent and Autonomous Systems, IAV, Toulouse, France, 2007.

[10] A. Witkin. *Scale-space filtering,* in proc. Of Artif. Intell, conference, pp: 1019-1021, 1983.

Proceedings of the International Multiconference on Computer Science and Information Technology pp. 25–32

ISBN 978-83-60810-27-9
ISSN 1896-7094

Evaluation of Clustering Algorithms for Polish Word Sense Disambiguation

Bartosz Broda, Wojciech Mazur
Institute of Informatics, Wrocław University of Technology, Poland
bartosz.broda@pwr.wroc.pl, wojciech.mzr@gmail.com

***Abstract*—Word Sense Disambiguation in text is still a difficult problem as the best supervised methods require laborious and costly manual preparation of training data. Thus, this work focuses on evaluation of a few selected clustering algorithms in task of Word Sense Disambiguation for Polish. We tested 6 clustering algorithms (K-Means, K-Medoids, hierarchical agglomerative clustering, hierarchical divisive clustering, Growing Hierarchical Self Organising Maps, graph-partitioning based clustering) and five weighting schemes. For agglomerative and divisive algorithm 13 criterion function were tested. The achieved results are interesting, because best clustering algorithms are close in terms of cluster purity to precision of supervised clustering algorithm on the same dataset, using the same features.**

I. Introduction

WORD Sense Disambiguation (WSD) deals with contextual resolution of lexical ambiguity. Most words in natural language have more than one lexical meaning (sense), but usually only one of them is active in a given context. Typical example of ambiguous word is *line*, which according to WordNet (an electronic thesaurus, cf. [1]) has 36 senses. WSD is important problem for applications in domain of Natural Language Processing (NLP). Machine translation cannot work without some form of disambiguation, but WSD can be helpful also for information retrieval, information extraction and computer aided lexicography among others [2].

WSD is a hard problem. Most difficulties arise from the fact that the concept of a meaning is vague. Usually, there are no clear boundaries between one sense or the other [3]. Typically, the problem of defining meaning is tackled with using dictionaries (which are called *sense inventory* in a context of WSD). I.e., from the algorithmic point of view sense inventories are used to enumerate all the meanings that a given word has. Now, the goal of WSD can be stated as choosing appropriate sense from sense inventory in a given context of a word.

There are two main approaches to WSD based on machine learning: supervised and unsupervised [2].[1] Supervised learning focuses on the usage of manually disambiguated examples of text snippets containing ambiguous words. We need to choose an appropriate sense inventory in advance, at early stages of the construction of supervised WSD system. Some features are extracted from those text snippets (or contexts[2]) and classifiers are trained using this manually labeled data. Most of the time, supervised approaches are superior to unsupervised in terms of accuracy of automatic disambiguation when used on the same type of texts that the systems were trained on.

Nevertheless, there is another issue connected with the problem of the definition of a meaning, i.e., an issue of creation of other resources used for automatic system performing WSD. This is especially evident in creation of corpora[3] manually annotated (tagged) with senses, which are used for training machine learning classifiers in a supervised setting. There are two important problems during manual sense tagging of a corpus: low *interannotator agreement* (IA) and high cost of annotation process. IA is a way of measuring how much annotations assigned by one annotator differers from annotations assigned by another annotator. IA is used for estimation of an upper bound on performance on automatic WSD. Typically, it is not enough to give a value of percentage agreement, because agreements and disagreements may arise by chance. Cohen's κ is widely used in computational linguistic community for this purpose, but there are also other measures [5]. The cost of annotation is high, because large effort is required during manual annotation. Mihalcea estimated that a construction of a corpus with sufficient amount of data for supervised classification algorithms for 20 000 ambiguous words would require 80 man-years of work [6].

On the other hand, unsupervised and semi-supervised algorithms can be used. The amount of manual labor required is much lower in learning without supervision. Unsupervised approaches to WSD tend to use unlabeled data and automatically find sense distinctions. Usually those methods involve some form of clustering. Harris' distributional hypothesis [7] can be used as a theoretical foundation for unsupervised methods of WSD. It states that "meaning of entities (...) is related to the restrictions on combinations of these entities relative to other entities.". In this context entities can be understood as words.

The main goal of this work is to compare various clustering algorithms in the task of unsupervised Word Sense Disambiguation for Polish data. In unsupervised WSD system deals with grouping of contexts for given word that express the

[1] There is a plethora of other approaches to WSD, e.g., based on translational equivalence or hand-written rules. We omit those for brevity. For extensive overview of other methods see, e.g., [2], [4].

[2] We will use term *context* to denote a passage of text containing ambiguous word.

[3] Here we define a corpus as a collection of texts prepared for linguistic processing

same meaning without providing explicit sense labels for each group (e.g., without using a dictionary) [8]. Also, this work is motivated by the fact that clustering is important for semi-supervised WSD algorithm called Lexicographer Controlled Semi-automatic word Sense Disambiguation [9], [10]. So far, the selection of the algorithm used in LexCSD was motivated by the performance of the given algorithm in other tasks and its analytical properties, because analysis of the performance of different clustering algorithms in similar settings (i.e., using similar dataset and features) for Polish WSD is difficult to find.

There are a few differences when dealing with WSD data in comparison to classical applications of clustering. To name just a few: the distributions of classes (senses) are skewed[4], data is represented in spaces of very large number of dimensions (thousands or even hundreds of thousands), for some classes only very specific, often overlapping among classes features are important and sometimes there is difficulty in distinguishing between two close classes.

The paper is organized as follows. First the selected clustering algorithms are briefly described. Evaluation section starts with the analysis of evaluation metrics used. Next, the corpus and experimental settings are described. Section III-D provides discussion of results. Section IV gives a summary of performed experiments and overviews direction of further works.

II. Selected algorithms for testing

For this work we have selected a few classical clustering algorithms, but we tried to choose algorithms representing a few different approaches to the problem of clustering. We started with *K-means* and *K-medoids* algorithms, which represent simple, hard and flat clustering methods. We choose *Growing Hierarchical Self-Organising Map* (GHSOM) as a representative of family of clustering using neural networks. GHSOM is also a hierarchical clustering algorithm. We experiment with standard hierarchical clustering algorithms with different criterion functions, both from agglomerative and divisive families of algorithms. Last but not least, we test also graph-based clustering algorithm. We have reimplemented K-means, K-medoids and GHSOM and use existing implementation of other algorithms [11].

We are focusing on clustering for WSD so we will use NLP-related terminology during description of algorithms. As a task of WSD is a contextual one, we will cluster *contexts* (text snippets) containing ambiguous word. From the context some real-valued features are extracted. So the context is a vector of features $\vec{v}$ in high dimensional space. We will use term *context* and *context vector* interchangeably. The exact nature of context and feature extraction process are described in Sec. III-B.

A. K-means and K-medoids

K-means is one of the simplest clustering algorithm. K-means defines cluster as a centers of mass of contexts being clustered [12]. Those centres are represented as centroids. Initially random contexts are chosen as centroids. Then we assign most similar contexts to each centroid. After this step new centroids are computed as a mean of all the contexts in a group. This process is then repeated until some stopping criterion is reached, e.g., number of iteration reaches some predefined threshold or the clustering solution do not change significantly between subsequent iterations.

K-medoids is similar in concept to K-means algorithm. The most fundamental difference between the two algorithms is that K-medoids uses real contexts from the dataset as a basis for clustering in contrast to centroids used in K-means (which are artificial contexts). One of the realisations of K-medoids is an approach called *Partition Around Medoids*, or PAM [13]. In PAM one starts with randomly selection of initial medoids. Then every swapping of every medoid with every context is tested in terms of decreasing *cost* of whole clustering solution. This approach has its drawbacks in terms of computational complexity, i.e., $O(k(n-k)^2)$, where n is number of contexts to cluster and k is number of medoids. Thus a few extensions have been proposed that, e.g., employ sampling (CLARA) or randomized search (CLARANS) [13]. Nevertheless, we use classical PAM, as both mentioned algorithms can have negative impact on quality in comparison to PAM. This approach is applicable in our experiments, as we use relatively small datasets.

B. Growing Hierarchical Self-Organizing Map

The Growing Hierarchical Self-Organizing Map (GHSOM) [14] is a natural extension of Kohonen's idea of Self-Organizing Maps (SOM) [15]. SOM is an *artificial neural network* consisting of many neurons. Every neuron consists of a weight vector. Training SOM is done in an unsupervised manner applying *winner takes most* strategy. Every feature vector is delivered to the network input several times. For every input vector the similarity with the neuron weight vector is computed. Weights of the most similar neuron (the winner) and its neighbourhood are updated to be even more similar to the input pattern. The learning algorithm is constructed in such a way, that the neighbourhood and the degree of the weight updating is decreasing over time.

GHSOM address one of the most important drawback of SOM — the a priori definition of the map structure. Rauber *et al.* proposed an algorithm for growing SOM both in a terms of the number of map neurons and the hierarchy [14]. After the training stage of SOM mean quantization error for every neuron i (mqe_i) is calculated as the average distance of every context recognised by the neuron i to its weight vector. The average MQE_j for whole map on level j is computed, too. If $MQE_j \geq \tau_1 \cdot MQE_{j-1}$ then the additional row or column of neurons is added to the map and the training stage is repeated. In the other case the mqe_i for every neuron is compared to MQE_j. If $meq_i \geq \tau_2 \cdot MQE_{j-1}$ then another layer of the map is created for contexts recognised by the neuron i.

[4] Not all senses are represented in the data equally; distribution of senses is biased towards a few frequent senses.

C. Agglomerative and Divisive Clustering

Agglomerative and divisive clustering algorithms produce *hierarchical* clustering trees called dendrograms. Agglomerative clustering starts in a situation that each context is contained in a separate cluster, then in each step two clusters maximising *criterion function* are merged. On the other hand, divisive algorithms starts with all contexts in one cluster which are repeatedly bisected according to the criterion function. We are using existing implementation of hierarchical algorithms from CLUTO[5] [11]. We use *rbr* variant of divisive algorithm, i.e., standard bisecting clustering is employed and is further optimized according to criterion function [16].

Criterion function is very important aspect of both agglomerative and divisive clustering algorithms as it drives the whole process. There are many criterion function available [17]. We have tested standard criterion functions used with agglomerative algorithms, i.e.: single link (slink), complete link (clink), average link (upgma) and weighted variants of single (wslink), complete (wclink) and average links (wupgma).

The second group of criterion function including $i_1, i_2, \varepsilon_1, G_1, G_1', H_1, H_2$ can be used with both agglomerative and divisive algorithms. The exact form of those functions are given by [11]:

$$I_1 = \text{maximize} \sum_{i=1}^{k} \frac{1}{n_i} \Big(\sum_{\vec{v},\vec{u} \in S_i} sim(\vec{v},\vec{u}) \Big) \quad (1)$$

$$I_2 = \text{maximize} \sum_{i=1}^{k} \sqrt{\sum_{\vec{v},\vec{u} \in S_i} sim(\vec{v},\vec{u})} \quad (2)$$

$$\varepsilon_1 = \text{minimize} \sum_{i=1}^{k} n_i \frac{\sum_{v \in S_i, u \in S} sim(\vec{v},\vec{u})}{\sqrt{\sum_{v,u \in S_i} sim(\vec{v},\vec{u})}} \quad (3)$$

$$G_1 = \text{minimize} \sum_{i=1}^{k} \frac{\sum_{v \in S_i, u \in S} sim(\vec{v},\vec{u})}{\sqrt{\sum_{v,u \in S_i} sim(\vec{v},\vec{u})}} \quad (4)$$

$$G_1' = \text{minimize} \sum_{i=1}^{k} n_i^2 \frac{\sum_{v \in S_i, u \in S} sim(\vec{v},\vec{u})}{\sqrt{\sum_{v,u \in S_i} sim(\vec{v},\vec{u})}} \quad (5)$$

$$H_1 = \text{maximize} \frac{I_1}{\varepsilon_1} \quad (6)$$

$$H_2 = \text{maximize} \frac{I_2}{\varepsilon_1}, \quad (7)$$

where k is total number of clusters, S is total number of contexts to cluster, S_i is a set of contexts assigned to i-th cluster, $n_i = |S_i|$, and $sim(\vec{v},\vec{u})$ is similarity between two context vectors $\vec{v}$ and $\vec{u}$.

D. Graph Partitioning Based Clustering

We use an implementation of min cut graph partitioning algorithm from CLUTO [11]. This algorithm starts with creation of neighbourhood graph based on similarities between contexts and then applies min cut to partition the graph into disjoint regions. Min cut uses approach that the size of graph edges in a partition is minimal.

This approach achieved high quality in research on semi-automatic extension of Polish WordNet [18] and was also used in Polish WSD based on weakly-supervised settings using LexCSD algorithm [10].

III. Experiments

A. Evaluation Measures

Evaluation of clustering algorithms can be done in many ways [19]. Some of them are based on *external criteria*, i.e., the comparison of the resulting clustering solution with some pre-existing categories that were created manually. On the other hand, one can use an *internal criteria* without resorting to gold standard clustering. The most important drawback of evaluation using internal criteria is that good score does not always corresponds to good results of clustering in a given application [20]. As we have developed semantically annotated corpus (SCWSD, see Sec. III-B) we can use it for the need of evaluation. The problem with SCWSD is its small size, so there is a risk of not capturing all of the peculiarities and biases of some large corpora in SCWSD.[6]

We used several measures for evaluation to capture different aspects of created groups. For measuring how homogeneous clusters are we used *Purity*:

$$Purity(\Omega, C) = \frac{1}{N} \sum_{k} \max_{j} |\omega_k \cap c_j|, \quad (8)$$

where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_k\}$ is a set of clusters, a $C = \{c_1, c_2, \ldots, c_j\}$ — a set of pre-existing categories. In our setting C is a set of contexts with ambiguous word annotated with the same sense. $Purity(\Omega, C) \in \langle 0, 1 \rangle$, where 1 is the best case. A drawback of $Purity$ is its preference for solutions with large number of groups. Assigning every context to a singelton cluster gives Purity of 1 [20].

The *Rand Index* measures accuracy on the basis of decisions performed for the subsequent context pairs. If we use TP for *true positive*, TN for *true negative*, FN for *false negative* and FP for *false positive.* the Rand Index is given by the following equation:

$$R_I = \frac{TP + TN}{TP + FP + FN + TN} \quad (9)$$

One of the drawbacks of using R_I for evaluation is the equal treatment of false positives and negatives. Using decision for context pairs we can also use standard measures of information retrieval, i.e., precision P, recall R and the harmonic mean of precision and recall F_β:

[5] CLUTO is a free software package implementing several clustering algorithms including partitioning, agglomerative and graph-based. Available at: http://glaros.dtc.umn.edu/gkhome/views/cluto/

[6] On the other hand, the total size of the dataset, i.e., 1344 contexts (Tab. I), is not very small in comparison to, e.g., [16], where the smallest dataset has 878 elements and the largest — 4069 elements.

$$P = \frac{TP}{TP + FP} \quad (10)$$

$$R = \frac{TP}{TP + FN} \quad (11)$$

$$F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \quad (12)$$

Another way of measuring clustering quality is to use *Normalized Mutual Information* (NMI). NMI takes into account the trade off between quality and number of clusters as opposed to Purity.

$$NMI(\Omega, C) = \frac{MI(\Omega; C)}{[H(\Omega) + H(C)]/2} \quad (13)$$

$$MI(\Omega, C) = \sum_{k,j} P(\omega_k \cap c_j) log \frac{P(\omega_k \cap c_j)}{P(\omega_k)P(c_j)} \quad (14)$$

$$H(\Omega) = -\sum_k P(\omega_k) log P(\omega_k), \quad (15)$$

where mutual information $MI(\Omega, C)$ is normalized by entropy $H(\Omega)$ and probabilities can be counted using maximum likelihood estimation (MLE). The normalization by entropy is performed for penalizing clustering solutions with large number of clusters, as entropy tends to increase to maximum with number of clusters. Thanks to the normalization $NMI \in < 0, 1 >$, where 0 corresponds to random clustering.

Last method used for evaluation is a variation of F-Measure used by Kulkarni and Pedersen in SenseCluster system [21]. Its main idea is contained in a sentence: "One sense—one cluster". It means that each cluster must have unique sense label. The label of each cluster is determined by most frequent class of cluster members, but one sense label cannot be assigned to multiple clusters. In particular, having more clusters than senses force us to treat members of unlabeled groups as unclustered. With that assumption in mind the standard measures of precision P_p and recall R_p are defined as:

$$P_p = Purity(\Omega', C) \quad (16)$$

Where Ω' is set of labeled clusters, C is set of classes defined above.

$$R_p = \frac{\#hits}{\#total\ instances} \quad (17)$$

Where *#hits* is number of elements with sense accordant to cluster sense label, *#total instances* - number of all elements including outliers.

Having P_p and R_p, we use F_β, with $\beta = 1$:

$$F(P)_1 = \frac{2 P_p R_p}{P_p + R_p} \quad (18)$$

B. Corpus Description

For the need of evaluation we have used recently developed, manually disambiguated corpus called *Korpusik US*[7] (rough English translation: Small Corpus for WSD, henceforth SCWSD) [10].

The corpus consists of parts of IPI PAN Corpus [22]. Only 13 ambiguous words were annotated. The chosen words represent the variety of different problems for WSD; some of the senses have homonymous character, i.e., they represent separate homonyms of the same morphological base form. The sense inventory was based on extended version of Polish WordNet [18]. Performing evaluation only on limited set of words is called *Lexical Sample Task* [2]. The following words were chosen for annotation:

- agent: a person who represents a company or artist, secret agent, chemical agent
- automat: automaton, machine, telephone, a coin-operated automatic machine, submachine gun, automatic car transmission ;
- dziób: beak, bow, nose, front part of a ship, informal mouth, face (semantically marked)
- język: tongue, natural language
- klasa: category, class, rank, classroom, mathematical category, savoir-vivre, social class, subject, excellence
- linia: line, route, edge, line separating two areas, power line, assembly line, telephone line, row, lineage, contour, figure, ruler, line of defence, line of products for sale, credit line, geometric line.
- pole: field, area, playing field, physical field.
- policja: police (organization), police station, 'policemen'.
- powód: reason', plaintiff.
- sztuka: art, act of craftsmanship, item, a beautiful girl, dramatic play, theatrical performance of a play, an amount of fabric (for example wool), bale, a piece of meat.
- zamek: castle, lock, zipper, breechblock, trap in hockey, a part of machine or any device that stops its action.
- zbiór: set, group, mathematical set, collection, harvest, an act of harvesting, an exercise book, file.
- zespół: team, band, group of machines, complex of buildings, syndrome, sport team, botanical 'association'.

The annotation of the corpus was done by two native speakers of Polish: a professional linguist and a computational linguist. The exact corpus statistics are show in Tab. I. There are 1344 annotated examples. After the annotation process we measured interannotator agreement using Cohen's κ [23]. The agreement is surprisingly high, 0.88 for whole corpus. Such an agreement is very high in comparison with other corpora annotated with fine-grained WordNet-based senses [2].

SCWSD was previously used in research on WSD for Polish [10]. Its previous version (sense inventory base on early version Polish WordNet) was also used in [9], [24]. The best reported precision in [10] using supervised classifiers is 72.42%. There is also a baseline associated with a corpus called Most Frequent Sense baseline (MFS), i.e., using a heuristic classifier that chooses always the most frequent sense. For SCWSD the MFS baseline is 44.56% (weighted average for all the words).

[7] The corpus is available for browsing http://nlp.pwr.wroc.pl/webann

TABLE I
SMALL CORPUS FOR WSD (SCWSD) — STATISTICS

Word	No. of senses	Annotated senses	Examples	κ
agent	5	1/9/3/47/10	70	0.80
automat	5	1/24/30/4/46	105	0.97
dziób	4	28/13/31/9	81	0.98
język	3	3/23/49	75	0.97
klasa	11	15/6/12/11/14/31/10/8/1/10/1	119	0.80
linia	13	13/3/2/2/4/2/11/13/4/3/1/2/21	81	0.72
pole	5	1/1/23/25/46	96	0.86
policja	3	17/25/22	64	0.73
powód	2	136/122	258	0.98
sztuka	6	12/10/2/11/41/19	95	0.84
zamek	4	18/19/36/19	92	1.00
zbiór	5	32/7/8/31/9	87	0.87
zespół	6	10/4/28/58/1/20	121	0.95

C. Experimental Settings

The experimental settings are the same as in experiments presented in [10], where approaches based on supervised and weakly supervised were tested. We use only small manually annotated corpus because we want to have a clear point of comparison with previous work. We use the same features as in [10], i.e., bag of words, to simplify discussion. Contrary to [10], there is no need to split the data into training and test sets for evaluation because of unsupervised nature of clustering algorithms, cf [16], [17], [20].

This features are extracted from text in the following process. First a text window surrounding ambiguous word of ±20 segments (tokens) is constructed.[8] Then the occurrence of a word[9] is noted in a feature vector. Every dimension corresponds to different word. The resulting vectors are sparse. Instead of using raw frequencies we tested a few weighting schemes coupled with *cosine* function for measuring similarities between contexts. The following measures were tested:

- Term frequency, inversed document frequency (henceforth, *tf.idf*), see [20]. We assume, that document is the same as context in this measure.
- *Logent* — values were scaled with logarithm and divided by entropy of a context (standard Shanon entropy counted as $\sum p \log p$, where p is estimated using MLE). This technique was proposed in *Latent Semantic Analysis* by Landauer and Dumais [26].
- Mutual information (definition following work of Lin [27]). We use *lin_cos* for denoting this measure.
- Discounted pointwise mutual information (*pmi*), see [28]. The most important difference between pmi and lin_cos is that pmi uses discounting factor to address the problem of overestimation of mutual information in case of infrequent events.
- Rank Weight Function (RWF) based on mutual information defined by Lin (*rwf_lin*), see [18].
- Rank Weight Function (RWF) based on pointwise mutual information (*rwf_pmi*), see [18].

Both RWF function works by using ranks instead of exact feature values. It allows for certain level of generalization from word occurrence frequencies, which can be accidental. RWF approach was previously used in a task of finding similar words in large corpus with very good results [18].

D. Results

Tab. II-VII present results averaged for all the words using different weighting schemes. The first thing to notice is that the results are very hard to grasp. We have noticed several regularities. Firstly, the results of GHSOM evaluation are high in terms of Purity, NMI and Rand Index and very low in terms of F-Measures. Secondly, the amount of data to analyse is very high: 5 weighting measures $\times$ 5 evaluation measures $\times$ 25 algorithm variants. And last but not least, there is no "best" algorithm, i.e., an algorithm that would rank highest in all the different evaluation measures.

To tackle the first problem we will analyse GHSOM in isolation from the other algorithms. We can observe, that the Purity measure for GHSOM has always got the highest values (over 70%). In case of growing hierarchical SOM, we cannot simply define desirable number of clusters, which is strictly determined by number of neurons in each layer. In our — relatively small — set of input data, reading results from all network layers leads to a number of singleton clusters. As was written above, Purity of such clusters equals 1. This fact artificially increases results and clearly shows the weakest side of the Purity measure. Evaluation using F-Measures shows totally opposite results, discarding GHSOM as a efficient clustering method.

There are also some *not available* results (*n/a*) for logent weighting in GHSOM rows. In this particular case, we observed uncontrolled growing of first layer of neural network. Tunning the parameters was also not helpful. The size of the layer reaching 20x20 (which represents 400 clusters), while having data set with only several groups was obviously too big, so we decided to discard such results. The growth of the network might be caused by the nature of logent weighting on this particular dataset. By taking logarithms of feature values the vectors become extremely sparse, because of large number of ones in the features.

One of the most important properties of GHSOM, i.e., not having to specify number of clusters in advance, becomes its most important drawback using standard evaluation measures. To overcome problems with GHSOM two steps can be taken: analysing the resulting network for searching the desired number of clusters or using evaluation measures that can capture other positive properties of GHSOM, most importantly the spatial relations between neurons.

For resolving the second problem with large amount of data to analyse we tested whether some of the evaluation measures are correlated to each other. We used Spearman rank correlation coefficients[10]. The results of this evaluation were

[8] A segment (token) is defined as word, words separators, but some words can be split onto several segments. For discussion see [25].

[9] A word is a fuzzy concept, more specifically we use a base forms of a word coupled with its flexemic class.

[10] We tested also Pearson's correlation coefficient. The results were also suggesting high correlation among measures, but rank correlation coefficient is more robust to outliers, in our case — GHSOM.

interesting: in most cases correlation was very high between Purity, NMI and RI. Average Spearman's $\rho = 0.83$ and the two-tailed p-value were always lower then 0.0001. Correlation between F_1 and $F(P)_1$ is lower, but also high ($\rho = 0.64$), where usually F_1 is a little bit lower. As $F(P)_1$ allows only for assigning given sense label to cluster only once, we will focus only on decision-based F_1, which scores every pair of clustered contexts.

Reducing the measures to only two (i.e., Purity and F_1) does not solve the problem of choosing the best combination of clustering and weighting scheme. Thus we created two rankings of all the pairs of <clustering algorithm, weighting scheme> ordered by Purity (R_1) and F_1 (R_2). To rank different weighting schemes we use a sum of average of both ranks. This approach can be interpreted as choosing a weighting scheme for which on average the clustering solutions are better then the other. The best weighting scheme is pmi, followed by lin_cos, tf.idf, rwf_lin, rwf_pmi and logent. The difference between pmi using discounting factor and without discounting factor (lin_cos) is minor. The same applies to RWF versions of mutual information based measures. The best results were achieved by agglomerative clustering with weighted average link criterion function while using pmi weighting (Agglo(wslink) in Tab. V). Following are measures using both agglomerative and divisive clustering using (weighted and unweighted) average link criterion for Agglo and e1, h2 and g1 for Rbr. The worst performing algorithms are K-means and K-medoids, which is not surprising—we are using them as another way of setting a baseline.

Those results are interesting, because in related tasks graph partitioning was performing better then the other algorithms. Also, best result achieved for pmi-based weighting scheme is interesting, but not as surprising as these family of measures achieves very good results in the task of finding similar words [18].

The results can be compared to precision of supervised algorithms, because precision corresponds to Purity in our evaluation. All the algorithms beat the baseline of selection always most frequent sense. The best supervised classification algorithm tested on the same dataset, using the same feature set achieved precision of 72.42% as reported by [10], the best clustering algorithm have Purity=71.36%. These results is very high, as usually unsupervised approaches have troubles with beating MFS baseline [2]. This can be explained with high quality of corpus annotations, small corpus size and partially balanced sense distribution in corpus.

TABLE II
AVERAGE RESULTS FOR TF.IDF WEIGHT

	Purity	**NMI**	**RI**	**F_1**	**$F(P)_1$**
Agglo(clink)	49,80	0,137	0,515	41,03	42,43
Agglo(e1)	63,85	0,281	0,679	41,50	50,46
Agglo(g1)	63,03	0,274	0,672	41,66	50,09
Agglo(g1p)	56,90	0,235	0,550	54,49	54,07
Agglo(h1)	57,66	0,243	0,638	41,49	44,57
Agglo(h2)	61,83	0,261	0,662	40,02	48,90
Agglo(i1)	53,36	0,211	0,512	45,37	45,77
Agglo(i2)	63,77	0,292	0,678	42,55	50,69
Agglo(slink)	48,17	0,090	0,385	48,68	45,64
Agglo(upgma)	61,50	0,301	0,637	**58,32**	**58,07**
Agglo(wclink)	49,80	0,137	0,515	41,03	42,43
Agglo(wslink)	48,17	0,090	0,385	48,68	45,64
Agglo(wupgma)	64,19	0,314	0,674	50,13	57,19
Graph	64,65	0,303	0,676	44,51	51,85
K-means	55,28	0,169	0,630	36,33	44,27
K-medoids	52,98	0,141	0,611	39,45	44,05
Rbr(e1)	68,38	0,342	0,707	44,97	53,73
Rbr(g1)	69,88	**0,378**	**0,727**	47,40	55,15
Rbr(g1p)	64,93	0,326	0,662	52,58	57,49
Rbr(h1)	66,00	0,332	0,685	43,88	51,80
Rbr(h2)	68,52	0,348	0,709	45,68	54,10
Rbr(i1)	58,72	0,275	0,561	47,18	50,76
Rbr(i2)	67,10	0,328	0,691	44,85	52,46
GHSOM	**74,55**	0,336	0,668	13,32	29,26
GHSOM – first	62,20	0,276	0,660	42,29	49,18

TABLE III
AVERAGE RESULTS FOR LOGENT WEIGHT

	Purity	**NMI**	**RI**	**F_1**	**$F(P)_1$**
Agglo(clink)	52,36	0,143	0,781	41,19	36,65
Agglo(e1)	51,99	0,142	0,768	44,59	37,98
Agglo(g1)	53,08	0,157	0,776	45,18	40,33
Agglo(g1p)	58,65	0,210	0,662	37,78	45,47
Agglo(h1)	53,38	0,168	0,800	34,96	33,53
Agglo(h2)	53,46	0,163	0,807	32,14	33,95
Agglo(i1)	52,42	0,154	0,777	44,59	39,15
Agglo(i2)	51,65	0,147	0,769	44,59	38,21
Agglo(slink)	52,23	0,145	0,778	43,82	37,80
Agglo(upgma)	51,84	0,143	0,773	43,05	37,69
Agglo(wclink)	51,14	0,126	0,778	40,99	35,53
Agglo(wslink)	52,76	0,151	0,779	43,86	38,11
Agglo(wupgma)	51,16	0,132	0,767	44,32	37,21
Graph	**65,40**	**0,283**	**0,907**	37,81	26,80
K-means	48,02	0,084	0,442	46,18	44,22
K-medoids	48,38	0,099	0,425	**47,65**	44,74
Rbr(e1)	53,70	0,170	0,792	34,53	34,79
Rbr(g1)	55,06	0,190	0,787	44,37	38,92
Rbr(g1p)	60,07	0,222	0,665	37,78	**46,00**
Rbr(h1)	52,40	0,164	0,788	34,43	33,78
Rbr(h2)	52,96	0,170	0,792	34,60	34,83
Rbr(i1)	51,64	0,156	0,782	35,94	34,01
Rbr(i2)	52,35	0,166	0,789	34,36	33,89
GHSOM	n/a	n/a	n/a	n/a	n/a
GHSOM – first	n/a	n/a	n/a	n/a	n/a

IV. CONCLUSIONS AND FURTHER WORKS

This paper presented evaluation of selected clustering algorithms in task of Word Sense Disambiguation for Polish. We have used simple lexical features to represent contexts of ambiguous words. The features were weighted using different weighting schemes. Using pointwise mutual information as a weighting scheme gave best results on average.

Evaluation of clustering was performed on manually disambiguated corpus using five standard evaluation measures [20], [21]. Interestingly, in our settings some of the measures are highly correlated, thus it is only necessary to use two of them, e.g., Purity and F-Measure. This might be caused by the fact, that we know how many clusters are in the data in advance, but one of the problem that the more elaborated measures are trying to address is the problem with generation of large number of clusters.

TABLE IV
AVERAGE RESULTS FOR LIN_COS WEIGHT

	Purity	**NMI**	**RI**	**F$_1$**	**F(P)$_1$**
Agglo(clink)	51,29	0,141	0,561	40,58	44,59
Agglo(e1)	63,62	0,300	0,688	42,64	51,87
Agglo(g1)	56,69	0,232	0,563	54,49	54,31
Agglo(g1p)	60,07	0,217	0,670	37,94	47,43
Agglo(h1)	62,95	0,294	0,672	41,99	49,27
Agglo(h2)	62,50	0,261	0,667	40,41	49,41
Agglo(i1)	53,07	0,170	0,476	46,62	46,15
Agglo(i2)	64,37	0,295	0,686	43,03	50,46
Agglo(slink)	48,24	0,099	0,395	49,23	45,94
Agglo(upgma)	61,28	0,291	0,637	**56,79**	56,22
Agglo(wclink)	51,29	0,141	0,561	40,58	44,59
Agglo(wslink)	48,24	0,099	0,395	49,23	45,94
Agglo(wupgma)	67,19	0,344	0,682	51,97	58,48
Graph	65,17	0,306	0,682	45,64	52,83
K-means	54,54	0,162	0,629	35,68	43,30
K-medoids	52,24	0,155	0,608	39,23	44,80
Rbr(e1)	69,57	0,367	**0,722**	46,81	55,44
Rbr(g1)	66,36	0,339	0,657	53,54	**58,92**
Rbr(g1p)	64,44	0,291	0,709	41,91	51,63
Rbr(h1)	67,71	0,337	0,694	44,91	52,54
Rbr(h2)	68,90	**0,368**	**0,722**	46,61	54,77
Rbr(i1)	59,91	0,287	0,576	47,53	52,26
Rbr(i2)	68,15	0,332	0,695	45,07	52,83
GHSOM	**73,51**	0,323	0,666	12,70	28,09
GHSOM – first	56,47	0,210	0,641	39,38	45,77

TABLE V
AVERAGE RESULTS FOR PMI WEIGHT

	Purity	**NMI**	**RI**	**F$_1$**	**F(P)$_1$**
Agglo(clink)	50,47	0,133	0,572	39,24	41,68
Agglo(e1)	61,53	0,268	0,661	40,53	48,29
Agglo(g1)	60,03	0,253	0,658	39,85	47,91
Agglo(g1p)	62,58	0,258	0,663	39,83	48,07
Agglo(h1)	65,33	0,303	0,686	43,94	53,30
Agglo(h2)	63,56	0,300	0,686	42,22	50,99
Agglo(i1)	56,85	0,240	0,560	49,08	49,71
Agglo(i2)	64,14	0,280	0,676	42,50	50,55
Agglo(slink)	48,32	0,096	0,391	48,81	45,79
Agglo(upgma)	61,66	0,311	0,641	**56,75**	58,39
Agglo(wclink)	50,47	0,133	0,572	39,24	41,68
Agglo(wslink)	48,32	0,096	0,391	48,81	45,79
Agglo(wupgma)	68,38	0,368	0,710	52,78	**59,46**
Graph	62,27	0,268	0,662	42,50	49,85
K-means	56,32	0,181	0,635	36,83	45,38
K-medoids	53,42	0,155	0,614	40,10	44,87
Rbr(e1)	70,92	**0,399**	**0,736**	48,64	55,60
Rbr(g1)	69,05	0,358	0,721	46,29	53,37
Rbr(g1p)	68,98	0,370	0,723	46,58	54,26
Rbr(h1)	67,19	0,316	0,683	44,03	51,05
Rbr(h2)	71,36	0,393	0,732	48,28	55,82
Rbr(i1)	64,45	0,338	0,646	50,49	56,04
Rbr(i2)	68,07	0,331	0,696	45,06	52,17
GHSOM	**74,71**	0,329	0,667	12,98	28,93
GHSOM – first	59,82	0,241	0,651	41,24	49,02

TABLE VI
AVERAGE RESULTS FOR RWF_LIN WEIGHT

	Purity	**NMI**	**RI**	**F$_1$**	**F(P)$_1$**
Agglo(clink)	49,29	0,121	0,578	38,16	42,45
Agglo(e1)	60,57	0,240	0,680	39,35	48,69
Agglo(g1)	52,10	0,133	0,775	43,84	38,19
Agglo(g1p)	55,39	0,204	0,625	38,15	44,40
Agglo(h1)	61,15	0,250	0,680	40,36	48,78
Agglo(h2)	60,07	0,214	0,669	37,89	47,65
Agglo(i1)	57,13	0,215	0,621	42,92	45,90
Agglo(i2)	61,44	0,246	0,676	39,23	47,95
Agglo(slink)	49,35	0,123	0,473	**46,79**	45,07
Agglo(upgma)	58,80	0,221	0,655	42,11	48,17
Agglo(wclink)	49,51	0,121	0,572	38,85	42,52
Agglo(wslink)	49,35	0,123	0,473	**46,79**	45,07
Agglo(wupgma)	60,36	0,226	0,670	38,89	48,34
Graph	61,48	0,248	0,677	42,00	50,24
K-means	57,21	0,185	0,637	37,03	45,01
K-medoids	53,42	0,176	0,563	43,53	43,53
Rbr(e1)	64,12	0,271	0,694	41,13	50,34
Rbr(g1)	54,06	0,182	**0,795**	35,39	34,75
Rbr(g1p)	63,43	0,289	0,687	43,83	52,51
Rbr(h1)	62,55	0,266	0,687	42,15	50,72
Rbr(h2)	63,58	0,280	0,700	41,45	49,96
Rbr(i1)	64,26	**0,301**	0,700	44,62	**53,18**
Rbr(i2)	63,28	0,285	0,701	42,11	50,71
GHSOM	**71,66**	0,298	0,663	11,09	26,65
GHSOM – first	54,17	0,149	0,615	35,02	42,88

TABLE VII
AVERAGE RESULTS FOR RWF_PMI WEIGHT

	Purity	**NMI**	**RI**	**F$_1$**	**F(P)$_1$**
Agglo(clink)	51,30	0,137	0,586	39,14	42,80
Agglo(e1)	58,11	0,201	0,658	37,74	46,67
Agglo(g1)	57,36	0,235	0,560	**51,54**	52,08
Agglo(g1p)	58,91	0,219	0,649	41,56	48,45
Agglo(h1)	59,14	0,218	0,661	38,12	46,36
Agglo(h2)	57,29	0,203	0,652	37,14	43,46
Agglo(i1)	58,05	0,236	0,648	42,64	45,25
Agglo(i2)	60,72	0,238	0,670	40,77	47,93
Agglo(slink)	47,90	0,085	0,416	47,94	44,38
Agglo(upgma)	55,81	0,205	0,627	45,52	48,28
Agglo(wclink)	51,14	0,143	0,594	38,44	42,88
Agglo(wslink)	47,82	0,089	0,415	47,91	44,16
Agglo(wupgma)	60,12	0,221	0,664	38,58	46,95
Graph	62,11	0,251	0,674	40,66	49,22
K-means	58,27	0,207	0,649	38,88	46,44
K-medoids	52,60	0,160	0,565	44,26	44,88
Rbr(e1)	60,23	0,219	0,666	37,85	46,53
Rbr(g1)	66,13	**0,333**	0,672	51,53	**56,24**
Rbr(g1p)	60,85	0,227	0,657	39,06	46,08
Rbr(h1)	60,30	0,224	0,666	38,25	45,55
Rbr(h2)	60,30	0,226	0,668	38,46	46,59
Rbr(i1)	61,60	0,241	**0,675**	41,56	49,14
Rbr(i2)	60,32	0,220	0,665	38,30	46,23
GHSOM	**72,24**	0,308	0,664	10,93	27,85
GHSOM – first	54,61	0,167	0,622	36,52	43,59

The results of evaluation are interesting. Divisive clustering algorithm are not worst then agglomerative. Previously used graph partitioning approach is sub-optimal. Not surprisingly, K-means and K-medoids produce clustering of lowest quality.

In the future we are planing to extend this work in a few ways. We want to evaluate some clustering algorithms that were tailored to a problem of Word Sense Disambiguation (e.g., Clustering by Committee [29]). Testing using more elaborated features can give more insight into nature of different clustering schemes. Evaluation of the algorithms on large corpus is needed (f.e., using whole IPI PAN Corpus [22]), because used corpus is small and potentially does not capture peculiarities of large amount of unrestricted texts. Of course, such evaluation has to be performed manually, based on examination of only statistical significant samples of results.

Application-based evaluation of clustering in performing WSD in weakly-supervised settings is also needed. We plan to use LexCSD algorithm for this purpose [9], [10]. This is especially important, as we have used graph partitioning clustering, which occurred to be a sub-optimal according to evaluation presented in this work.

Acknowledgment

Work financed by Innovative Economy Programme project POIG.01.01.02-14-013/09.

References

[1] C. Fellbaum *et al.*, *WordNet: An electronic lexical database*. MIT press Cambridge, MA, 1998.

[2] E. Agirre and P. Edmonds, Eds., *Word Sense Disambiguation: Algorithms and Applications*. Springer, 2006.

[3] A. Kilgarriff, "Word senses," in *Word Sense Disambiguation: Algorithms and Applications*, E. Agirre and P. Edmonds, Eds. Springer, 2006.

[4] R. Navigli, "Word sense disambiguation: A survey," *ACM Comput. Surv.*, vol. 41, no. 2, pp. 1–69, 2009.

[5] R. Artstein and M. Poesio, "Inter-coder agreement for computational linguistics," *Computational Linguistics*, vol. 34, no. 4, pp. 555–596, 2008.

[6] R. Mihalcea, "The Role of Non-Ambiguous Words in Natural Language Disambiguation," in *Proceedings of the Fourth RANLP*, 2003.

[7] Z. S. Harris, *Mathematical Structures of Language*. New York: Interscience Publishers, 1968.

[8] T. Pedersen, "Computational approaches to measuring the similarity of short contexts: A review of applications and methods," *South Asian Lang. Review*, to appear.

[9] B. Broda and M. Piasecki, "Semi-supervised word sense disambiguation based on weakly controlled sense induction," in *4rd Int. Symp. Adv. in AI and Applications*, 2009.

[10] M. M. B. Broda and M. Piasecki, "Evaluating lexcsd — a weakly-supervised method on improved semantically annotated corpus in a large scale experiment," in *Proceedings of Intelligent Information Systems*, S. T. W. M. A. Kłopotek, A. Przepiórkowski and K. Trojanowski, Eds., 2010.

[11] G. Karypis, "CLUTO a clustering toolkit," Univ. of Minnesota, Tech. Report, 2002.

[12] C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. The MIT Press, 2001.

[13] R. Ng and J. Han, "Efficient and effective clustering methods for spatial data mining," in *Proceedings of the International Conference on Very Large Data Bases*. Citeseer, 1994, pp. 144–144.

[14] A. Rauber, D. Merkl, and M. Dittenbach, "The growing hierarchical self-organizing maps: exploratory analysis of high-dimensional data," 2002.

[15] T. Kohonen, S. Kaski, K. Lagus, J. Salojrvi, J. Honkela, V. Paatero, and A. Saarela, "Self organization of a massive document collection," *IEEE Transactions on Neural Networks,*, vol. 11, pp. 574–585, 2000.

[16] Y. Zhao, G. Karypis, and U. Fayyad, "Hierarchical clustering algorithms for document datasets," *Data Mining and Knowledge Discovery*, vol. 10, no. 2, pp. 141–168, 2005.

[17] Y. Zhao and G. Karypis, "Empirical and theoretical comparisons of selected criterion functions for document clustering," *Machine Learning*, vol. 55, no. 3, pp. 311–331, 2004.

[18] M. Piasecki, S. Szpakowicz, and B. Broda, *A wordnet from the ground up*. Oficyna wydawnicza Politechniki Wroclawskiej, 2009.

[19] R. Forster, "Document clustering in large german corpora using natural language processing," Ph.D. dissertation, University of Zurich, 2006.

[20] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[21] A. Kulkarni and T. Pedersen, "Senseclusters: unsupervised clustering and labeling of similar contexts," in *ACL '05: Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, Morristown, NJ, USA, 2005, pp. 105–108.

[22] A. Przepiórkowski, *The IPI PAN Corpus: Preliminary version*. Institute of Computer Science PAS, 2004.

[23] R. Artstein and M. Poesio, "Inter-coder agreement for computational linguistics," *Computational Linguistics*, vol. 34, no. 4, pp. 555–596, 2008.

[24] R. Młodzki and A. Przepiórkowski, "The wsd development environment," in *Proc. 4rd Language and Technology Conference, Poznań, Poland*, Z. Vetulani, Ed., 2009.

[25] A. Przepiórkowski, "The potential of the IPI PAN Corpus," *Poznań Studies in Contemporary Linguistics*, vol. 41, pp. 31–48, 2006.

[26] T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition," *Psychological Review*, vol. 104, no. 2, 1997.

[27] D. Lin, "Automatic retrieval and clustering of similar words," in *Proceedings of the Joint Conference of the International Committee on Computational Linguistics*. ACL, 1998, pp. 768–774.

[28] P. Pantel and D. Lin, "Discovering word senses from text," in *Proc. ACM Conference on Knowledge Discovery and Data Mining (KDD-02)*, Edmonton, Canada, 2002, pp. 613–619.

[29] P. Pantel, "Clustering by committee," Ph.D. dissertation, Edmonton, Alta., Canada, Canada, 2003, adviser-Dekang Lin.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 33–36

ISBN 978-83-60810-27-9
ISSN 1896-7094

Generation of First-Order Expressions from a Broad Coverage HPSG Grammar

Ravi Coote and Andreas Wotzlaw
Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE
Neuenahrer Str. 20, 53343 Wachtberg, Germany
{ravi.coote, andreas.wotzlaw}@fkie.fraunhofer.de

Abstract—**This paper describes an application for computing first-order semantic representations of English texts. It is based on a combination of hybrid shallow-deep components arranged within the middleware framework Heart of Gold. The shallow-deep semantic analysis employs Robust Minimal Recursion Semantics (RMRS) as a common semantic underspecification formalism for natural language processing components. In order to compute efficiently first-order representations of the input text, the intermediate RMRS results of the shallow-deep analysis are transformed into the dominance constraints formalism and resolved by the underspecification resolver UTool. First-order expressions can serve as a formal knowledge representation of natural text and thus can be utilized in knowledge engineering or textual reasoning. At the end of this paper, we describe their application for recognizing textual entailment.**

Index Terms—**recognizing textual entailment; logical inference; HPSG-based text analysis; first-order logics**

I. Introduction

MANY applications depend on a formal representation of natural language sentences in form of *first-order logic* (FOL). For instance, in recognizing textual entailment [1] the approaches based on logical inference depend on FOL formulas as a semantic representation of the input texts. Furthermore, enhanced first-order knowledge representations of natural language sentences can be accomplished by integrating knowledge from resources like the lexical base *WordNet* [2] or the ontological database *YAGO* [3] into FOL formulas. Other applications can be found in the area of *knowledge engineering* and *information integration* (see, e.g., [4]).

Basically, for a production of fine-grained FOL expressions a broad coverage of syntactic structures and English words is preferable. To this end, the *English Resource Grammar* (ERG, see [5]), a broad-coverage, linguistically precise *Head-driven Phrase Structure Grammar* (HPSG) of English can be used. ERG utilizes *Minimal Recursion Semantics* (MRS, see [6]) as a formalism for building scope underspecified semantic representations from HPSG grammars.

Combining deep parsing techniques with shallow ones can make parsing processes more robust, i.e., less error-prone and faster [7]. For that reason, we use *Heart of Gold* (HOG, see [7]), a framework for combining NLP components like, e.g., shallow statistical parsers, named entity recognizers, and deep syntactic parsers.

Neither literature nor implemented systems producing FOL from an HPSG grammar with RMRS as semantic formalism could be found so far. For that reason, we decided to build such a system which we present in this paper.

Related work: Other approaches of wide coverage syntactic and semantic analysis with FOL output as input for textual entailment were presented in [8]. However, in contrast to our approach they are not based on a purely linguistically motivated grammar formalism with a broad coverage like, e.g., ERG. Furthermore, our formula generation system has successfully been employed as underlying analysis basis in a framework for recognizing textual entailment (see [9]).

II. Semantic Representation Formalisms

In this section we describe shortly the semantic formalisms MRS and RMRS on which our application builds.

Minimal Recursion Semantics: Scope underspecification is a well-known technique in computational semantics of natural language [10]. MRS is a description language over formulas of FOL languages with *generalized quantifiers*. For instance, the sentence *"Every wizard acts in a circus"* illustrates the well-known problem of scopal ambiguity. Is it one and the same circus in which every wizard acts or are there possibly several different circuses in which the wizards act? Thus, the sentence has two scopal *readings* which can be represented by the following FOL formulas:

$$a(x_9, circus(x_9), every(x_5, wizard(x_5), and(act(e_2, x_5), in(e_2, x_9)))) \quad (1)$$

$$every(x_5, wizard(x_5), a(x_9, circus(x_9), and(act(e_2, x_5), in(e_2, x_9)))). \quad (2)$$

MRS allow multiple formulas, which differ only in their scopal configuration like, e.g., (1) and (2), to be expressed with exactly one single compact formula. To achieve this, in MRS predicates of the formulas are decoupled from each other by removing any nesting of predicates, assigning different *labels* $l_1, l_2, ...$ to them, and adding *holes* $h_1, h_2, ...$ to scope relevant predicates. A single scope underspecified representation of (1) and (2) above can then be given as an MRS by

$$\begin{aligned} < \{ & l_3 : every(x_5, h_6, h_4), l_7 : wizard(x_5), \\ & l_8 : act(e_2, x_5), l_8 : in(e_{10}, e_2, x_9), \\ & l_{11} : a(x_9, h_{13}, h_{12}), l_{14} : circus(x_9)\}, \\ & \{h_6 \ qeq \ l_7, \ \ h_{13} \ qeq \ l_{14}\} > \end{aligned} \quad (3)$$

978-83-60810-27-9/09/$25.00 © 2010 IEEE

More specifically, (3) contains a set of labeled (decoupled) first-order predicates ($l_3 : every, l_7 : wizard, ...$) with holes ($h_6, h_4, ...$) and a set of *qeq-constraints* ($h_6\, qeq\, l_7, ...$). A *qeq-constraint* states a directive enforcing a particular label l to be in the scope of a particular hole h. In (3), the first qeq-constraint enforce label l_7 to be in the scope of h_6. Scope specified formulas are obtained by assigning, or *plugging*, labeled predicates to holes in a manner that is consistent with the qeq-constraints. For instance, (1) is obtained by plugging l_7 into h_6, l_{14} into h_{13}, l_3 into h_{12}, and l_8 into h_4. Such formalized assignments are called also pluggings.

Robust Minimal Recursion Semantics: RMRS is a generalization of MRS. It can not only be underspecified for scope as MRS, but also partially specified, e.g., when some parts of the text cannot be resolved by a given NLP component. Furthermore, in RMRS due to possible lack of morphological analysis, predicates are allowed to lack for their arguments. Hence, it can be used as a semantic representation formalism of shallow NLP components. HOG supports integration of shallow NLP components by using RMRS as an exchange format. Additionally, RMRS defines an in-group relation ing which describes conjunctively connected labels, e.g., the in-group relation $\{h8\ ing\ h10001\}$ in Figure 2. For a detailed description of MRS and RMRS see [6] and [11], respectively.

III. Generation of FOL Formulas

The application for generation of FOL formulas illustrated in Figure 1 consists of two parts:

a) Semantic analysis with the shallow-deep approach realized by HOG, and
b) Resolving of RMRS realized by UTool.

Aditionally, a graphical user interface (GUI) enclosing these parts was developed to control the analysis process and to inspect the results of the analysis components.

HOG is configured to control the overall workflow of the hybrid shallow and deep analysis. In particular, it transforms and transports data among various NLP components of the application. First, HOG processes the input text and computes its semantic representation as RMRS. Afterwards, the generation of *pluggings* representing readings of the RMRS is performed with the underspecification solver *UTool* [12]. Finally, the FOL formulas are constructed out of the computed pluggings and the original RMRS. In the following the processing steps are described in more detail.

Syntactic-semantic analysis: Shallow parsing techniques are used to retrieve superficial information from the sentences without a deep structure. In our system shallow parsing begins with tokenization by JTok (distributed with HOG) which splits the sentence by its tokens. Those tokens are passed to the statistical part-of-speech tagger TnT [13]. In parallel to these two components, SProUT [14], a finite state machine named entity recognizer, prepares information about named entities found. The results from TnT and SProUT are merged together via XML-transformation from HOG and prepared for input to the deep HPSG parser PET [15]. In that manner, PET is supplied with information for words which possibly are not contained in the deep HPSG lexicon (e.g., unusual named entities). Afterwards, PET parses the pre-annotated input text by employing the HPSG grammar ERG. With the help of this processing step, a fully syntactic annoted phrase structure is computed, from which predicate argument structures can directly be established. During the deep HPSG analysis, PET composes (robust minimal recursion) semantics according to the semantic algebra for HPSG grammars [16]. According to the analysis procedure described above, HOG produces the RMRS in Figure 2 for the following example sentence:

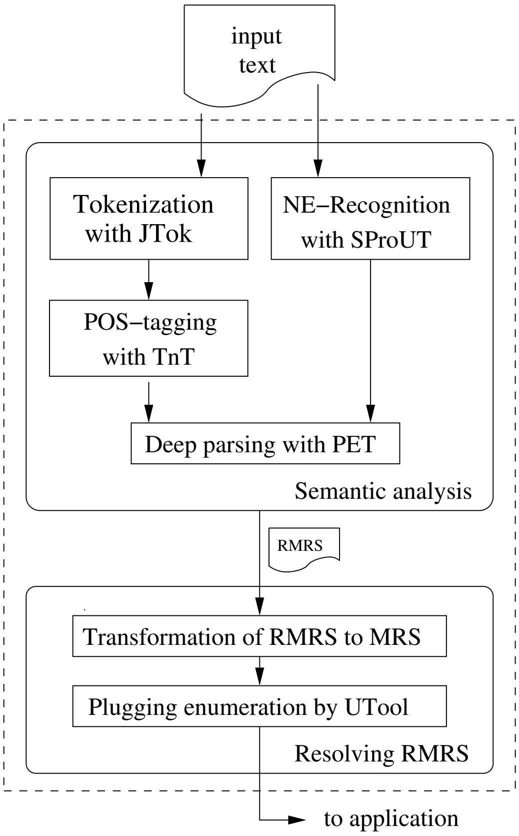


Fig. 1. Overall system for formula generation.

```
A wizard acts in a circus show in Paris.
```

Resolving RMRS: Unfortunately, a sentence with n quantifiers can have up to $n!$ readings [6], e.g., RMRS in Figure 2 with the quantifiers a_q, $udef\_q$, and $proper\_q$ has 4! scopal readings. More adverse is that about 8% of the sentences of the Rondane treebank provided with ERG have more than 100,000 readings according to the ERG analyses, and about 4% have more than one million readings [17]. Thus, it is required to enumerate all readings efficiently and to eliminate those which are logically equivalent. These tasks are performed by UTool in time of $O(n^2)$ per solved form (see [18]).

As in Section II described, for MRS the scopal readings are obtained by plugging labels l to holes h. RMRS has to be resolved analogously. However, in its standard configuration, UTool resolves only MRS. Therefore, by following the in-

$$\begin{bmatrix}
\text{TEXT} & \\
\text{TOP} & h1 \\
\text{RELS} & \left\{ \begin{array}{lll}
\begin{bmatrix} \textit{\_a\_q} & \\ \text{LBL} & h3 \\ \text{ARG0} & x5 \\ \text{RSTR} & h6 \\ \text{BODY} & h4 \end{bmatrix} &
\begin{bmatrix} \textit{\_wizard\_n} & \\ \text{LBL} & h7 \\ \text{ARG0} & x5 \end{bmatrix} &
\begin{bmatrix} \textit{\_act\_v} & \\ \text{LBL} & h8 \\ \text{ARG0} & e2 \\ \text{ARG1} & x5 \end{bmatrix} \\
\begin{bmatrix} \textit{\_in\_p} & \\ \text{LBL} & h10001 \\ \text{ARG0} & e10 \\ \text{ARG1} & e2 \\ \text{ARG2} & x9 \end{bmatrix} &
\begin{bmatrix} \textit{\_a\_q} & \\ \text{LBL} & h11 \\ \text{ARG0} & x9 \\ \text{RSTR} & h13 \\ \text{BODY} & h12 \end{bmatrix} &
\begin{bmatrix} \textit{compound\_rel} & \\ \text{LBL} & h14 \\ \text{ARG0} & e16 \\ \text{ARG1} & x9 \\ \text{ARG2} & x15 \end{bmatrix} \\
\begin{bmatrix} \textit{udef\_q\_rel} & \\ \text{LBL} & h17 \\ \text{ARG0} & x15 \\ \text{RSTR} & h19 \\ \text{BODY} & h18 \end{bmatrix} &
\begin{bmatrix} \textit{\_circus\_n} & \\ \text{LBL} & h20 \\ \text{ARG0} & x15 \end{bmatrix} &
\begin{bmatrix} \textit{\_show\_n} & \\ \text{LBL} & h10002 \\ \text{ARG0} & x9 \\ \text{ARG1} & u21 \end{bmatrix} \\
\begin{bmatrix} \textit{\_in\_p} & \\ \text{LBL} & h10003 \\ \text{ARG0} & e23 \\ \text{ARG1} & e2 \\ \text{ARG2} & x22 \end{bmatrix} &
\begin{bmatrix} \textit{proper\_q\_rel} & \\ \text{LBL} & h24 \\ \text{ARG0} & x22 \\ \text{RSTR} & h26 \\ \text{BODY} & h25 \end{bmatrix} &
\begin{bmatrix} \textit{named\_rel} & \\ \text{LBL} & h27 \\ \text{ARG0} & x22 \\ \text{CARG} & \text{Paris} \end{bmatrix}
\end{array} \right\} \\
\text{HCONS} & \{h6 \text{ qeq } h7, h13 \text{ qeq } h14, h19 \text{ qeq } h20, h26 \text{ qeq } h27\} \\
\text{ING} & \{h8 \text{ ing } h10001, h8 \text{ ing } h10003, h14 \text{ ing } h10002\}
\end{bmatrix}$$

Fig. 2. RMRS of shallow deep analysis of the example sentence as attribute value matrix.

structions from [11], we developed an XSLT-based procedure for transforming RMRS into MRS to make the output RMRS of HOG suitable for resolving with UTool. Afterwards, UTool can automatically compute the assignments according to which the labels should be linked to holes. In particular, the following procedure is applied (see [17] for details):

1) Translation of MRS into *dominance constraints* [17], a closely related underspecification formalism, and
2) Efficient enumeration of solved forms of the dominance constraints.

Dominance constraints can be represented as *dominance graphs*. Thus, an MRS corresponds to a dominance graph. In a dominance graph originating from an MRS, nodes correspond to MRS labels h whereas (dashed drawn) dominance edges correspond to MRS qeq-constraints. For instance, the RMRS from Figure 2 which was produced by HOG is translated by UTool into dominance constraints which is shown as a dominance graph in Figure 3.

UTool computes all solved forms of the dominance graph according to the algorithm that is based on graph connectivity (see [17]). One possible reading of the dominance graph from Figure 3 (and simultaneously of the RMRS in Figure 2) is given as a solved dominance graph in Figure 4.

Consequently, from each solved dominance graph, UTool recursively generates a set of pluggings. The plugging for Figure 4 looks like the following one:

```
h24(h27,h11(h17(h20,h14),h3(h7,h8)))
```

Finally, we parse all pluggings sequentially and replace each label by its corresponding predicates according to the original MRS. After that procedure is finished, we get the formula for the first plugging:

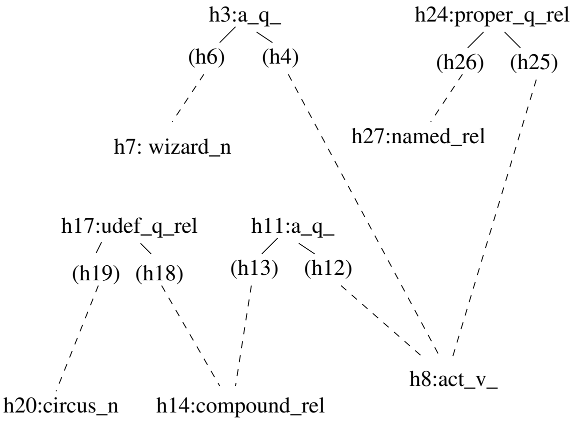


Fig. 3. Dominance graph.

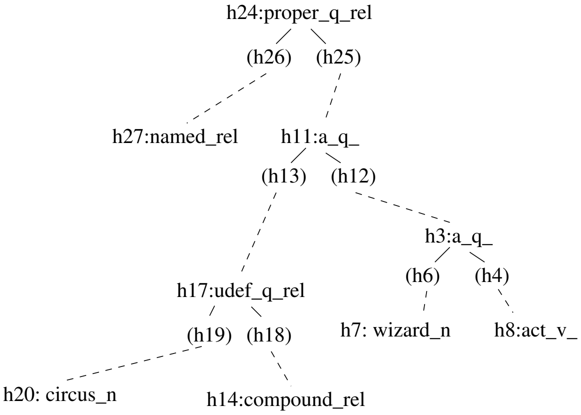


Fig. 4. Solved dominance graph.

```
proper_q_rel(X22,
    named_rel(X22,paris),
    a_q_rel(X9,
        udef_q_rel(X15,
            circus_n_1_rel(X15),and(
            compound_rel(E16,X9,X15),
            show_n_of_rel(X9))),
            a_q_rel(X5,
                wizard_n_1_rel(X5),and(
                act_v_1_rel(E2,X5),and(
                in_p_rel(E10,E2,X9),
                in_p_rel(E23,E2,X22))))))
```

IV. Applications Using FOL Expressions

There are many applications depending on a logical representation of natural language (see, e.g., [1] or [4]). The application presented here was successfully implemented in our experimental system for recognizing textual entailment (RTE) [9]. In RTE [19], the aim is to identify the logical relations between two texts, thesis T and hypothesis H, e.g.,

```
T: A wizard acts in a circus show in Paris.
H: Some magician remains in a capital
   town of France.
```

In particular, given a pair $\{T, H\}$, our system was designed to find answers to the following conjectures [20]:

1) T entails H,
2) $T \wedge H$ is inconsistent, or
3) H is informative with respect to T, i.e., is H a new and consistent information in relation to T?

In the example above, T entails H. To prove it, our system uses model-theoretic approach combined with logical inference. Syntax and semantics of the texts representing T and H are first analyzed with HOG and the resulting semantic representations in form of RMRS are translated into FOL as described in Section III. Afterwards, they are passed to external automated reasoning tools like *model builders* and *theorem provers* which check what kind of logical relation between T and H holds.

V. Conclusion and Future Work

In this paper an application based on a combination of linguistic resources and tools is presented that enable for an efficient generation of first-order logic formulas from a broad coverage HPSG grammar combined with shallow analyses. Since the semantic composition is performed by a deep HPSG parsing, it is required for a successfull generation of FOL expressions that the deep parsing succeeds. Unfortunately, it can fail if well-formedness of syntactical structures is too weak, e.g., in SMS dialogues or transcriptions of spontaneous speech.

The application can be improved through integration of coreference resolvers, so that different object variables pointing to the same individual can be identified. Finally, the application could be supplemented with statistical models describing scopal position settings in natural language sentences, so that scope resolved readings are produced in order of descending probability of their occurrence.

Furthermore, RMRS is the common semantic formalism for the HPSG grammars within the context of the *LinGO Grammar Matrix* [21] like the Japanese HPSG grammar *JaCY* [22], the *Korean Resource Grammar* [23], the *Modern Greek Resource Grammar* [24], the Norwegian *NorSource Grammar* [25], and the Spanish Resource Grammar *SRG* [26]. Because all of these grammars interface to RMRS, an exchange of the ERG in our system can be considered and a high degree of multilinguality achieved.

Acknowledgment

The authors would like to thank Matthias Hecking for many fruitful discussions on the work presented in this paper.

References

[1] P. Blackburn, J. Bos, M. Kohlhase, and H. D. Nivelle, "Inference and Computational Semantics," in *In Third International Workshop on Computational Semantics (IWCS-3)*. Kluwer, 1998.

[2] C. Fellbaum, Ed., *WordNet: An Electronic Lexical Database.* MIT Press, 1998.

[3] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A Core of Semantic Knowledge," in *16th international World Wide Web conference (WWW 2007)*. New York, NY, USA: ACM Press, 2007.

[4] J. Sowa, *Knowledge Representation: Logical, Philosophical and Computational Foundations*. Pacific Grove, CA: Brooks/Cole, 2000.

[5] A. Copestake and D. Flickinger, "An open-source grammar development environment and broad-coverage English grammar using HPSG," in *Proceedings of the Second conference on Language Resources and Evaluation (LREC-2000)*, Athens, Greece, 2001.

[6] A. Copestake, D. Flickinger, C. Pollard, and I. A. Sag, "Minimal recursion semantics: An introduction," *Research on Language and Computation*, vol. 3, pp. 281–332, 2005.

[7] U. Schäfer, "Integrating deep and shallow natural language processing components – representations and hybrid architectures," Ph.D. dissertation, Faculty of Mathematics and Computer Science, Saarland University, 2007.

[8] P. Blackburn and J. Bos, "Underspecification, Resolution and Inference for Discourse Representation Structures," 2004.

[9] A. Wotzlaw and R. Coote, "Recognizing Textual Entailment with Deep-Shallow Semantic Analysis and Logical Inference," in *Proceedings of the 4th International Conference on Advances in Semantic Processing (SEMAPRO 2010)*, Florence, Italy, 2010.

[10] H. Bunt, "Semantic underspecification: Which technique for what purpose?" in *Computing Meaning*, H. Bunt and R. Muskens, Eds. Springer, 2007, vol. 3.

[11] A. Copestake, "Report on the design of RMRS," University of Cambridge, Tech. Rep., 2003.

[12] A. Koller and S. Thater, "Efficient solving and exploration of scope ambiguities," Proceedings of the ACL-05 Demo Session, 2005.

[13] T. Brants, "TnT - A Statistical Part-of-Speech Tagger," in *Proceedings of Eurospeech*, 2000.

[14] W. Drożdżyński, H.-U. Krieger, J. Piskorski, U. Schäfer, and F. Xu, "Shallow Processing with Unification and Typed Feature Structures — Foundations and Applications," *Künstliche Intelligenz*, vol. 1, 2004.

[15] U. Callmeier, "PET. A Platform for Experimentation with Efficient HPSG Processing Techniques," *Journal of Natural Language Engineering*, vol. 6, no. 1, 2000.

[16] A. Copestake, "An Algebra for Semantic Construction in Constraint-based Grammars," in *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL 2001)*, 2001.

[17] S. Thater, "Minimal recursion semantics as dominance constraints: Graph-theoretic foundation and application to grammar engineering," Ph.D. dissertation, 2007.

[18] A. Koller and S. Thater, "An improved redundancy elimination algorithm for underspecified representations," in *ACL*, 2006.

[19] I. Dagan, B. Dolan, B. Magnini, and D. Roth, "Recognizing textual entailment: Rational, evaluation and approaches," *Natural Language Engineering. Special Issue on Textual Entailment*, vol. 15, no. 4, pp. i–xvii, 2009.

[20] P. Blackburn and J. Bos, *Representation and Inference for Natural Language. A First Course in Computational Semantics*. CSLI, 2005.

[21] D. F. Bender, Emily M. and S. Oepen, "The Grammar Matrix: An Open-Source Starter-Kit for the Rapid Development of Cross-Linguistically Consistent Broad-Coverage Precision Grammars ," in *Procedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics.*, Taipei, Taiwan., 2002.

[22] M. Siegel and E. M. Bender, "Efficient deep processing of japanese." in *In Proceedings of the 3rd Workshop on Asian Language Resources and International Standardization. Coling 2002 Post-Conference Workshop*, Taipei, Taiwan, 2002.

[23] K. Jong-Bok and Y. Jaehyung, "Parsing mixed constructions in a typed feature structure grammar," in *Lecture Notes in Artificial Intelligence*, vol. 3248. Springer, Feb. 2005.

[24] V. Kordoni and N. Julia, "Deep analysis of modern greek," in *Lecture Notes in Computer Science*, J.-H. L. Keh-Yih Su, Jun'ichi Tsujii, Ed., vol. 3248. Springer, Berlin 2005.

[25] L. Hellan, "From Grammar-Independent Construction Enumeration to Lexical Types in Computational Grammars," in *Coling 2008: Proceedings of the workshop on Grammar Engineering Across Frameworks*. Manchester, England: Coling 2008 Organizing Committee, August 2008, pp. 41–48.

[26] M. Montserrat, B. Núria, E. Sergio, and S. Natalia, "The spanish resource grammar: pre-processing strategy and lexical acquisition," in *Proceedings of the Workshop on Deep Linguistic Processing, Association for Computational Linguistics (ACL-DLP-2007)*, T. e. a. Baldwin, Ed., 2007.

ISBN 978-83-60810-27-9
ISSN 1896-7094

PSO based modeling of Takagi-Sugeno fuzzy motion controller for dynamic object tracking with mobile platform

Meenakshi Gupta, Laxmidhar Behera, and K. S. Venkatesh
Department of Electrical Engineering,
Indian Institute of Technology Kanpur, Kanpur, India
Email: {meenug, lbehera, venkats}@iitk.ac.in

***Abstract*—Modeling of optimized motion controller is one of the interesting problems in the context of behavior based mobile robotics. Behavior based mobile robots should have an ideal controller to generate perfect action. In this paper, a nonlinear identification Takagi-Sugeno fuzzy motion controller has been designed to track the positions of a moving object with the mobile platform. The parameters of the controller are optimized with Particle swarm optimization (PSO) and stochastic approximation method. A gray predictor has also been developed to predict the position of the object when object is beyond the view field of the robot. The combined model has been tested on a Pioneer robot which tracks a triangular red box using a CCD camera and a laser sensor.**

I. Introduction

OBJECT detection and tracking is an essential ingredient of any motion planning controller employed for mobile robot navigation. Mobile robot navigation is known as the ability of a robot to act based on its knowledge and sensor values in order to reach its goal position as efficiently and as reliably as possible [1]. Wide variety of sensors such as sonar, laser range finder, infra-red, Global Positioning System (GPS) and vision are used for mobile robot navigation. The vision based navigation is widely used [2], since vision gives the rich information about the surroundings. Vision is an attractive sensor as it helps in the design of economically viable systems with simpler sensor limitations. It facilitates passive sensing of the environment and provides valuable information about the scene that is unavailable to other sensors [3].

An ample of work has been done on vision based object tracking. Ramesh et.al. in [4] proposed the Mean shift algorithm for object tracking that can be used for the images with static distribution. The Continuously Adaptive Mean Shift Algorithm (CAMShift), which is an adaptation of the Mean Shift Algorithm has been proposed by Bradski [5] to track the head and face movement using a one dimensional histogram (hue) consisting of quantized channel from the HSV color space. CAMShift operates on a probability density image obtained by histogram back-projection. In this paper, a hybrid CAMShift algorithm [6], that overcomes the assumption of single hue, has been used for object tracking. The computational cost of mean matching algorithm used in the hybrid CAMShift algorithm is high. So, instead of using mean matching algorithm for the detection of object when the CAMShift fails to detect the object, a gray predictor has been used to predict the position of the object. Gray predictor re-initializes the CAMshift window not only when CAMshift fails to detect the object but also when the object goes out from the robot's view.

The outputs of the hybrid CAMShift algorithm are the centroid coordinates of the object in the image frame. In order to obtain the global position and orientation of the object or even just to determine their relative pose, various algorithms of calibration and transformation are required. All the proposed approaches formulate the vision-based navigation problem as a two-step process: first, to transfer the visual features back to pose information, and then make a motion plan in the pose space. The calibration techniques, that transfer the visual features from image space to pose space introduce unnecessary uncertainty into the system. In this paper, a simple transformation technique has been proposed to transfer centroid coordinates of the object from image frame to the robot frame.

Once the coordinates of the object centroid are known in the robot frame, the next task is to design motion controller to effectively track the object. Since the primary focus of machine intelligence and advanced robotics is to capture the human faculties in the robot, fuzzy logic controllers are often a good choice. These controllers are developed to utilize human expert knowledge in controlling various systems and they have capability to express knowledge in the form of linguistic rules. Among various fuzzy modelling themes, the Takagi-Sugeno (T-S) model has been one of the most popular frameworks as it exhibits both high nonlinearity and simple structure [7][8]. In this paper, a T-S fuzzy controller has been modeled to control the motion of the robot while tracking the object. The structure identification of the premise part (i.e. membership functions) of rules of T-S fuzzy controller is carried out using PSO [9], while the identification of consequent part (i.e. weight parameters) of rules of T-S fuzzy controller is carried out using stochastic approximation method [10].

II. Outline of our approach

The goal of this work is to design a vision and laser sensor based optimized motion controller for the mobile robot to make it track the moving object in an effiecient manner. The outline of our approach for tracking the moving object is illustrated in figure 1. The first step inculdes the initialization of the robot and selection of object in the current frame. Next, frames are captured from the resulting video stream and the hybrid CAMShift algorithm is run over to detect the presence of the object in the scene. In case object is detected the algorithm returns the centroid coordinates of the object in image frame. If hybrid CAMShift fails to detect the object then the output of algorithm will be origin coordinates. Failure of the algorithm occurs either due to fast motion of the object or when the object goes out from the robot's view. In case of failure of the algorithm, a gray predictor is used to predict the centroid coordinates of the object. If the predicted x-coordinate lies in the range of image width, then the failure of the algorithm is due to fast motion of the object. In this case, the center coordinates of the CAMShift window are replaced by the predicted coordinates of the gray predictor. In case, when predicted x-coordinate does not lies in the range of image width, the reason of the algorithm failure is the absence of the object in the robot's view. The robot is then commanded to turn by an angle, calculated using the predicted x coordinate as follows:

$$\beta = -\frac{\phi}{2} + \frac{x}{x_{max}}.\phi \qquad (1)$$

where, β is the required turning angle, x_{max} is the maximum x coordinate of the image (width of the image in pixels), ϕ is the view angle of the camera.

The CAMShift window is then reinitialized with the center x coordinate of the window as $(x_{max}/2)$ and center y coordinate of the window as the predicted y coordinate of the gray predictor. In both the cases, the CAMShift window size is taken to be equal to its initial window size. Once the object centroid coordinates are obtained in the image frame, a coordinate converter, as described in section IV, is used to transform the object centroid coordinates from the image frame to the robot frame. The object centroid coordinates in the robot frame are then sent to an optimized T-S fuzzy motion controller, described in section VI, to generate desired translational and rotational velocities for the robot. The robot is then commanded to move with the translational and rotational velocities as generated by the controller.

Rest of the paper is organized as follows. In section III, the gray fuzzy predictor is discussed to predict the position of the object when it goes out from the robot's view or the hybrid CAMShift algorithm fails to detect the object. Section IV gives the details of the transformation of object coordinates from the image frame to the the robot frame. Section V briefly explains the PSO method and the T-S fuzzy model. Section VI presents modelling of the T-S fuzzy controller for the object tracking using PSO and stochastic approximation method. Experimental results are presented in section VI and finally the paper is concluded in section VII.

III. Gray predictor

A system with partial known information and certain unknown information is defined as a gray system. The gray theory, originally developed by Deng [11], employed the method of data generation instead of statistic regulation, to obtain more regular generating sequence from those initial random data. The gray prediction is to establish a gray model extending from the past information to the future based upon the past and present known or undeterminate information. Then the gray model can be used to predict the future variation trend of the system.

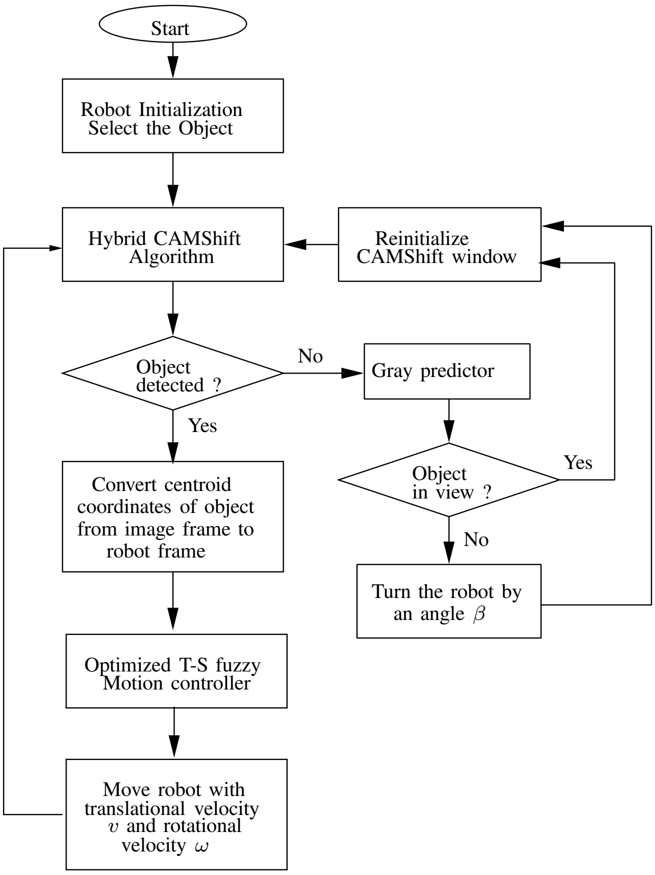


Fig. 1. flow diagram of our approach

Gray Prediction model for tracking The procedure of gray prediction model is as follows:

- Establish the initial sequence from observed data

$$\mathbf{x}^{(0)} = (x^{(0)}(1), x^{(0)}(2), x^{(0)}(3), ..., x^{(0)}(n) \qquad (2)$$

where, $x^{(0)}(i)$ represent the base line data with respect to time i.

- Generate the first-order accumulated operation sequence (AGO) sequence $\mathbf{x}^{(1)}$ based on the initial sequence $\mathbf{x}^{(0)}$

$$\mathbf{x}^{(1)} = (x^{(1)}(1), x^{(1)}(2), x^{(1)}(3), ..., x^{(1)}(n) \qquad (3)$$

where,

$$x^{(1)}(k) = \sum_{i=0}^{k} x^{(0)}(i) \qquad (4)$$

- Compute the mean value of the first-order AGO sequence:

$$z^{(1)}(k) = 0.5x^{(1)}(k) + 0.5x^{(1)}(k-1) \qquad (5)$$

- Define the first-order gray-differential equation of sequence $\mathbf{x}^{(1)}$ as:

$$\frac{dx^{(1)}(k)}{dk} + az^{(1)}(k) = b \qquad (6)$$

  where, a and b express the estimated parameters of gray prediction model.

- Utilize the least square estimation, we can drive the estimated first-order AGO sequence $\hat{x}^{(1)}(k+1)$ and the estimated inversed AGO sequence $\hat{x}^{(0)}(k+1)$ as follows:

$$\hat{x}^{(1)}(k+1) = \left[x^{(0)}(1) - \frac{b}{a}\right]e^{-ak} + \frac{b}{a} \qquad (7)$$

$$\hat{x}^{(0)}(k+1) = \hat{x}^{(1)}(k+1) - \hat{x}^{(1)}(k) \qquad (8)$$

  where, parameter a and b can be conducted by following equations:

$$\left[\begin{array}{c} a \\ b \end{array}\right] = (B^T B)^{-1} B^T y \qquad (9)$$

$$B = \left[\begin{array}{cc} -\frac{1}{2}(x^{(1)}(1) + x^{(1)}(2)) & 1 \\ -\frac{1}{2}(x^{(1)}(2) + x^{(1)}(3)) & 1 \\ \cdots & .. \\ -\frac{1}{2}(x^{(1)}(n-1) + x^{(1)}(n)) & 1 \end{array}\right] \qquad (10)$$

$$y = \left[x^{(0)}(2), x^{(0)}(3), x^{(0)}(4), ..., x^{(0)}(n)\right]^T \qquad (11)$$

In this paper, we modeled two gray predictor to predict the x and y coordinate of the object centroid. We used last six centroid coordinates to generate accumulated operation sequence.

IV. Transformation of the object coordinates from the image frame to the robot frame

To design any motion controller for the robot, it is necessary to transform the coordinates of the gravity center of the object from the image frame to the robot frame. As x coordinate of the object in the image frame is invariant to the size (width) of the object, it can be used to calculate the direction of the object with respect to the robot. But y coordinate of the object in the image frame varies with the size (height) of the object, so, it cannot be used to determine the distance of the object with respect to the robot. To get the accurate coordinates of the gravity center of the object in the robot frame, x co-ordinate of the object in the image frame and laser sensor data are used. As view angle of the camera and size of the image are known, direct relationship between the x coordinate of the object in the image frame and angle of the object w.r.t. the robot can be established as:

$$\alpha = -\frac{\phi}{2} + \frac{x}{x_{max}}\phi \qquad (12)$$

α is the angle of the of the object w.r.t. the robot at a particular x coordinate of the object in the image frame. In our case, camera view angle is $40°$, and size of image is 640×480.

Once the angle of the object w.r.t. the robot is obtained, the distance of the object can easily be measured with the help of the laser data. Laser can give the distance reading from $-90°$ to $+90°$ with a resolution of $0.5°$. The nearest integer value of the angle α is choosen and the minimum of the three laser reading at $\alpha - 1$,α and $\alpha + 1$ is taken. As the object is in motion, so it may happen that at a particular angle α, the laser ray may not come back from the object. So, $1°$ offset is put to get the accurate reading. Using this transformation, the polar coordinates (α, r) of the gravity center of the object in the robot frame are obtained.

V. Description of particle swarm optimization and T-S fuzzy model

A. Particle swarm optimization

PSO is an optimization technique developed by Kennedy and Eberhart [9]. It is inspired by the formation of swarms by animals such as bird flocking and fish schooling. The principle behind PSO is that each individual in the swarm, called a particle, will move towards the best performing particle in the swarm while exploring the best experience each particle has [12]. The particle update their velocities as follows [13]:

$$\begin{aligned} v(k+1) &= \eta(k+1).v(k) \\ &\quad + c_1(k+1).r_1.(P^{lbest}(k) - P(k)) \\ &\quad + c_2(k+1).r_1.(P^{gbest}(k) - P(k)) \end{aligned} \qquad (13)$$

where, k is the generation number, v denote the particle velocities, η denotes the inertia weight, $r_1$and r_2 are random numbers between 0 and 1, c_1 is the cognitive parameter, c_2 is the social parameter, P^{lbest} is the local best solution and P^{gbest} is the global best solution of the group.

The inertia weight η represents the degree of momentum of the particles. This parameter is used for balancing between local and global explorations. In early generations, it is set higher, so that the particles are allowed to have much exploration capability and pursue an aggressive search of the solution space. Once the algorithm is found to converge towards the optimum, this coarse tuning is gradually converted to finer tuning by making η smaller in later generations. In this paper, a linearly adaptable inertia weight is employed [14], which starts with a high value η_{max} and linearly decreases to η_{min} at the maximum number of generations. This means that $\eta(k+1)$ is calculated from

$$\eta(k+1) = \eta_{max} - \frac{\eta_{max} - \eta_{min}}{Gen_{max}}.Gen \qquad (14)$$

where, Gen_{max} is the maximum number of generations and Gen is the current generation number.

The constants c_1 and c_2 represent the weights of the stochastic acceleration terms that pull each particle toward the local best and global best positions. With a large cognitive component and small social component at the beginning, particles are allowed to move around the search space, instead of moving toward the best solution. In the latter part of the optimization, a small cognitive component and large social component are used, to allow the particles to converge on the global optima. In this paper, we use linearly time-varying acceleration coefficients over the evolutionary procedure. Therefore, the acceleration coefficients $c_1(k+1)$ and $c_2(k+1)$ can be expressed as follows [15]:

$$c_1(k+1) = c_{1max} - \frac{c_{1max} - c_{1min}}{Gen_{max}}.Gen \tag{15}$$

$$c_2(k+1) = c_{2min} - \frac{c_{2min} - c_{min}}{Gen_{max}}.Gen \tag{16}$$

To limit the searching space v is limited to be within a certain range of $v_{min} \leq V \leq v_{max}$.

The particle positions is updated as:

$$P(k+1) = P(k) + v(k+1) \tag{17}$$

Where, P is the positions of the particle. P should also be limited to be within a certain range of $P_{min} \leq P \leq P_{max}$ for limiting the searching space.

The evaluation of the particle performance is based on a problem specific fitness function that decides the 'closeness' of the particle to the optimal solution. The particle which has the best fitness in any generation till the current generation is known as global best particle and its position is known as global best solution (P^{gbest}). For each particle, there is a local best solution (P^{lbest}), which is the position of the particle at generation g in which that particle has the best fitness till the current generation.

B. The T-S Fuzzy Model

The T-S fuzzy model constructs a map from input space to output space through a fuzzy average of local models. The local models can be either linear or nonlinear. In this paper, the map is built using local linear models. The i^{th} rule in a T-S fuzzy model with k inputs has the following form:

$$R^i : IF\ x_1\ is\ A_1^i\ and\ x_2\ is\ A_2^i\ ....x_k\ is\ A_k^i$$
$$THEN \qquad y_i = w_i^T x + b_i$$

where, R^i is the i^{th} rule (i = 1,2....m); m-is the number of rules; x =$[x_1$,x_2,.....$x_k]^T$ is a input vector; $w_i \in \mathbf{R}^{1\times k}$; b_i is the constant; A_1^i,A_2^i,....,A_k^i are fuzzy sets and y_i is the consequence of the i^{th} rule.

The possibility that the i^{th} rule will fire is given by the minimum of all the membership functions associated with the i^{th} rule (Mamdani's implication [16]).

$$\mu_i = \min(\mu_1^i, \mu_2^i, ...\mu_k^i) \tag{18}$$

where, μ_i is the membership value for the i^{th} rule and μ_k^i is the membership value of the k^{th} input in the A_k^i fuzzy set. The weighted membership value for the i^{th} rule is given by:

$$\sigma_i = \frac{\mu_i}{\Sigma_{i=1}^k \mu_i} \tag{19}$$

By using center of gravity method for defuzzification, the overall output of the T-S fuzzy system is given by:

$$y = \sum_{i=1}^{k} \sigma_i * y_i \tag{20}$$

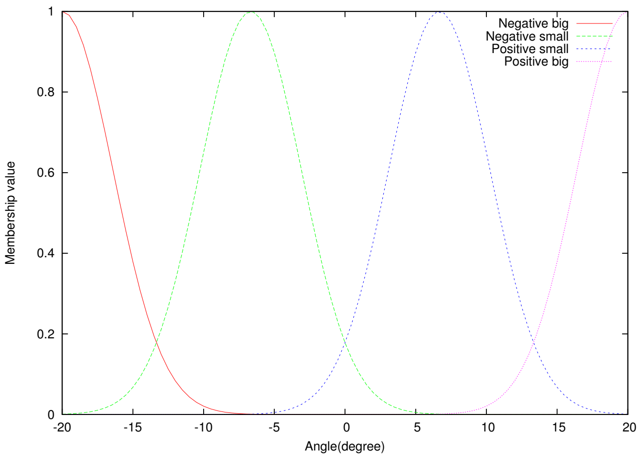


Fig. 2. Fuzzy zones for angle α

VI. Modelling of the T-S fuzzy controller for object tracking

Modelling of the T-S fuzzy controller required the followings steps:

1) Acquisition of real-time data for training and testing the T-S fuzzy model.
2) Design and develop the T-S fuzzy model using training data.
   - Identification of the parameters of the premise part of the rules.
   - Identification of the parameters of the consequent part of the rules.
3) Verification of the designed T-S fuzzy controller to demonstrate the desired object tracking behavior on the Pioneer robot.

A. Acquisition of real-time data for training and testing

To collect the training and testing data, the robot had driven manually using a joystick to set both translational and rotational velocities, guiding the robot to follow the moving red box. The human driver had no visual contact with the object and the robot, he used the robot's camera video stream and his sensor motor coordination to steer the robot towards the box.

The robot was driven in this manner for one hour. During this time, the placement angle of the object w.r.t. the robot and the distance of the object from the robot, and the robot's translational and rotational velocities were logged every $500ms$. The robot's maximum translational velocity is set to 750 mm/sec.The object position w.r.t. the robot was estimated with the hybrid CAMShift algorithm and laser data, as discussed in section II and IV.

B. Design of T-S fuzzy controller

For object tracking behavior the inputs to the T-S fuzzy model are the object position in the robot frame, i.e., object placement angle w.r.t. the robot and the object distance from

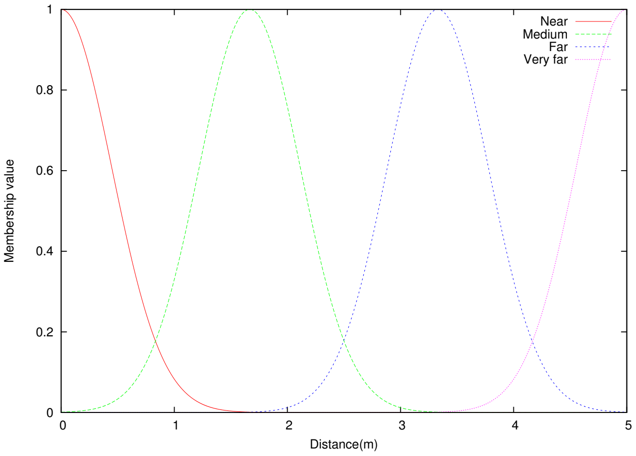


Fig. 3. Fuzzy zones for distance d

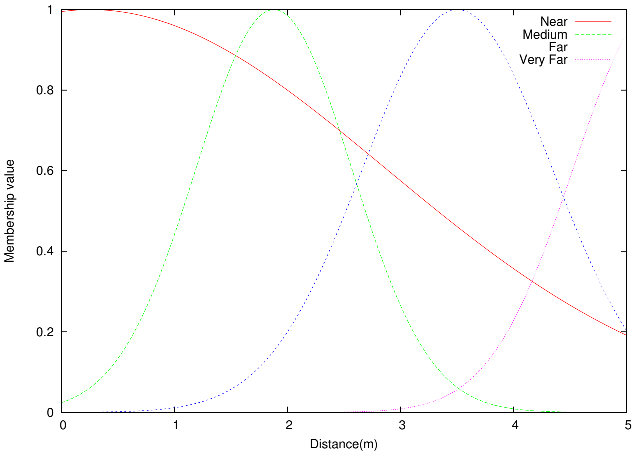


Fig. 4. Fuzzy zones for distance d

the robot, and the current translational and rotational velocities ($V(t)$,$\omega(t)$) of the robot. Outputs of the T-S fuzzy model are desired translational and rotational velocities ($V(t+1)$,$\omega(t+1)$) of the robot required to reach the object.

Placement angle (α) and distance (d) of the object are fuzzified into four zones each as shown in figure 2 and 3 respectively. Initially, a Gaussian membership function is taken for each zone and later on membership functions are updated using particle swarm optimization. The total number of rules (k) is equal to the product of number of zones for α and number of zones for d ,i.e., 4 ×4 = 16. For each rule the membership value is calculated as follows:

$$\mu_i = \prod(\mu_{\alpha_i}, \mu_{d_i}) \tag{21}$$

where, μ_{α_i} is the membership value of α in i^{th} rule and μ_{d_i} is the membership value of d in i^{th} rule.

For each rule the outputs are given as:

$$V_i(t+1) = w_i^T x + b_i \tag{22}$$

$$\omega_i(t+1) = w_i^{'T} x + b_i' \tag{23}$$

where, x = $[d$,α,$V(t)$,$\omega(t)]^T$ is a input vector; w_i and w_i' are parameter vectors to be updated to make the model for a given behavior and b_i and b_i' are constants.

The overall outputs of the T-S fuzzy system are given as:

$$V(t+1) = \sum_{i=1}^{k} \sigma_i * V_i(t+1) \tag{24}$$

$$\omega(t+1) = \sum_{i=1}^{k} \sigma_i * \omega_i(t+1) \tag{25}$$

After defining the structure for the T-S fuzzy controller, the parameters of the structure are identified as follows:

1) Identification of the parameters of the premise part of the rules using particle swarm optimization: The membership functions used in the premise part of the T-S fuzzy controller are all of Gaussian forms. The parameters that define the Gaussian membership function are mean m and the deviation σ. The Gaussian membership function is defined as:

$$G_{mf}(x) = e^{-\frac{(x-m)^2}{2\sigma^2}} \tag{26}$$

Since, we have defined 2 inputs each having 4 zones, so there are 8 membership functions and a total of 16 parameters that need to be updated. Therefore, in the PSO, each particle is to have 16 dimensions. We have defined 20 particle in the swarm and total searching iterations has been set to 2000. The minimum and maximum value of inertia weight have been set to be 0.1 and 0.9 respectively. For weights of the stochastic acceleration minimum and maximum value have been set to 0.5 and 2.5 respectively. The fitness function that evaluates the fitness of each particle has been defined as:

$$f(x(k) = \sum_{g=0}^{G_{max}} \epsilon^2 \tag{27}$$

where, $x(k)$ is the k^{th} particle of the swarm, $Gmax$ is the maximum number of generation and ϵ is the output error. At the completion of the all iterations, the membership functions for the inputs of the T-S fuzzy controller have been modified significantly as shown in figure 4 and 5.

2) Identification of parameters of the consequent part of the rules using stochastic method: There are several methods described in the literature for the parameters estimation [17]. Least-mean square algorithm based on the idea of stochastic approximation is widely used. It was developed by Widrow and Hoff [18] and is used for adjusting the weights in a liner adaptive system. For consequent part parameters identification we have used stochastic approximation method, as described by the Jelena in [10]. Once the training is over, the learned T-S fuzzy model with 16 rules is validated using test data. One sample rule of the learned T-S fuzzy model is given here:

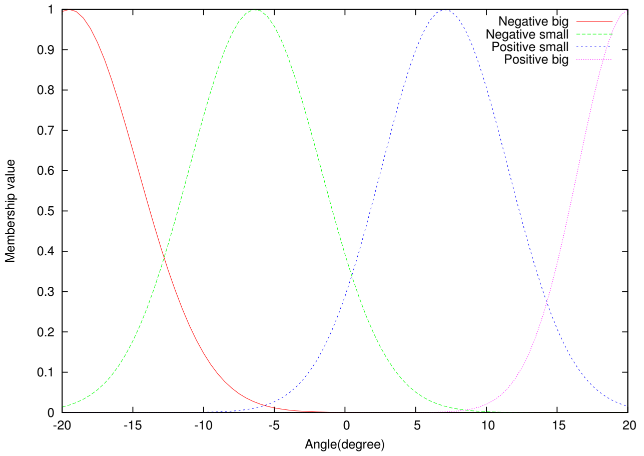


Fig. 5. Fuzzy zones for angle α

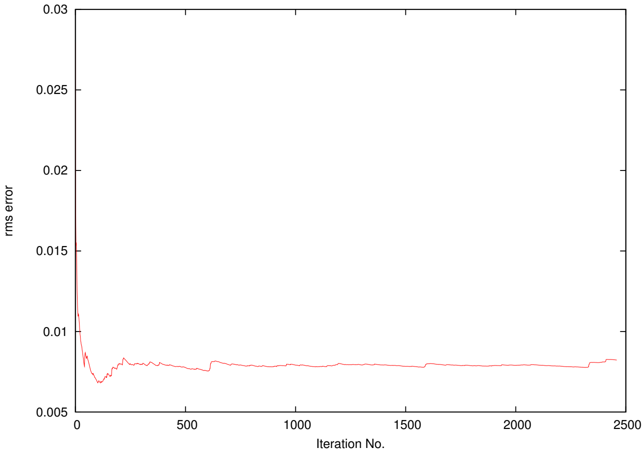


Fig. 6. RMS error during testing

- If α is Negative big and d is Near, then

$$\begin{aligned} V_1(t+1) &= 0.6506d - 0.00009\alpha - 0.00173V(t) \\ &\quad +0.00007\omega(t) - 0.0169 \\ \omega_1(t+1) &= 0.0046d + 1.37\alpha - 0.0295V(t) \\ &\quad -0.026\omega(t) - 0.173 \end{aligned}$$

C. Experimental results and Observations

We applied the testing data set and observed the error. The RMS error during testing of the model is shown in figure 6. Figure 7 shows the desired and actual translational velocities of the robot for the object tracking behavior during testing. Desired and actual rotational velocities of the robot for the object tracking behavior during testing is shown in figure 8. As compared to our previous work [19], the rms error has reduced significantly. The reason of error reduction is the updation of the parameters of the membership functions in this work. Searching mode for the robot, which is a slow process is not required in this case, as the grey predictor predict the position of the object quite efficiently, when the algorithm fails to detect the object. A test run of the designed model in action can be seen in the following video [20].

VII. Conclusion

This paper has presented a T-S fuzzy model based sensor-motor coordination scheme for object tracking. The object is detected using vision sensor and the laser is used to transform the image coordinates of the object into the robot frame coordinates. Particle swarm optimization is used to optimized the parameters of membership functions of the T-S fuzzy model. Since the robot behavior is expressed as if-then rules, the behavior modelling can be easily interpreted. A gray predictor is developed to predict the position of object, when it is not in the view of the robot. The control scheme has been implemented on a Pioneer robot for tracking a triangular box. The experimental result shows that the robot is able to track any object in any arbitrary trajectory using the proposed controller.

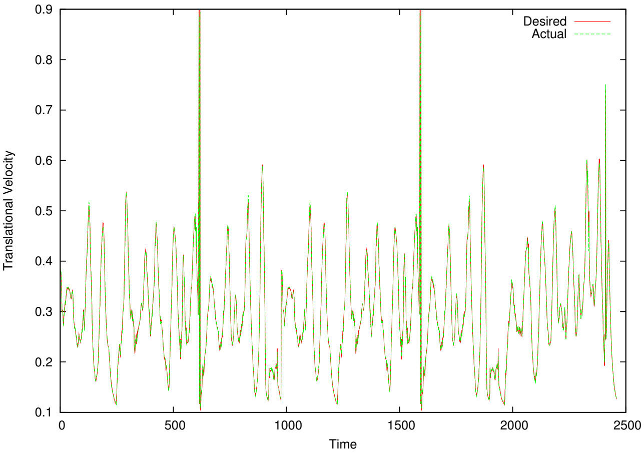


Fig. 7. Desired and actual Translational Velocity

References

[1] R. Siegwart and I. Nourbakhsh, *Introduction to Autonomus Mobile Robots*. The MIT Press, Massachusetts Institute of Technology Cambridge, 2004.

[2] A. Ohya, A. Kosaka, and A. Kak, "Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing," *IEEE Transcations on Robotics and Automation*, vol. 14, pp. 969–978, December 1998.

[3] D. S. Kumar, "Vision-based robot navigation using an online visual experience," Master's thesis, Center for Visual Information Technology, International Institute of Information Technology, Hyderabad, India, June 2007.

[4] V. Ramesh, D. Comanciu, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2000, pp. 142–149.

[5] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, 1998, 2nd Quarter.

[6] M. Gupta, B. Uggirala, and L. Behera, "Visual navigation of a mobile robot in a cluttered environment," in *Proceedings of the 17th International Federation of Automatic Control World Congress*, July 2008, pp. 14 816–14 821.

[7] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.

[8] M. Sugeno, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, vol. 28, pp. 15–33, 1988.

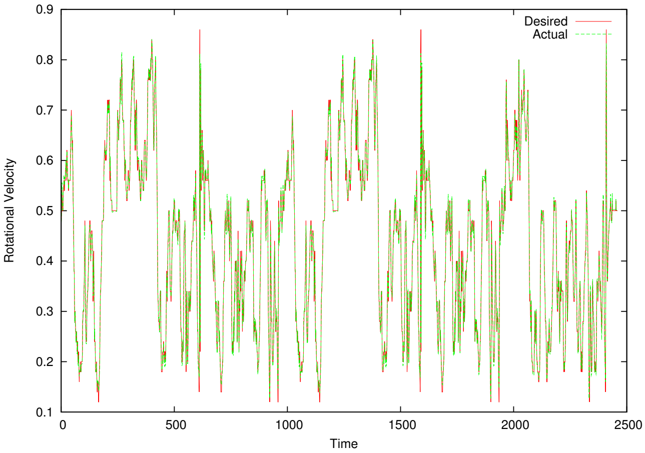


Fig. 8. Desired and actual Rotational Velocity

[9] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *IEEE Internatinal Conference on Neural Networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[10] J. Godjevac and I. f Ecublens, "A learning procedure for a fuzzy system: application to obstacle avoidance," in *Proceedings of the International Symposium on Fuzzy logic*, 1995, pp. 142–148.

[11] D. Julong, *Grey Forecasting Control, In Grey System*. China Ocean Press, 1988.

[12] G. Fang, N. M. Kwok, and Q. Ha, "Automatic fuzzy membership function tuning using the particle swarm optimisation," in *Pacific-Asia Workshop on Computational Intelligence and Industrial Application*. IEEE, 2008, pp. 324–328.

[13] C. N. Ko and C. J. Wu, "A pso-tuning method for design of fuzzy pid controllers," *Journal of Vibration and Control*, December 2007.

[14] R. C. Eberhart and Y. Shi, "Comparison between genetic algorithms and particle swarm optimization," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, 1998, pp. 611–616.

[15] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, 2004.

[16] T. J. Ross, *Fuzzy logic with engineering applications*. McGraw-Hill, International Editions, 1995.

[17] K. J. Astrom and P. Eykhoff, "System identification - a survey," in *Aotomatica*, 1971, pp. 123–168.

[18] L. Ljung and T. Soderstrom, *Theory and Practice of Recursive Identification*, A. S. Willsky, Ed. The MIT Press Signal Processing, Optimization, and Control Series, 1983.

[19] M. Gupta, T. N. Kumar, L. Behera, K. S. Venkatesh, and A. Dutta, "Environment modelling in mobile robotics through takagi-sugeno fuzzy model," in *Irish Signal and System Conference*, June 2009.

[20] M. Gupta, "Object tracking," http://www.youtube.com/watch?v=HICmUFJRgyY.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Hierarchical Object Categorization with Automatic Feature Selection

Md. Saiful Islam, Andrzej Sluzek
Centre for Computational Intelligence, School of Computer Engineering
Nanyang Technological University, Singapore-639798
Email: saiful@pmail.ntu.edu.sg, assluzek@ntu.edu.sg

Abstract—**In this paper, we have introduced a hierarchical object categorization method with automatic feature selection. A hierarchy obtained by natural similarities and properties is learnt by automatically selected features at different levels. The categorization is a top-down process yielding multiple labels for a test object. We have tested out method and compared the experimental results with that of a nonhierarchical method. It is found that the hierarchical method improves recognition performance at the level of basic classes and reduces error at a higher level. This makes the proposed method plausible for different applications of computer vision including object categorization, semantic image retrieval, and automatic image annotation.**

I. Introduction

IN computer vision, object categorization is the problem of deciding the class of an object present in a given image. The major challenge in object categorization is the extraction of suitable features from the images with consideration of geometric and photometric transformations of an object in different image planes, intra class variations, object deformations, partial occlusions, and background cluttering etc. Different kinds of extractable features have been proposed over the last three decades [1]. It is generally agreed that a particular feature may not be suitable for all classes of objects. Some features may be specialized for discriminating between certain classes while being useless in the general cases. Hence, dividing the problem into sub-problems and selecting the appropriate subset of extractable features for each of them may be useful.

Different classes of objects could be clusterized by their natural superclass-subclass relationships and can be represented by a hierarchical (parent-child) structure. For example, taxonomy of biological objects is generally arranged in such a structure. In a hierarchy, a subclass by definition has the same properties and constraints as the superclass plus some additional properties or constraints. Human is capable to perceive such superclass-subclass relationship. Findings of visual neuroscience [2] also show that a combination of several kinds of features is used in primate cerebral cortex for categorization. Features are used in a hierarchical fashion starting from simple features and later using a combination of several features. Our work is motivated by these two observations: firstly it is assumed that there exists a hierarchical relation between object classes and secondly visual categorization is a hierarchical process by using a combination of suitable features.

In general, a hierarchy (e.g. taxonomy) is a result of long time research in the scientific community. The hierarchies, discovered by different researchers, may vary depending on the available data, resources, and method of analysis of data using striking features. Learning of a hierarchy (e.g. taxonomy of biological objects) is a different problem than discovering it. Basically, the learning is a supervised process - it is a process of becoming a specialist in the domain. Such a specialist can be relied on for authentic specification of an unknown instance of an object class (i.e. determining Kingdom, Phylum, Class, Order, ..., Species). In visual object categorization, we learn a given hierarchy by automatically selecting a suitable subset of extractable features and a classification function for each of the superclasses. Once such a hierarchy is learnt the classification method provides a detail specification of an unknown test object (e.g. the given sample is a ‘fruit’ and an ‘apple’). An overview of the method can be found in Fig. 1.

In this work, we have proposed a method of learning a given hierarchy by automatic selection of a set of suitable features for each of the superclasses. The main contribution of this work is to automatically select non uniform features for categorization of diverse class of objects. We consider the superclasses at different levels of hierarchy as independent multi-class classification problems. Then an appropriate classification function (model) is learnt with the selected features. The specification process is carried out in a top-down approach which starts from the root of the hierarchy and follow a single branch of the hierarchy depending on the decision at each internal node until we get a decision about the basic class. This yields a detail specification of the given sample. The proposed method has been tested on different hierarchies and promising results have been obtained.

There are several potential advantages of a hierarchical method [2], [3] such as faster model selection and flexible features choice for improved classification. However, an important objective of hierarchical categorization is to achieve better classification performance on the basic classes. In addition to this we get a detail specification of a

This work was not supported by any organization

given sample with multiple labels. In many applications the ontology of a certain domain may be encoded in terms of object class hierarchy based on their semantic meanings [4]. That is why, hierarchical categorization may be useful for such applications including object classification, semantic image retrieval, image annotation by assigning multiple concepts (i.e. labels) etc.

The outline of the remaining of the paper is as follows. We first review the related work in Section II. We discuss recent advances in hierarchical methods of object classification and learning of such a hierarchy. In Section III we describe our method of hierarchical object categorization. In this section, we also briefly discuss about features and hierarchies used. In section IV, we present the experimental setup, database used, and results. Section V concludes the paper with a discussion of the limitations of current work and our future directions.

II. Related Works

Hierarchical approaches for object categorization have become increasingly popular over the years as it has been regarded as a mean to improve the performance of object classification [3], [5]. Particularly, in pursuit of a general categorization system capable of recognition a vast number of object categories, a need for such a hierarchical structuring has emerged. Hierarchical systems combine existing classes into more complex entities (i.e. superclasses) to achieve more compact object representation enabling fast and robust categorization with better generalization properties [6].

Decision tree [7] based approach is an earlier example to exploit hierarchical feature selection method. Decision tree use heterogeneous and diverse types of features and select them dynamically at different levels. However, this obtained structure does not necessarily follow the semantically meaningful hierarchy (e.g., the subtrees of a decision tree not necessarily correspond to semantically meaningful superclasses.) An extension of this method can be found in [8] where the classification system rather pursuits the hierarchy which is obtained by the semantic information of object class. However, this method uses manually selected features at different levels of hierarchy. In order to improve classification performances, several classification models are also combined together by using them at different levels [5].

The primary concern is how to find the hierarchical relation among the basic classes. Natural hierarchy could be a good choice. Handcrafted hierarchy based on the natural similarity is used for classification in [5]. Hierarchy is constructed through a statistical analysis of elementary similarity in [8]. Some works are also dedicated to discover such a relation automatically. Both supervised and unsupervised learning methods have been investigated. Supervised learning has shown a significant generalization for small sample sets by sharing features between object classes [9]. Recently, unsupervised learning, based on common visual elements also shows comparable performance [4].

III. Hierarchical Categorization

There are two basic steps in object categorization: training and testing. In a regular nonhierarchical object recognition method some train samples for each of the classes are provided. Features are extracted from the training samples and used to learn a classifier (a pre-specified function). For example, *n* SVMs (assuming one-versus-rests approach) are learnt for an *n* class problem. In the testing process, the classifier determines the class of a test object and assigns a label to it. In a simple model, a few specific features (sometimes only one) are used for classification [10], [11]. Recently, automatic feature selection method has emerged [12], where a subset of features from a given set is selected before the training process. However, these selected features remain fixed and are used to learn the classification function. Briefly, in these methods of categorization, all the classes are treated equivalently with the same set of features.

In hierarchical categorization, we assume that a hierarchical relation of the basic object classes is provided in addition to the training samples of each of them. Then, the original classification task is divided into some (hierarchically defined) superclasses. Each of these superclasses is learnt with a suitable subset of features. The two basic steps of hierarchical learning are as follows:

1. Automatic selection of a set of features for each of the superclasses
2. Learning the appropriate classification function for each superclass with the selected features.

Unlike in the regular method, the automatically selected features become non uniform for different superclasses. Once the hierarchy is learnt, a novel test object is categorized in a top-down fashion by assigning multiple labels. Hence we rename the testing process as ‘specification’ which gives a detail description of the test object. The basic idea of hierarchical categorization is show by the block diagram in Fig. 1.

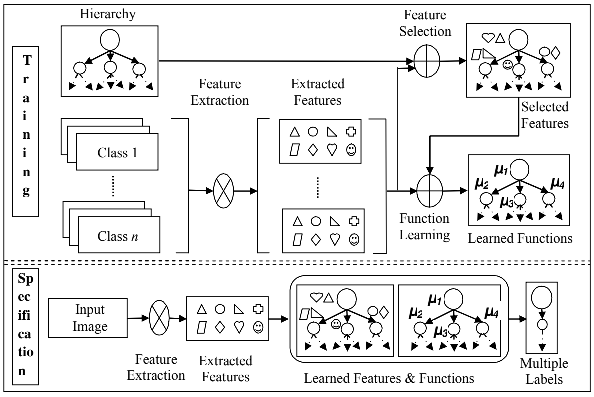


Fig. 1 Block diagram of hierarchical object categorization

We describe the different steps of hierarchical object categorization below. The first step of learning and classification is the extraction of features for respective images. A brief introduction about the image features is given in subsection *A*. A short description of hierarchical representation is given in subsection *B*. The learning and specification methods are described in subsection *C* and *D* respectively. Finally, we discuss the computation cost of the proposed method in subsection *E*.

A. Image Features

For object categorization, we consider locally defined low-level image features. For the recognition of a partially occluded object in cluttered environment, local image features are preferable. These features are generally invariant to different kinds of geometric and photometric transformations as well.

Features are generally extracted from affine covariant interest regions (image patches) by computing different kinds of descriptors of such an image patch. They are represented as a vector of *m* dimensional features space $\Re^m$. An up-to-date review of different methods of local feature detection and description can be found in [1]. All these features are indeed quite successfully used for many applications such as wide baseline matching for stereo pairs [13], content based image and video retrieval (CBIR) from large databases [14], model based object recognition [15], etc. However, it is generally believed that the local-feature-based nearest-neighbor classification method can overcome the intra-class variation of objects to some extent.

Local features are utilized in two different ways in different applications: (i) unquantized features, which comprises two basic steps: feature extraction and feature matching; (ii) quantized bag-of-features and hyperfeatures, which includes the following four steps: feature extraction, feature clustering, frequency histogram construction, and feature matching. Matching approaches depend on distance matrices. For unquantized features, Euclidean distance and Mahalanobish distance are generally preferred. On the other hand Chi-squared (χ^2) distance and Earth Movers Distance (EMD) are popular for histogram comparison [11].

B. Hierarchical Representation

As discussed in section II, three approaches can be identified in the literature to obtain a hierarchy for a given set of object classes: handcrafted, statistical, and learning-based. The final outcome depends on the similarity and dissimilarity among object classes. These similarities and dissimilarities are rather abstract ideas. However, given a measure of similarities and dissimilarities, we can obtain a hierarchy for a give dataset following any of these methods. The hierarchy could vary depending on the designer's perspective and application as well.

In this work, we intuitively group some of the apparently similar classes into a superclass. Two alternative hierarchies are depicted in Fig. 2 for ETH-80 [7] database. As shown in Fig. 2(a); dog, cow, and horse classes are put into a superclass named 'animal' which shares some common shapes. These superclasses always may not have common perceivable shape features rather some other common properties. For example, different types of artefacts may not have any shape similarity; instead they share a common properties of being man-made (e.g. car and cup). Similarly 'small' and 'big' (Fig. 2(b) superclasses have vague shape similarities. These super-classes can be further clustered into the higher-level super-superclasses, and so on. Eventually, we obtain a multilevel hierarchy. For simplicity, we consider only a two-level hierarchy in this work.

A hierarchy can be represented by a general tree where each node can have a different number of children. The root (considered as the first level of decision making) corresponds to the world under consideration. The leaves are the basic classes and each of the internal nodes represents a superclass. Suppose, we have a set of *n* classes $\{C_i \mid i = 1\ldots n\}$, i.e. a tree with *n* leaves. We clusterize the set $\{C_i\}$ into *k* subsets (superclasses) $\{S_j \mid i = 1\ldots k\}$. Thus we form a subtree for each superclass where the root is represented by the superclass, and the basic classes under the superclass become leaves. We may further cluster superclasses into super-superclasses and add another level in the hierarchy and so on. Finally, we add the root which corresponds to the world (database) to form the tree.

C. Automatic Feature Selection

Suppose we have a categorization function (μ) and the input image *X* of an object belonging to class $y \in \{+1, -1\}$. If $\hat{y} = \mu(X)$ then for an ideal classification function we always expect that $|y - \hat{y}| = 0$. As discussed above, the function may have different choices of features. Let us assume we have *l* features $\{f_1, \ldots, f_l\}$ extractable for the function, and an arrangement of features $\alpha(\{f_1, \ldots, f_l\})$ is feasible. We wish to find an α which minimizes categorization error. Now the problem of feature selection is to automatically decide an arrangement α on a given training set such that

$$\arg\min_{\alpha} \sum | y - \hat{y} | \qquad (1)$$

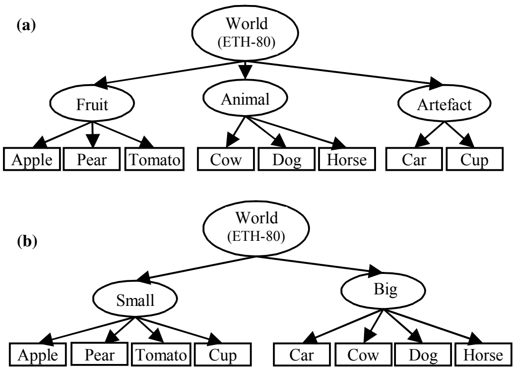


Fig. 2 Two different object class hierarchies for ETH-80 database. (a) HRC-1, (b) HRC-2

The arrangement α could be defined in many ways such as a subset of the features or a weighted contribution of all the features [12], [16].

In this work, we consider the use of SVM classification with multiple kernels as proposed in [12]. Each of the kernels uses a particular feature. The k-th kernel using feature f_k can be written as $K_k(\mathbf{x}, \mathbf{y})=\exp(-\gamma_k f_k(\mathbf{x}, \mathbf{y}))$, where the kernel matrices are strictly positive definite. The basic idea of multiple kernel method is to concatenate the contribution of the features by summing up the individual kernels. The contribution of each of the features is weighted as follows:

$$K = \sum_k d_k K_k \tag{2}$$

An optimal kernel K is learnt from a given training set by leaning the weight parameters d_k; $k = 1, \ldots, l$. The optimization is carried out in the SVM framework to achieve the best classification on the training set.

D. Learning a Hierarchy

Suppose we are given a hierarchy and a set of training images of each of the n basic classes. The hierarchical structure divides the learning problem into $O(n\text{-}1)$ independent sub-problems. We consider each of the internal nodes (superclasses) including root of the tree as a separate learning problem. Suppose S is such a node having c children. For a child $C_i \in \text{Children}(S)$ we obtain the training set T_i by concatenating the training samples of all basic classes under this branch. Now it becomes a c class problem with T_i training images for each, where $i = 1, \ldots, c$. If we consider the one-versus-rest method, we need to train c SVMs. The learning consists of the following two steps

1. Feature learning: in this step the weights d_k, $k = 1, \ldots, l$ for the features are learned from the training samples. The features having significant contribution are retrained and associated with the node.
2. Function learning: in this step the parameters of c SVMs are learnt and associate them with the node.

The learning process of a hierarchy is summarized in the following Algorithm 1. Here, the input is the hierarchy (H) represented as a tree and the set of training features (F) for all basic classes. The algorithm returns the tree with associated features and functions for each of the internal nodes.

E. Specification

In this step, we obtain a detail specification of a given test object. We start from the root of the hierarchy tree. From the learning process we already know the relevant features for this node and we extract only them for the input object. Then, the associated learned function is used to decide the subclass (the next level super-class in the hierarchy). We repeat the process until we reach a leaf (the basic class). Hence, the method follows a particular branch of height $O(\log_2 n)$ of the tree (i.e. goes through $O(\log_2 n)$ decision processes) yielding a specification with $O(\log_2 n)$ labels.

The specialization method is described by Algorithm 2. Here we supply the hierarchy associated with learned features (H') and classification functions (H'') together with a test object X. The algorithm returns a list of labels denoted by the set S.

Algorithm 1:

```
Hierarchy_Learning ($H$, $F$)
    Initilalize($Q$)          // initializes an empty queue
    $H' = H'' = H$;           // initialization
    $T$ = Root($H$)           // function Root returns the root of tree $H$
    EnQ($Q$, $T$)             // function EnQ enqueues $T$ into the queue
    while $Q \neq$ Empty
        $T$ = DeQ($Q$)     // function DeQ returns top element in the queue
        if Is_Leaf($T$) = false  // Is_Leaf determines whether $T$ is a leaf
            for all $C_i \in$ children ($T$)    // for all children of the node $T$
                EnQ ($Q$, $C_i$)
                // union of all training features of leaves for the subtree
                $F_i$ = Find_all_Training_Features ($C_i$)
            $d_k$ = Learn_weight ($\{F_i\}$)  // learn weight $d_k$ for k = 1, …, $l$
            $\alpha = \{□\}$                 // initialize $\alpha$ (a set of selected features)
            for all $k$ = 1 to $l$
                if $d_k$ > Threshold   // user defined threshold
                $\alpha = \{\alpha \cup f_k\}$      // adding significant features in the set
            $H'$ = Associate ($\alpha$, $H'$, $T$)   // associate $\alpha$ with the node $T$
            $\mu$ = Learn_Function ($\alpha$, $\{F_i\}$) // learn classifier for the node $T$
            $H''$ = Associate ($\mu$, $H''$, $T$)  // associate $\mu$ with the node $T$
    return $H'$, $H''$.
```

Algorithm 2:

```
Specification ($H'$, $H''$, $X$)
        $T$ = Root($H'$)    // function Root returns the root of tree $H'$
        $S = \{□\}$       // initialize the empty set of labels
        while Is_Leaf($T$) == false
            // function Is_Leaf determines whether $T$ is a leaf or not.
            $\alpha$ = Get_Ftrs ($H'$, $T$)   // Get_Ftrs returns learned features at $T$
            $F$ = Extract_features ($\alpha$, $X$) //extracting relevant features
            $\mu$ = Get_Func ($H''$, $T$)  // Get_Func returns learned function
            $T = \mu(F)$        // decision at current node
            $S = S \cup$ Label ($T$)    // append the node label
        $S = S \cup$ Label ($T$)        // append the leaf label
        return $S$.
```

F. Computational Cost

Consider the one-vs-rests method used in a regular nonhierarchical categorization process we need to train n SVMs for a n class problem. Similarly, we need to evaluate n function for testing process. The hierarchical method does not decrease the computational efficiency for the learning process which still remains $O(n)$. However, the real computational cost in the train process is multiplied by a constant depending of the branching factor of the tree. One the other hand, the specification process become $O(\log_2 n)$ instead of $O(n)$ in a nonhierarchical categorization process. This is a desirable property in many cases of categorization problem where training is merely an off-line process but the specification is rather online.

IV. Experimental Setup

We have tested the hierarchical method and compared the performance with a regular nonhierarchical method of object categorization. We have chosen a multiple kernel learning (MKL) SVM framework with automatic features selection. An implementation of this method is available online (http://www.robots.ox.ac.uk/~vgg/software/MKL). It combines SIFT [17], *self-similarity* [18], and *geometric blur* [10] features with the multiple kernel learning of Varma and Ray [12] to obtain state-of-the art performance. SIFT and self-similarity features are quantized into Bag-of-Words and then spatial histograms are computed. Geometric Blur feature is used without any quantization. Here, seven RBF kernels are combined linearly. For matching, χ^2 distance is used for former two features and correlation-based distance measures [10] are used for the later feature. One-versus-rests SVMs are trained and classification results are obtained by assigning each image to the class that obtains the largest SVM discriminant score.

Our implementation of hierarchical categorization was based on this MKL implementation. Algorithms 1 and 2 were implemented on the top of this implementation with the same set of parameters for the SVMs. For the convenience of comparison, we used the same set of features and kernels. The basic difference was that the features are selected independently for each node of the hierarchy and it was non-uniform. We tested the method on ETH-80 database for the two hierarchies (HRC-1 and HRC-2) in Fig. 2 and compared the results with the nonhierarchical MKL method.

A. Database

The methods were tested on ETH-80 database [7]. In this database, there are eight classes of both biological and artificial objects. In each of the classes there are ten objects. Fig. 3 shows the eight classes with ten objects for each of them. Each object is represented by 41 views with uniform background spaced evenly over the upper viewing hemisphere.

For this database, the suggested test mode is leave-one-object-out crossvalidation. Here all the 79 objects (i.e. 79×41 images) are used as training set and test is carried out with the remaining one unknown object (i.e. 41 images). The results are averaged over all 80 possible test objects.

For more effective evaluation, we carried out experiments with different numbers of training objects. For three different sets of experiments we use two objects, five objects and eight objects of each class. The remaining objects are used as a test set for the respective experiment. For instance, with the eight training objects, for each class we use 8×41 images as the training set and 2×41 images as the test set. In all these experiments we carried out multiple tests for a particular number of training objects. These training objects were selected randomly but kept the same for all the corresponding experiments.

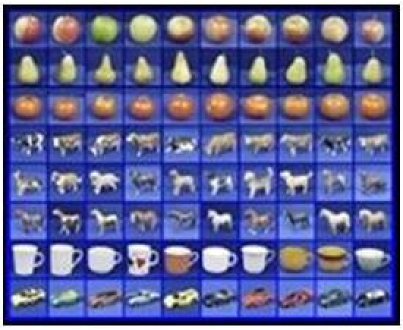


Fig. 3 Ten objects of each of the eight classes in ETH-80 database

B. Results

Fig. 4 shows the recognition rate for both of the methods. This was obtained by averaging the two tests for each experiment with different numbers of training objects. Here we carried out experiments with 2, 5, and 8 training objects. The performance of MKL method is compared with our hierarchical method using two hierarchies (HRC-1 and HRC-2). It can be seen that the hierarchical method has performed somewhat better than the regular MKL method in most of the cases. In these three experiments HRC-1 and HRC-2 are in average superior by 1.0% and 0.8%, respectively, than the nonhierarchical MKL method.

In Fig. 4 the comparison is performed on the basic classes. We have also performed some analysis on the higher levels of hierarchy (only the level just above the leave exists in the given hierarchies). For example, in HRC-1 in Fig. 2(a) there are three superclasses (fruits, animals, artifacts). We compared the recognition error at this level with MKL method. For the nonhierarchical MKL method we obtained equivalent results just by concatenating (bottom-up) the results on basic classes. Here a misclassification within the same superclass is not counted. For example, if an apple is classified as tomato, it is not considered as a misclassification.

Fig. 5 shows the error rate for the second level of hierarchy. Here we need to put the results in separate graphs due to the difference in superclasses in second level. For example, the superclass fruit has three basic classes but the superclass small has four. Thus, to make the result comparable with MKL method, we concatenate the results of the respective three and four classes. It can be seen that hierarchical methods results less error in the higher level of hierarchy. It means that if we wish to retrieve objects of a superclass (e.g. fruits) we will achieve better accuracy. In these three experiments HRC-1 and HRC-2 have average error rates lower by 2.7% and 2.3%, respectively, compared to a nonhierarchical MKL method.

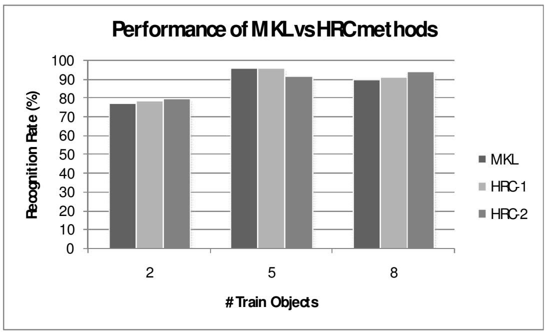
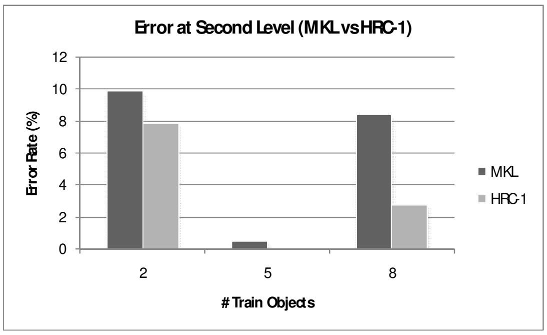
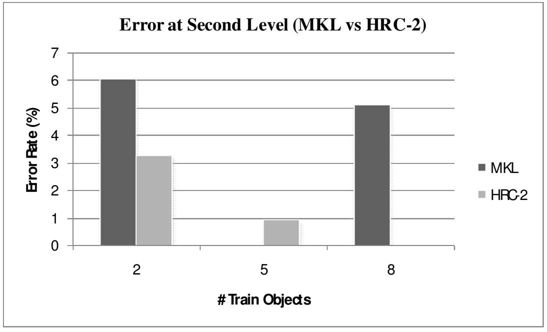


Fig. 4 Comparison of recognition rates between MKL and hierarchical methods



(a)



(b)

Fig. 5 Comparison of error rate between MKL and hierarchical method at second level

V. Conclusion

This paper describes a method of automatic feature selection for hierarchical object recognition. Automatic feature selection and hierarchical object categorization techniques have evolved recently. We are motivated by biological vision systems to put them together to improve recognition performance. In this work, we have tested our proposed method on two hierarchies, given for an ideal database. The hierarchies are based on natural similarities and properties of the basic object classes. We wanted to investigate feasibility of the proposed method. In fact, the experimental results are found consistent with our expectation. With the same parameters of a SVM based classification framework, we achieved better categorization performance without any increase in computational costs. This was true for both of the hierarchies. More importantly, we obtained less error in categorization on the superclass level. All these implicate the potentiality of this method especially in object categorization, semantic retrieval, automatic annotation, and other areas.

At the primary stage, we have tested our method on a small database. Here we assumed that the objects in all basic classes and superclasses are unimodal. This is an unrealistic assumption indeed. So trying with multimodal classification method at different levels of hierarchy should further improve the categorization performance. Furthermore, we have used the same classification model at different levels of hierarchy. A method which is capable to automatically select different models of categorization could further improve the results.

Acknowledgment

The research presented in this paper is a part of A*STAR Science \& Engineering Research Council grant 072 134 0052. The financial support of SERC is gratefully acknowledged.

References

[1] J. Li and N. M. Allinson, "A comprehensive review of current local features for computer vision," Neurocomputing, vol. 71, pp. 1771-1787, 2008.

[2] E. Rolls and G. Deco, Computational Neuroscience of Vision: Oxford University Press, 2002.

[3] S. Fidler and A. Leonardis, "Towards scalable representations of object categories: learning a hierarchy of parts," in IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, 2007, pp. 1-8.

[4] J. Sivic, B. C. Russell, A. Zisserman, W. T. Freeman, and A. A. Efros, "Unsupervised discovery of visual object class hierarchies," in IEEE Conference on Computer Vision and Pattern Recognition 2008, pp. 1-8.

[5] A. Zweig and D. Weinshall, "Exploiting object hierarchy: combining models from different category levels," in IEEE 11th International Conference on Computer Vision, Rio de Janeiro, Brazil, 2007, pp. 1-8.

[6] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Learning hierarchical models of scenes, objects, and parts," in The 10th IEEE International Conference on Computer Vision 2005, pp. 1331 - 1338.

[7] B. Leibe and B. Schiele, "Analyzing appearance and contour based methods for object categorization," in IEEE Conference on Computer Vision and Pattern Recognition Madison, USA, 2003, pp. 409-415.

[8] I. Autio, "Using natural class hierarchies in multi-class visual classification," Pattern Recognition, vol. 39, pp. 1290-1299, 2006.

[9] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Describing visual scenes using transformed objects and parts " International Journal of Computer Vision, vol. 77, pp. 291-330, 2008.

[10] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: discriminative nearest neighbor classification for visual category recognition," in IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 2126- 2136.

[11] J. Zhang and M. Marszalek, "Local features and kernels for classification of texture and object categories: a comprehensive study," International Journal of Computer Vision, vol. 73, pp. 213 - 238, 2006.

[12] M. Varma and D. Ray, "Learning the discriminative power-invariance trade-off," in IEEE International Conference on Computer Vision, 2007.

[13] T. Tuytelaars and L. V. Gool, "Matching widely separated views based on affine invariant regions," International Journal of Computer Vision, vol. 1, pp. 61-85, 2004.

[14] J. Sivic and A. Zisserman, "Video data mining using configurations of viewpoint invariant regions," in IEEE Conference on Computer Vision and Pattern Recognition, 2004, pp. 488-495.

[15] V. Ferrari, T. Tuytelaars, and L. V. Gool, "Simultaneous object recognition and segmentation by image exploration," in Eight European Conf. Computer Vision, 2004, pp. 40-45.

[16] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1 - 8.

[17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, vol. 60, pp. 91-110, 2004.

[18] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in IEEE Conference on Computer Vision and Pattern Recognition, 2007.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 53–58

ISBN 978-83-60810-27-9
ISSN 1896-7094

Selecting the best strategy in a software certification process

V. Babiy, R. Janicki, A. Wassyng
Department of Computing and Software
McMaster University
Hamilton, ON., Canada

A. D. Bogobowicz
Faculty of Mathematics
and Natural Sciences
University of Cardinal
Stefan Wyszynski
Warsaw, Poland

W. W. Koczkodaj
Department of Mathematics
and Computer Science
Laurentian University
Sudbury, ON., Canada

***Abstract*—In this paper, we propose the use of the pairwise comparisons (PC) method for selection of strategies for software certification. This method can also be used to rank alternative software certification strategies. The inconsistency analysis, provided by the PC method, improves the accuracy of the decision making. Some current methods of software certification are presented as they could be modified by the proposed method. Areas of potential future research are discussed in order to make the software certification process more feasible and acceptable to industry.**

I. Introduction

THE PROCESS of software certification is time consuming and hence expensive. Most software systems, since they are usually not critical, never go through a certification process. Thus software is still being developed without any consideration given to software certification [1]. In most modern systems, software is one of the most complex components. At the same time it is considered as one of the most error prone, despite the increasing demand for reliable software. As a result, there is an apparent need for a viable dynamic software certification process which can adjusted to the dynamic demands of the rapidly evolving software industry [2]. We propose to use the pairwise comparisons method as a means of selecting a strategy for a software certification process. It is quite likely that a certifying body would need to adjust their certification strategy for almost every project. This is because projects are different, especially when they are designed for different domains. A better software certification strategy may require a smaller amount of resources in order to adjust to new scenarios. The use of this approach will provide insight and understanding of the software certification process. For every project, some properties will be more important than others, while some properties will be completely irrelevant. The pairwise comparisons method can provide a scheme where the software certification strategy can be modified easily and adapted to different scenarios.

II. Product Based Certification

The objective of product based certification is to deduce whether the product conforms to requirements and provide an evaluation of the developers abilities to produce new products while conforming to requirements [3]. ISO IEC 14598 provides instructions on how to evaluate a software product. It uses the ISO IEC 9126 standard which describes how general attributes can be subdivided into less general attributes. In practice, both standards are applied in parallel. ISO IEC 14598 has four phases: defining the evaluation requirements, identifying the evaluation, building the evaluation schedule and executing the evaluation schedule. In defining the evaluation requirements, attributes and sub-attributes for the product evaluation are defined. These attributes and sub-attributes could be taken from the McCall's and Blundell's quality models [4], [5]. In identifying the evaluation, a collection of metrics are defined for the evaluation of all attributes and sub-attributes. In addition, metrics which will evaluate relationships between a product and its environment are also defined. While building the evaluation schedule, a detailed evaluation plan is constructed. Finally, the evaluation schedule is executed [6].

III. Process Based Certification

The IEC 61508 (Functional safety of electrical/electronic/programmable electronic safety-related systems) and DO-178B (Software Considerations in Airborne Systems and Equipment Certification) standards follow a process based methodology and thus implicitly view the software certification process as one that places the emphasis on checking that an approved process was adhered to. These standards describe the collection of practices which should be followed during software development. They claim that it would be easier to achieve validation and verification of software by following the proposed practices. The IEC 61508 and DO-178B standards should be followed in correlation with other regulations where they outline the significance of software failure. The Development Assurance Levels (DALs), from the domain of civil aerospace, are an example of this correlation which dictate levels of criticality. The automotive and European rail industries use Safety Integrity Levels (SILs). The (DALs) and (SILs) are not similar in their applications, despite the strong tendency for them both to focus on risk reduction. The more critical the software, the greater the need for risk reduction to be an essential attribute of the software. Greater demands upon the (SILs) and (DALs)

978-83-60810-27-9/09/$25.00 © 2010 IEEE

lead to stricter demands from the software development process. The DO-178B argues that the verification of a system should be accomplished through extensive testing, while highly emphasizing the need for a good traceability process and a manual review of the components [7]. The verification process supported by DO-178B is subdivided into four levels. There are twenty eight properties at the lowest level, D. They validate tools, high level requirements, and the configuration of the development process. The next level, C, deals with twenty nine properties. They validate low level requirements, testing and code coverage. The next level, B, deals only with eight properties and logic. The highest level, A, is responsible for sixty six attributes. At this level, while the overall quality of the product is evaluated, a significant focus is allocated to traceability [7].

IV. Motivation

In situations where it is difficult or infeasible to use an algorithm, we revert to the use of heuristics in order to find solutions. There are a large number of attributes which should be considered during the certification process. As projects evolve rapidly and grow in complexity we need mechanisms to assign consistent weights to attributes and properties. The pairwise comparisons method is ideal for this task because it can reduce inconsistencies while still maintaining some acceptable margin of error. We describe a process on how to assign consistent weights to attributes and properties. Once inconsistency is minimal, preferably not zero, the developed software certification strategy can be used as a dynamic entity in the software certification process [8], [9].

V. Pairwise Comparisons Method

The pairwise comparisons method was used for the first time in 1785 by Condorcet. He used this method in the election process where voters rank candidates based on their preference [10]. The method was a voting system which used matrices for particular pairwise comparisons with rows representing each candidate as a runner and columns representing each candidate as an opponent. It was Fechner who specified pairwise comparisons as a scientific method in 1860, although only from the psychometric perspective [11]. Thurstone, in 1927, provided a mathematical analysis of this method and called it the *law of comparative judgments* [12]. The *law of comparative judgments* can be used to scale a collection of attributes based on simple comparisons between attributes taken two at a time. Although, Thurstone referred to it as a law, it can be more appropriately identified as a measurement model which could be of important use for software certification. This model allows experts to synthesize diverse procedures involved in software certification. The hierarchy reduces the number of comparisons from $O(n^2)$ to approximately $O(n \ln n)$, making it applicable to a wide variety of problems. For example, a moderate case with 49 features would require 1,176 comparisons without a hierarchy and only 168 comparisons if these 49 features are arranged into a hierarchy by grouping seven features. Measurements of length, such as a meter or foot, or by mass and weight are commonly used and accepted. Society has become accustomed to having standards for the majority of tasks, and sometimes it is difficult to understand standards, which often occur in the software industry, without an acceptable universal measuring method. In the case of software certification, many strategies may need to be developed for a single project. It is safe to conclude that developing a single certification strategy is not feasible and would not work for all types of projects, because some projects have very little in common [9], [13], [14].

TABLE I: Comparison scale

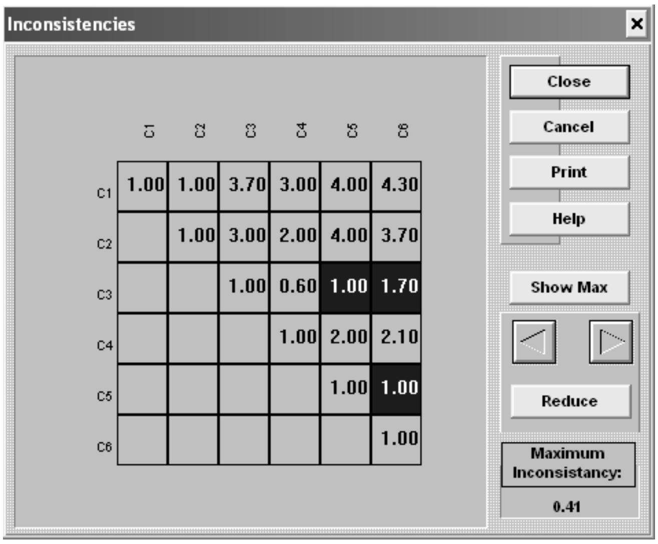
Code	Definition of intensity or importance
1	Equal or unknown importance
2	Weak importance of one over another
3	Moderate to essential importance
4	Demonstrated importance
5	Absolute importance
3.5 etc	Intermediate importance

A. Inconsistency Analysis

The pairwise comparisons method does not impose any limit on the number of criteria. Setting the maximum number of entities on one level to seven is accepted as a heuristic, because seven items gives 21 distinct pairs to compare. The first step in pairwise comparisons is to establish the relative preference of two criteria for situations in which it is impractical or irrelevant to provide absolute estimations. The relative comparison coefficients a_{ij} for criteria $C_1, C_2, \ldots, C_n$ are expected to satisfy $a_{ii} = 1$ and $a_{ij} = 1/a_{ji}$. The first constraint is related to comparing a given attribute with itself. The second constraint is a consequence of the obvious fact that $x/y = 1/(y/x)$ for $x, y \neq 0$. A scale from 1 to 5, as demonstrated in Table I, is used for expressing the importance of one attribute over others. This is accomplished in a pair. Other scales also exists, but as described by [8] larger values lose meaning in the comparison process.

The absolute estimation of the weights defining the importance of analyzed software certification criteria is practically unobtainable through either statistical or formal procedures. It would be beneficial to have experiments which may contribute to the accuracy of the estimates. However, it is unrealistic to expect such experiments to take place. This approach allows us to improve the processing of often subjective expert assessments in the certification process. We propose the use of a comparison scale that is demonstrated in Table I for the subjective expression of relative preference.

Reference	Criterion
C_1	Functionality
C_2	Reliability
C_3	Usability
C_4	Efficiency
C_5	Maintainability
C_6	Portability



	C1	C2	C3	C4	C5	C6
C1	1.00	1.00	3.70	3.00	4.00	4.30
C2		1.00	3.00	2.00	4.00	3.70
C3			1.00	0.60	1.00	1.70
C4				1.00	2.00	2.10
C5					1.00	1.00
C6						1.00

Maximum Inconsistancy: 0.41

Fig. 1: Relative importance of considered software quality attributes

The values of relative importance, which are given in Figure 1, have been entered by a single person solely for demonstration of the method. We have used the concluder software which can be download from the website that is maintained by W. W. Koczkodaj. [1] The values were deduced from the comparison in pairs. In a real scenario, the values should be reasoned about by a team of experts. The attributes have been taken from the ISO/IEC 9126 software standard. They are also known as the top six level attributes, which are considered to be key attributes for software quality [15]. Now we consider the process to identify the best alternative. Let us denote the attributes by $A_1, A_2, \ldots, A_n$ (n is the number of compared attributes), their actual weights by $\gamma_1, \gamma_2, \ldots, \gamma_n$, and the matrix of the ratios of all weights by $\Gamma = [\gamma_i/\gamma_j]$. The matrix of pairwise comparisons $\mathrm{M} = [a_{ij}]$ represents the assessments between individual pairs of alternatives ($\mathrm{M_i}$ versus $\mathrm{M_j}$, for all $i, j = 1, 2, ...n$) chosen usually from a given scale. The elements a_{ij} are considered to be estimates of the ratios γ_i/γ_j, where γ is the vector of actual weights of the attributes. All the ratios are positive and satisfy the reciprocity property $a_{ij} = 1/a_{ji},\ i, j = 1, 2, \ldots, n$. The inconsistency concept was explained in [16]. The distance based inconsistency indicator is defined as the maximum over all triads $\{a_{ik}, a_{kj}, a_{ij}\}$ of elements of M (with all indices i, j, k distinct) of their inconsistency indicators. It is defined as:

$$ii = \min\left(\left|1 - \frac{a_{ij}}{a_{ik}a_{kj}}\right|, \left|1 - \frac{a_{ik}a_{kj}}{a_{ij}}\right|\right) \tag{1}$$

Three is the minimal number of attributes which may cause inconsistency. Comparing two attributes will often lead to inaccuracy. The distance based inconsistency is the minimum distance from three ideal triads with no inconsistency when the third value is substituted using the consistency condition $a_{ij} \times a_{jk} = a_{ik}$. Since we are not in a position to determine which ratio is incorrect, all three assessments must be reconsidered before we attempt finding a consistent approximation for a given pairwise comparisons matrix. The stress on localizing the most inconsistent assessments is expressed by adding the *consistency-driven* to the name of the method since it is easier to remedy implications of an error when we are able to localize it. There is no practical reason to continue decreasing the inconsistency indicator to zero. Only the high values of the inconsistency indicator are considered as unacceptable and harmful. A very small value, or zero, may indicate a faked result rather than a true estimate. The practical challenge in working with the pairwise comparisons method comes from the lack of consistency of the pairwise comparisons matrices. Depending on the strategy it may take some time to get the matrix consistent [9], [13], [14], [17].

VI. Demonstration of the Matrix Adjustment

Assume the following attributes are considered for evaluation: safety, security, reliability, resilience, robustness, knowing, testability, adaptability, modularity, complexity, portability, usability, reusability, efficiency and learn-ability. They are considered in [18] as a general group of attributes of any software. All the attributes are subdivided into two main categories, such as development and maintenance. These groups are subdivided further and weights are assigned as demonstrated in Table II. It is safe to assume that some areas of software evolution are based on intuition and experience. In situations where there is more than just one person making decisions there is a greater possibility for inconsistency to occur. Industry must rely on the subjective judgments of experts in situations where practical methods of measure are unknown [9], [13].

TABLE II: On the left inconsistent strategy and on the right consistent strategy

Attribute	Percent	Attribute	Percent
development	**80%**	**development**	**80%**
efficiency	17.40%	efficiency	17.40%
resilience	8.66%	resilience	8.66%
robustness	4.96%	robustness	4.96%
adaptability	3.78%	adaptability	3.78%
modularity	46.23%	modularity	46.23%
complexity	**26.58%**	**complexity**	**19.19%**
portability	**14.06%**	**portability**	**17.29%**
reusability	**5.58%**	**reusability**	**9.74%**
reliability	16.37%	reliability	16.37%
knowing	9.14%	knowing	9.14%
safety	5.23%	safety	5.23%
security	2%	security	2%
maintenance	**20%**	**maintenance**	**20%**
usability	11.49%	usability	11.49%
learn-ability	4.95%	learn-ability	4.95%
testability	3.56%	testability	3.56%

[1] website to download concluder: http://www.cs. laurentian.ca/wkoczkodaj/

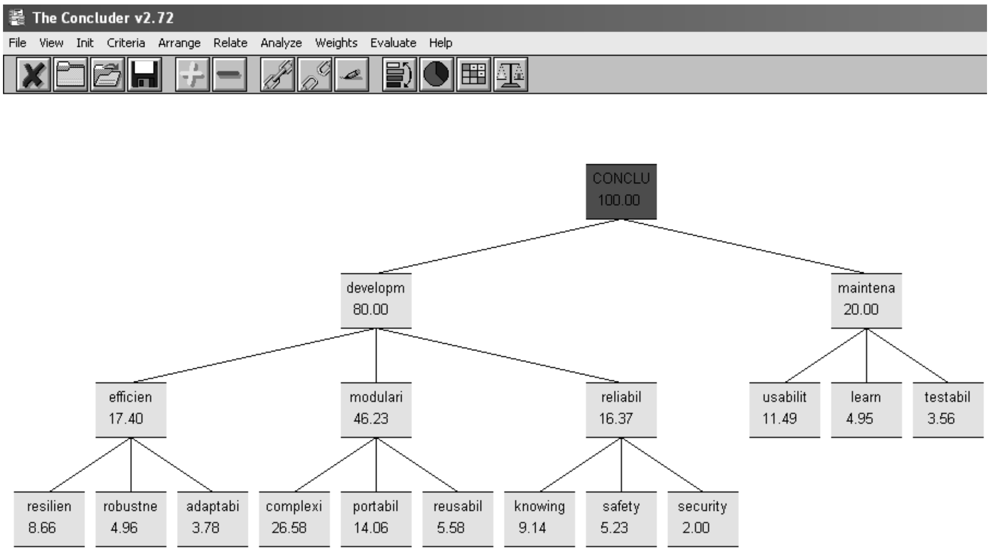


Fig. 2: Inconsistent strategy.

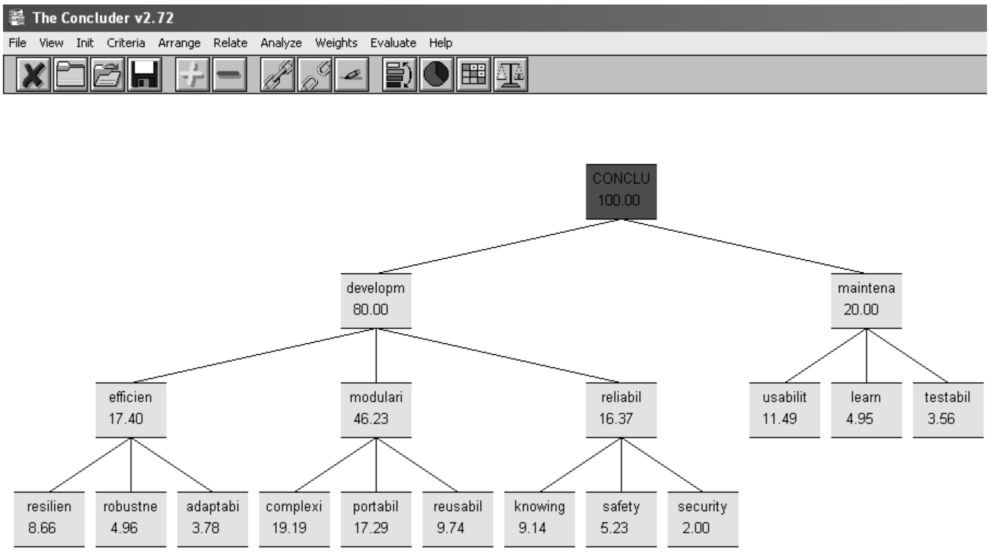


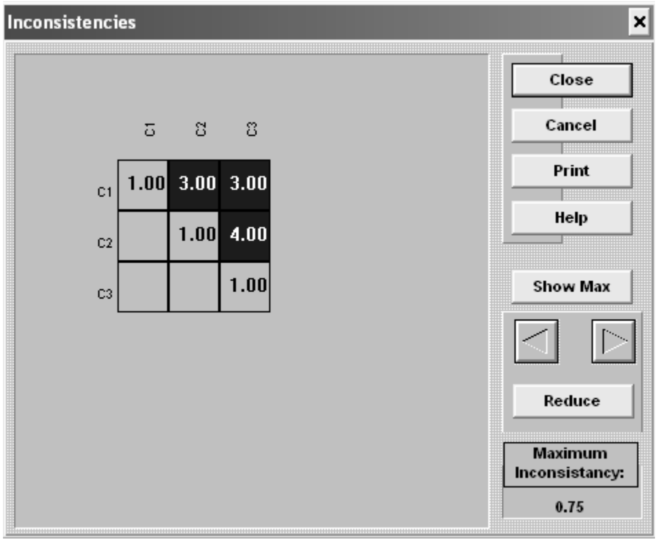
Fig. 3: Consistent strategy.

From Table II we can evaluate the complexity, portability and reusability triad where C_1 = complexity, C_2 = portability and C_3 = reusability. This triad has an inconsistency of 0.75 which is shown in Figure 4. As described in [8] it is not recommended. According to [8] the acceptable inconsistency is around 0.33. We have to adjust the values in order to bring the inconsistency down. After the adjustment, and as demonstrated in Figure 5, the inconsistency has decreased to 0.3 which is more acceptable.

In Table II, the weights of the attributes are allocated based on the significance of each attribute, and the most important criteria is complexity. After the correction we can see a new percentage redistribution, which is shown in Table II. The appropriate concluder's strategy models which demonstrate the percentage redistribution are given in Figure 2 and Figure 3. The redistribution could be evaluated and adjusted by many experts in order to achieve a situation in which the redistribution is accepted by all experts [19]. We think the consistency method is a preferred choice for the construction of consistent software certification strategy. The statistical evidence of the accuracy improvement with pairwise comparisons from approximately 15% to 5% for the one dimensional case (randomly generated bars) in [20], and from approximately 25% to 15% for randomly generated 2D shapes [21] support our expectations of improvement.

VII. CONCLUSIONS

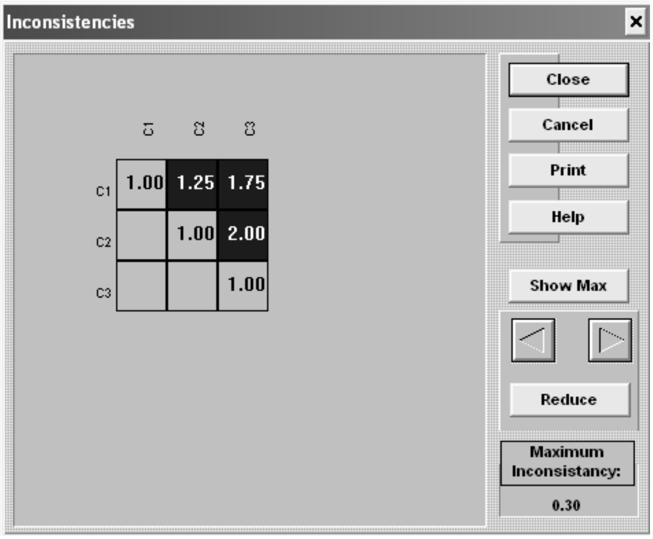
This study demonstrates how the use of pairwise comparisons to relate such intangible software attributes as resilience



	C1	C2	C3
C1	1.00	3.00	3.00
C2		1.00	4.00
C3			1.00

Maximum Inconsistancy: 0.75

Fig. 4: Inconsistency analysis for a group with three attributes. The inconsistency is 0.75.



	C1	C2	C3
C1	1.00	1.25	1.75
C2		1.00	2.00
C3			1.00

Maximum Inconsistancy: 0.30

Fig. 5: Inconsistency analysis for a group with three attributes. The inconsistency is 0.3.

or reusability can result in computing weights that may be useful in establishing a software certification strategy.

Software systems are developed for different purposes. Properties such as safety, reliability and modularity could have different priorities for different projects. Software providers are obliged to provide a guarantee that their software will operate reliably, but a certification process can not imply nor guaranty that the software will not fail in all unexpected situations [22].

The desire for companies to certify their software may be driven by their ability to increase sales and to maintain a competitive advantage in the industry. This comparative advantage could be achieved if companies would develop their products while conforming to the product's requirements and industry regulations [23], [24]. A more effective software certification strategy, in terms of better resource allocation, contributes to software development cost savings. We expect that certification methods which were utilized for past projects would fit, with some minor modifications, into our software certification strategy where we use the pairwise comparisons (PC) method. This may allow for a more accurate and consistent software certification process.

Acknowledgments

The authors would like to thank the Natural Science and Engineering Council of Canada (NSERC) for its partial support of this work. V. Babiy, R. Janicki and A. Wassyng also acknowledge the partial support of the Ontario Research Fund - Research Excellence.

References

[1] J. Oh, D. Park, B. Lee, J. Lee, E. Hong, and C. Wu, *Certification of Software Packages Using Hierarchical Classification*, 2004, vol. 3026.

[2] R. Moraes, J. Dures, E. Martins, and H. Madeira, *Component-Based Software Certification Based on Experimental Risk Assessment*. Springer Berlin / Heidelberg, 2007.

[3] J. Souter, "Process certification and product testing coming together," in *Software Quality Improvement Through Process Assessment, IEE Colloquium*, Mar 1992, pp. 5/1–5/6.

[4] N. E. Fenton and S. L. Pfleeger, *Software Metrics*, 2nd ed. 20 Park Plaza, Boston, MA: PWS Publishing Company, 1997.

[5] J. K. Blundell, M. L. Hines, and J. Stach, "The measurement of software design quality," *Annals of Software Engineering*, vol. 4, no. 1, pp. 235–255, 1997.

[6] K. Lee and S. J. Lee, "A quantitative evaluation model using the iso/iec 9126 quality model in the component based development process," *Computational Science and Its Applications*, pp. 917–926, 2006.

[7] T. P. Kelly, *Improvements in System Safety*. Springer London.

[8] W. W. Koczkodaj, "A new definition of consistency of pairwise comparisons," *Mathematical and computer modelling*, vol. 18, no. 7, pp. 79–84, 1993.

[9] W. W. Koczkodaj and W. O. Mackakey, "Mineral-positional assessment by consistency-driven pairwise comparisons," *Explor. Mining Geol.*, vol. 6, no. 1, pp. 23–33, 1997.

[10] M. J. A. N. Caritat, *Marquis de Condorcet: Essai sur l'Application de L'Analyse à la Probabilité des Décisions Rendues à la Pluraliste des Voix.* Imprimerie Royale., 1972.

[11] G. T. Fechner, *Elements of Psychophysics*. Rinehart and Winston, New York, 1965.

[12] L. L. Thurstone, "Law of comparative judgements," *Psychological Review*, vol. 34, pp. 273–286.

[13] W. W. Koczkodaj, M. Orlowski, L. Wallenius, and R. M. Wilson, "A note on using a consistency-driven approach to cd-rom selection," *Library software review*, vol. 16, no. 1, pp. 4–11, 1997.

[14] R. Janicki and W. W. Koczkodaj, "A weak order solution to a group ranking and consistency-driven pairwise comparisons," *Applied Mathematics and Computation*, vol. 94, no. 2-3, pp. 227–241, 1998.

[15] A. Rae, P. Robert, and H. Hans-Ludwig, *Software Evaluation for Certification*, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 1994.

[16] S. Bozóki and T. Rapcsák, "On Saaty's and Koczkodaj's inconsistencies of pairwise comparison matrices," *Journal of Global Optimization*, vol. 42, no. 2, pp. 157–175, 2007.

[17] R. Janicki, "Ranking with partial orders and pairwise comparisons," *Lecture Notes in Computer Science*, vol. 5009, pp. 442–451, 2008.

[18] I. Sommerville, *Software Engineering*, 8th ed. Edinburgh Gate, Harlow Essex, CM20 2JE, England: Pearson Education Limited, 2007.

[19] W. Hasselbring and R. Reussner, "Toward trustworthy software systems," *Computer*, vol. 39, no. 4, pp. 91–92, 2006.

[20] W. W. Koczkodaj, "Statistically accurate evidence of improved error rate by pairwise comparisons," *Percept Mot Skills*, vol. 82, pp. 43–48, 1996.

[21] P. Adamic, V. Babiy, R. Janicki, T. Kakiashvili, W. W. Koczkodaj, and R. Tadeusiewicz, "Pairwise comparisons and visual perceptions of equal area polygons," *Perceptual and Motor Skills*, vol. 108, no. 1, pp. 37–42, 2009.

[22] J. Voas and K. Miller, "Software certification services: Encouraging trust and reasonable expectations," *IT Professional*, vol. 8, no. 5, pp. 39–44, 2006.

[23] F. G. Keith and I. Vertinsky, "Antecedents to certification of software development processes," in *Standardization and Innovation in Information Technology, 2007. SIIT 2007. 5th International Conference*, Oct. 2007, pp. 81–90.

[24] P. Caliman, "Software product quality evaluation and certification: the qseal consortium methodology," *http://www.cse.dcu.ie/ essiscope/sm4/qseal.doc.*, Aug. 2009.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Extrapolation of Non-Deterministic Processes Based on Conditional Relations

Juliusz L. Kulikowski

M. Nalecz Institute of Biocybernetics and
Biomedical Engineering, PAS
4, Ks. Trojdena str.
02-109 Warsaw, Poland
Email: juliusz.kulikowski@ibib.waw.pl

***Abstract*—A problem of extrapolation of a large class of processes and of their future states forecasting based on their occurrence in the past is considered. Discrete-time discrete-value processes are presented as instances of relations subjected to the general relations algebra rules. The notions of relative relations, parametric relations and non-deterministic relations have been introduced. For extrapolated process states assessment relative credibility levels of process trajectories are used. The variants of direct one-step, indirect one-step and direct multi-step process extrapolation are described. The method is illustrated by numerical examples.**

I. Introduction

EXTRAPOLATION of processes is a basis of decision making in control, management, administration, social and/or economical planning, governing, etc. Roughly speaking, it consists in predicting future states of a process on the basis of its known current and past states as well as of some knowledge concerning influencing the process circumstances. Even extrapolation of deterministic processes, whose form can a priori be calculated using some analytical rules, may become a non-trivial problem. A process described by nonlinear differential equations may not only need high computational costs for being predicted but in certain cases it cannot be for long periods exactly predicted because of extreme equations' sensitivity to the initial conditions which with limited accuracy only can be established [1]. Extrapolation of non-deterministic processes is charged by inevitable uncertainty. In the simplest case of stochastic processes extrapolation the uncertainty takes the form of a statistical extrapolation error [2]. The rate of the error can be calculated when the probability distribution of the process is a priori known; otherwise, it should be experimentally evaluated [3, 4]. However, when non-probabilistic indeterminism of the processes takes place, the problem of based on the decision making, including extrapolation, becomes much more complicated [5]. On the other hand, namely non-deterministic processes whose future states are to be predicted in many application areas play an important role. The effectiveness of statistical methods of process extrapolation is satisfactory when the analyzed process's behavior depends on numerous unknown factors of approximately comparable strength. In a more general case of non-deterministic processes, some of the factors may be dominating, while their number and individual properties (e.g. nature, parameters, way of influence on the extrapolated process, etc.) remain unknown their existence by final effects only being manifested. Effective predicting of future states of such processes, on one hand, depends on making use of all available and relevant information, and on the other one, on neglecting all side information making the prediction incorrect. Till now, no commonly accepted definition of non-deterministic processes, excepting some of their particular cases, exists. In this paper an attempt to non-deterministic processes extrapolation based on extended theory of relations [6,7] is presented. Roughly speaking, it is assumed that the past, current and future states of an analyzed process as elements of an universe *U* are instances of a fuzzy multivariable relation; the problem of process extrapolation can thus be interpreted as the one of conditional relation identification when its instances are partially known. Moreover, some general concepts of relative logic (more widely presented in [9,10]) to an evaluation of the extrapolated states of the processes are used. The aim of this paper is to show that extrapolation of non-deterministic processes even if no numerical their instances' logical value is available is still possible. Such conclusion seems to be important in the case of extremely high indeterminism level of the considered processes.

The paper is organized as follows: preliminary extended relation theory concepts are described in Sec. II; in Sec III process extrapolation rules based on relations are presented. An example of process extrapolation under uncertainty is also given there. Conclusions and suggestions concerning future works are presented in Sec. IV.

This work was partially supported by the Ministry of Science and Higher Education of Poland.

II. Extended Fuzzy Relations

It will be considered a countable and linearly ordered family F of mutually isomorphic sets $U^{(i)}$, $i\in[\ldots-2,-1,0,1,2,\ldots]$. Within this paper the indices i will be interpreted as discrete: negative – past, null – current, positive – future time-instants. The family F will be called an universe, the sets $U^{(i)}$ – instant state repertories and their elements $\mathbf{u}^{(i)}_k$ – instant states. Any subfamily $F^{(m)}\subseteq F$, m = 1,2,3,…, preserving the linear order of F will be called its sub-universe; the family of all sub-universes of F (including F itself as well as empty subfamily $\varnothing$) will be denoted by Ω_F. For each sub-universe $F^{(m)}$ a Cartesian product of its instant state repertories:

$$C^{(m)} = \times_{(i)} U^{(i)},\quad U^{(i)}\in F^{(m)},\ F^{(m)}\in \Omega_F, \tag{1}$$

will be called a repertory of scenarios. Any subset

$$R^{(m)} \subseteq C^{(m)} \tag{2}$$

is a relation described in the sub-universe $F^{(m)}$ called below a process while its instances $\mathbf{y}\in R^{(m)}$ will be called process trajectories. A process is thus defined as a set of trajectories described in a given universe F or in any its sub-universe $F^{(m)}$. Any subset $R^* \subset R^{(m)}$ is called a partial process (a partial relation) of $R^{(m)}$ while any projection $R^{(p)} = R^{(m)}{}_{C(p)}$ of $R^{(m)}$ on a repertory of scenarios $C^{(p)}$ corresponding to a sub-universe $F^{(p)}\subset F^{(m)}$ is called a sub-process (a sub-relation) of $R^{(m)}$. A partial process consists thus of selected trajectories while a sub-process consists of cancelled (e.g. to a shortened time- interval) sub-trajectories of $R^{(m)}$. In the above-mentioned cases $R^{(m)}$ is called a broadening of R^* and an extension of $R^{(p)}$.

For a given sub-universe $F^{(m)}$ we denote by $\Omega^{(m)}$ the family of all processes, including an empty process $\theta^{(m)}$ and a "trivial" process $C^{(n)}$, that on $F^{(m)}$ can be defined. Similarly, it will be denoted by Ω_F the family of all processes that can be defined on any sub-families $F^{(m)}$ of the universe F. Then, an extended algebra of relations that can be described on F and its sub-universes is given by a quintuple [7]:

$$A_F = [F, \Omega_F, \underline{\cup}, \underline{\cap}, \underline{\neg},] \tag{3}$$

where $\underline{\cup}$, $\underline{\cap}$, $\underline{\neg}$, stand, respectively, for extended sum, intersection and negation of relations. Let us remark that an extended difference of relations $R^{(m)} \underline{\setminus} R^{(n)}$ is given by the formula:

$$R^{(m)} \underline{\setminus} R^{(n)} = R^{(m)} \underline{\cap} (\underline{\neg} R^{(n)}). \tag{4}$$

All concepts of the extended algebra of relations can thus directly be applied to the processes.

Example 1

Let I be a countable set of real integers i called time-instants and let U be a countable set of elements u_k, $k\in[\ldots,-2,\ -1,0,1,2,\ldots]$ called states. Then any function:

$$f_\nu:\ I \rightarrow U \tag{5}$$

$\nu\in N$, where N denotes a set of natural numbers, describes a discrete process trajectory (whose interpretation depends on a given application area). We take into consideration four time-instants: $i_1, i_2, i_3, i_4 \in I$ such that $i_1 < i_2 \le i_3 < i_4$ and two discrete time-intervals: $I', I''\subseteq I$, $I' = [i_1,\ldots,i_3]$, $I'' = [i_2,\ldots,i_4]$. Evidently, it is $I' \cap I'' \neq \varnothing$ where $\varnothing$ denotes an empty set. We shall denote by $f_\nu|_{I'}$ and $f_\nu|_{I''}$ the function f_ν cancelled, respectively, to the time-interval I' and I''. The above-described notions can also be interpreted in the relations theory terms. For this purpose an universe takes the form of linearly ordered family of sets $F = U^{\,I}$. Their Cartesian product C_F plays the role of a multidimensional discrete space whose elements (discrete vectors) correspond to the process trajectories. We also shall denote by $F' = U^{\,I'}$ and $F''=U^{\,I''}$ the sub-universes of F consisting of the sets marked by indices belonging, respectively, to the discrete time-intervals I' and I''; the corresponding Cartesian products (repertories of scenarios) will be denoted by C' and C''. Any process trajectory $\mathbf{y}\in C_F$ outlines thus in unique way in C' and C'' process sub-trajectories $\boldsymbol{y'}$ and $\boldsymbol{y''}$ called, respectively, the projections of $\mathbf{y}$ on the intervals I' and I'' as shown in Fig. 1 (the sub-trajectories are marked in gray, their common part is shown in black color). On the other hand, the process trajectory $\mathbf{y}$ will be called an extension of $\boldsymbol{y'}$ (respectively, of $\boldsymbol{y''}$) on the interval I.

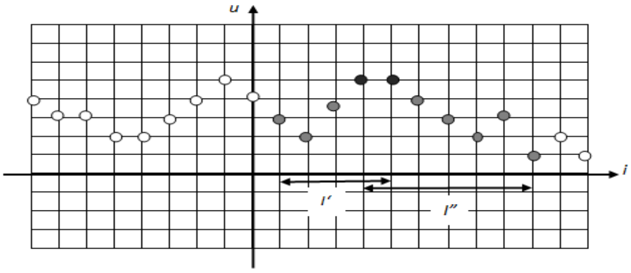


Fig. 1. Example of a discrete process trajectory **y** and its sub-trajectories ***y'*** and ***y''***.

Any set of discrete process trajectories $\mathbf{Y^*} = \{\mathbf{y}^{(\nu)}\}$, $\nu\in N$, described on a given interval $I^* \subseteq I$, is a subset of Cartesian product $\mathbf{Y^*} \subseteq C^*$; hence, it defines a relation in the family of sets F^*. On the other hand, each relation defined in F^* describes a discrete process whose trajectories constitute the instances of the relation. As it has been mentioned above, on the basis of the family Ω_F of all relations that can be defined on the sub-families of F it can be defined an extended algebra A_F of relations; it also becomes an algebra of discrete processes described in C_F.

Let us denote by $\boldsymbol{Y'}$ and $\boldsymbol{Y''}$ two discrete processes described, respectively, on the (not obviously disjoint) intervals I' and I''. Then:

a) an extended sum $\boldsymbol{Y'}\ \bar{\underline{\cup}}\ \boldsymbol{Y''}$ of the processes is a discrete process defined on the interval $I' \cup I''$ as a set of all trajectories $\mathbf{y}^{(\nu)}$ such that their projections on I' are the trajectories of Y' or their projections on I'' are the trajectories of Y'';

b) an extended product $Y' \underline{\cap}\ Y''$ of the processes is a

c) discrete process defined on the interval $I' \cup I''$ as a set of all trajectories $\mathbf{y}^{(\nu)}$ such that their projections on I' are the trajectories of Y' and their projections on I'' are the trajectories of Y'';

d) an extended difference $Y' \underline{\setminus}\ Y''$ of the processes is a discrete process defined on the interval $I' \cup I''$ as a set of all trajectories $\mathbf{y}^{(\nu)}$ such that their projections on I' are the trajectories of Y' and their projections on I'' are not the trajectories of Y''.

The following example will illustrate the above-given notions. Two discrete time-intervals are defined as follows (see Fig. 1):

$$I' = [1,2,3,4,5], \quad I'' = [4,5,6,7,8,9,10];$$

(the intervals are overlapping at the time-instants 4 and 5).

In these intervals two discrete processes are given by their trajectories:

$$Y' = \{[3,2,4,5,5], [2,3,4,5,4], [5,3,4,6,5]\},$$
$$Y'' = \{[3,2,4,5,5,6,4], [5,5,4,3,2,3,1], [6,5,0,3,4,5,5], [1,2,0,3,4,6,7]\}.$$

Their algebraic combinations will be described as processes in a joint time-interval:

$$\mathrm{I} = [1,2,3,4,5,6,7,8,9,10].$$

Then one obtains the following algebraic combinations of the processes:

$$Y' \underline{\cup}\ Y'' = \{[3,2,4,5,5,*,*,*,*,*], [2,3,4,5,4,*,*,*,*,*], [5,3,4,6,5,*,*,*,*,*], [*,*,*,3,2,4,5,5,6,4], [*,*,*,5,5,4,3,2,3,1], [*,*,*,6,5,0,3,4,5,5], [*,*,*,1,2,0,3,4,6,7];$$

where * denotes any element (instant state) of U;

$$Y' \underline{\cap}\ Y'' = \{[3,2,4,5,5,4,3,2,3,1], [5,3,4,6,5,0,3,4,5,5]\};$$
$$Y' \underline{\setminus}\ Y'' = \{[2,3,4,5,4,*,*,*,*,*]\},$$
$$Y'' \underline{\setminus}\ Y' = \{[*,*,*,3,2,4,5,5,6,4], [*,*,*,1,2,0,3,4,6,7]\}.$$

Negations of relations (complements of relations) can be defined as extended differences of trivial relations and the relations under consideration:

$$\underline{\neg} Y' \equiv \mathrm{C_F} \underline{\setminus}\ Y',$$
$$\underline{\neg} Y'' \equiv \mathrm{C_F} \underline{\setminus}\ Y'',$$

It can be remarked that the extended sum $Y' \underline{\cup}\ Y''$ represent bunches of trajectories satisfying Y' in the time-interval I' or Y'' in the time-interval I'' (or satisfying both of them), while the extended intersection $Y' \underline{\cap}\ Y''$ can geometrically be interpreted as a consistent prolongation of Y' by Y''. The intersection of relations defines also the following two sub-relations of Y' and $\mathbf{Y''}$ called their conditional relations:

$$Y'|Y'' = (Y' \underline{\cap}\ Y'')_{F'} \times \mathrm{C}_{F''\setminus F'} =$$
$$= \{[3,2,4,5,5,*,*,*,*,*], [5,3,4,6,5,*,*,*,*,*]\},$$
$$Y''|Y' = \mathrm{C}_{F'\setminus F''} \times (Y' \underline{\cap}\ Y'')_{F''} =$$
$$= \{[*,*,*,5,5,4,3,2,3,1], [*,*,*,6,5,0,3,4,5,5]\},$$

(lower indices at the Cartesian products and relations denote their projections on the given sub-families of sets) read, respectively, as: "Y' assuming that Y'' " and "Y'' assuming that Y' ".

A pair of relations Y' and Y'' is called mutually independent if $Y' \underline{\cap}\ Y'' \equiv Y' \times Y''$; in such case it is $(Y'|Y'')_{F'} \equiv Y'$ and $(Y''|Y')_{F''} \equiv Y''$ •

The conditional relations play a substantial role in process extrapolation. From a formal point of view all algebraic operations illustrated in Example 1 can also be interpreted as super-relations (relations between relations): a/ described on a composite Cartesian product $C' \times C'' \times \mathrm{C}_{F'' \cup F''}$ in the case of extended sum, intersection or differences and b/ on $C' \times \mathrm{C}_{F'' \cup F''}$ or on $C'' \times \mathrm{C}_{F'' \cup F''}$ in the case of complementary relations.

Till now, deterministic relations and their application to deterministic processes description were considered. A notion of parametric relation is a step to non-deterministic relations definition. Let $\mathrm{W}^{(m)}$ be a set of real parameter values. We take into consideration a Cartesian product $\mathrm{C}^{(m)}$ and a relation $\mathrm{R}^{(m)}$ as defined by (1) and (2). A Cartesian product

$$\mathrm{D}^{(m)} = \mathrm{C}^{(m)} \times \mathrm{W}^{(m)} \tag{6}$$

and a relation

$$\mathrm{P}^{(m)} \subseteq \mathrm{D}^{(m)} \tag{7}$$

such that: 1) the projection $\mathrm{P}^{(m)}|_{\mathrm{C}}{}^{(m)}$ of $\mathrm{P}^{(m)}$ on $\mathrm{C}^{(m)}$ is identical to $\mathrm{R}^{(m)}$, 2) to each instance of $\mathrm{R}^{(m)}$ exactly one value w, $\mathrm{w} \in \mathrm{W}^{(m)}$, is assigned. Apparently, if additional parameters are considered as other relations' variables, the nature of the relation is not changed and the extended relations algebra rules hold as well. However, this means that following from this algebra rigid rules of inheritance of parameter values on algebraic combinations of the trajectories are imposed. For the sake of practical utility parametric relation should be defined rather so that parameter transformation rules are defined independently on the algebraic operation rules. Parametric relations are then created by parameterization of some ordinary relations consisting in: 1st choosing the sets of parameters W, 2nd assigning parameter values to the instances of the relations, and 3rd establishing a set Q of functional relations:

$$\mathrm{q}_{\cup} \subseteq \mathrm{W}^{(m)} \times \mathrm{W}^{(n)} \times \mathrm{W}, \tag{8a}$$
$$\mathrm{q}_{\cap} \subseteq \mathrm{W}^{(m)} \times \mathrm{W}^{(n)} \times \mathrm{W}, \tag{8b}$$
$$\mathrm{q}_{\setminus} \subseteq \mathrm{W}^{(m)} \times \mathrm{W}^{(m)} \times \mathrm{W}, \tag{8c}$$

assigning in unique way the values of parameters, respectively, to the instances of extended sums, intersections, and differences of relations. The extended algebraic operations on parametric relations can then be defined as super-relations:

$$P^{(m)}\cup^* P^{(n)} = [(P^{(m)}\ \bar{\cup}\ P^{(n)})\ \bar{\cap}\ q_{\cup}]_{R(m)\times R(n)\times W}, \qquad (9a)$$

$$P^{(m)}\cap^* P^{(n)} = [(P^{(m)}\ \bar{\cap}\ P^{(n)})\ \bar{\cap}\ q_{\cap}]_{R(m)\times R(n)\times W}, \qquad (9b)$$

$$P^{(m)}\setminus^* P^{(n)} = [(P^{(m)}\ \bar{\setminus}\ P^{(n)})\ \bar{\cap}\ q_{\setminus}]_{R(m)\times R(n)\times W}, \qquad (9c)$$

which mean that 1st the algebraic operations should separately be performed on the component relations and on the assigned to them parameters, 2nd their results should be integrated by intersection of relations, and 3rd redundant parameter values from the super-relations should be removed by their projection on reduced sub-families of states. The form of the relations $q_{\cup}$, $q_{\cap}$ and $q_{\setminus}$ is not predetermined, it depends on the application purposes. If a family of parametric relations in the same universe F is considered, a family G_W of parameter sets instead of a single set W should be taken into consideration.

For a given universe F, its Cartesian product C_F, a family G_W of sets of parameters, a family Q of parameters transformations and a family Ω_D of all parametric relations reached by parameterization of the relations of Ω_F an algebra of parametric relations can be defined as an 8-tuple:

$$A_P = [F, C_F, G_W, Q, \Omega_{D}, \cup^*, \cap^*, \setminus^*]. \qquad (10)$$

Example 2

There will be considered the relations ***Y'*** and ***Y"*** described in Example 1. It is established a set of parameters W = [0,…,1] and the parameters are assigned to the trajectories as follows:

P' = {[3,2,4,5,5; 0.2)], [2,3,4,5,4; 0.6), [5,3,4,6,5; 0.3)]},

P"={[3,2,4,5,5,6,4; 0.3], [5,5,4,3,2,3,1; 0.1], [6,5,0,3,4,5,5; 0.5], [1,2,0,3,4,6,7; 0.4]} (parameters being separated from other components by semicolons). It is assumed that the parameters will be assigned to the sum and intersection of the relations in the below-described way.

If w_p, w_q ∈ W are parameter values assigned, respectively, to two given trajectories $\boldsymbol{u'}_p \in \boldsymbol{P'}$, $\boldsymbol{u''}_q \in \boldsymbol{P''}$ then:

1. if $\boldsymbol{u'}_p\ \bar{\cup}\ \boldsymbol{u''}_q$ satisfies ***P'*** only then parameter value w_p is to it assigned;
2. if $\boldsymbol{u'}_p\ \bar{\cup}\ \boldsymbol{u''}_q$ satisfies ***P"*** only then parameter value w_q is to it assigned;
3. if $\boldsymbol{u'}_p\ \bar{\cup}\ \boldsymbol{u''}_q$ satisfies both ***P'*** and ***P"*** then parameter value $\max(w_p,w_q)$ is to it assigned;
4. for $\boldsymbol{u'}_p\ \bar{\cap}\ \boldsymbol{u''}_q$ the parameter value $\min(w_p,w_q)$] is to it assigned;

According to this the following parameter values (weights) to the trajectories will be assigned:

P' ∪* ***P"*** = {[**3,2,4,5,5**,4,5,5,6,4; 0.2],
[**2,3,4,5,4,**4,5,5,6,4; 0.6], [**5,3,4,6,5**,4,5,5,6.4;0.3],
[**3,2,4,5,5,4,3,2,3,1**; 0.2)],
[**2,3,4,5,4,**4,3,2,3,1; 0.6], [**5,3,4,6,5**,4,3,2,3.1;0.3],
[**3,2,4,5,5**,0,3,4,5,5; 0.2)],
[**2,3,4,5,4,**0,3,4,5,5; 0.6],
[**3,2,4,5,5**,0,3,4,6,7; 0.2)],
[**2,3,4,5,4,**0,3,4,6,7; 0.6], [**5,3,4,6,5**,0,3,4,6.7;0.3],
[3,2,4,**3,2,4,5,5,6,4**; 0.3],
[2,3,4,**3,2,4,5,5,6,4**; 0.3], [5,3,4,**3,2,4,5,5,6,4**; 0.3],
[2,3,4,**5,5,4,3,2,3,1**; 0.1], [5,3,4,**5,5,4,3,2,3,1**; 0.1],
[3,2,4,**6,5,0,3,4,5,5**; 0.5],
[2,3,4,**6,5,0,3,4,5,5**; 0.5], [**5,3,4,6,5,0,3,4,5,5**; 0.5],
[3,2,4,**1,2,0,3,4,6,7**; 0.4],
[2,3,4,**1,2,0,3,4,6,7**; 0.4], [5,3,4,**1,2,0,3,4,6,7**; 0.4]};

P' ∩* ***P"*** = {[**3,2,4,5,5,4,3,2,3,1**;0.1],
[**5,3,4,6,5,0,3,4,5,5;** 0.3]};

¬****P'*** = (*C'* \ ***F'***) ×{0.4};

¬****P"*** = (*C"* \ ***F"***) ×{0.5},

where the sub-trajectories which contributed to the resulting parameter values have been indicated in bold.

Let us remark that in the above-given case the parameters are not additive; they may reflect relative “importance”, ”reliability” or other sort of “usefulness” of the trajectories. Moreover, to all trajectories of a negation (¬****P'*** or ¬****P"***) the same parameter values (0.4 or 0.5, respectively) have been assigned •

Example 2 shows a way to blurring (fuzzying) the relations: fuzzy relation is a parametric relation whose parameter values assigned to the instances reflect a level of credibility that the relation is by them satisfied. The expressions (8a-c, 9) hold thus for the fuzzy relations assuming that the sub-relations $q_{\cap}$, $q_{\cup}$, $q_{\neg}$ have been chosen adequately to instances' credibility description purposes. For this purpose they should satisfy some general triangular norms and co-norms conditions [9]. Such conditions are satisfied not only in the above- presented Example 2 but also in some other cases. Among them the following one deserves some attention.

Example 3

Once again, the relations ***Y"*** and ***Y"*** will be taken into consideration. However, instead of exact numerical credibility values their symbolic denotations to their instances will be assigned as follows:

P' = {[3,2,4,5,5; ε_1)], [2,3,4,5,4; ε_5), [5,3,4,6,5; ε_2)]},
P" ={[3,2,4,5,5,6,4; ε_2], [5,5,4,3,2,3,1; ε_0],
[6,5,0,3,4,5,5; ε_4], [1,2,0,3,4,6,7; ε_3]}

under additional assumptions:

$$0 < \varepsilon_0 < \varepsilon_1 < \varepsilon_2 < \varepsilon_3 < \varepsilon_4 < \varepsilon_5 < \varepsilon_6,$$

level ε_0 being assigned to all instances not belonging to ***P'*** or to ***P"***.

Then, according to the formerly presented rules 1 – 4, the following credibility values to the instances of algebraic combinations of relations will be assigned:

P' ∪* ***P"*** = {[**3,2,4,5,5**,4,5,5,6,4; ε_2],
[**2,3,4,5,4**,4,5,5,6,4; ε_6], [**5,3,4,6,5**,4,5,5,6.4; ε_3],
[**3,2,4,5,5,4,3,2,3,1**; ε_2)],
[**2,3,4,5,4**,4,3,2,3,1; ε_6], [**5,3,4,6,5**,4,3,2,3.1; ε_3],

[**3,2,4,5,5**,0,3,4,5,5; ε_2)],
[**2,3,4,5,4,**0,3,4,5,5; ε_6],
[**3,2,4,5,5**,0,3,4,6,7; ε_2)],
[**2,3,4,5,4,**0,3,4,6,7; ε_6], [**5,3,4,6,5**,0,3,4,6.7; ε_3],
[3,2,4,**3,2,4,5,5,6,4**; ε_3],
[2,3,4,**3,2,4,5,5,6,4**; ε_3], [5,3,4,**3,2,4,5,5,6,4**; ε_3],

[2,3,4,**5,5,4,3,2,3,1**; ε_1], [5,3,4,**5,5,4,3,2,3,1**; ε_1],
[3,2,4,**6,5,0,3,4,5,5**; ε_5],
[2,3,4,**6,5,0,3,4,5,5**; ε_5], [**5,3,4,6,5,0,3,4,5,5**; ε_5],
[3,2,4,**1,2,0,3,4,6,7**; ε_4],
[2,3,4,**1,2,0,3,4,6,7**; ε_4], [5,3,4,**1,2,0,3,4,6,7**; ε_4]};
$\boldsymbol{P'} \cap^* \boldsymbol{P''} = \{$[**3,2,4,5,5,4,3,2,3,1**; ε_2],
[**5,3,4,6,5,0,3,4,5,5;** ε_3]};
$\neg^*\boldsymbol{P'} = (C' \setminus \boldsymbol{F'}) \times \{\varepsilon_4\}$;
$\neg^*\boldsymbol{P''} = (C'' \setminus \boldsymbol{F''}) \times \{\varepsilon_5\}$ •

Definition. Fuzzy relations whose credibility levels are given in symbolic form with assumed inequalities instead of their exact numerical values are below called non-deterministic relations.

It is important to remark that in the non-deterministic relation case for finding the most credible algebraic combination of relation instances (e.g. their intersection) no exact numerical credibility values excepting assumed inequalities between them are needed.

III. Inference Rules Based on Non-Determined Relations

According to the definition of relative relations the following identities hold:

$$(\boldsymbol{F'}|\,\boldsymbol{F''})\ \overline{\cap}\ \boldsymbol{F''} \equiv \boldsymbol{F'}\ \overline{\cap}\ \boldsymbol{F''} \equiv (\boldsymbol{F''}|\boldsymbol{F'})\ \overline{\cap}\ \boldsymbol{F'}. \tag{11}$$

Assuming that $\boldsymbol{F'}$ and $\boldsymbol{F''}$ are two processes described, respectively, on discrete time-intervals $I' = [i_1,...,i_2]$, $I'' = [i_2,...,i_3]$ where the time-instants $i_1,\ldots,i_2-1$ belong to the "past", i_2 belongs to the (common for I' and I'') "present time" and $i_2+1,\ldots,i_3$ belong to the "future" it will be considered a problem of extrapolation of the process on I'' under the assumption that its behavior in the time-interval I'' has been observed. No special assumptions about the nature of the instant states of the process are made (the processes are thus not obviously numerical) excepting that the $U^{(i)}$ for all i are some isomorphic finite sets, $|U^{(i)}| = K$, K being a natural number. For credibility levels of the trajectories of $\boldsymbol{F'}$ and $\boldsymbol{F''}$ assessment it will be introduced a set of symbolic parameters $\mathbf{e} = [\varepsilon_0, \varepsilon_1,\ldots, \varepsilon_K]$. The components of $\mathbf{e}$ in K! ways can be linearly semi-ordered so that the component ε_0 remains at the first position. Moreover, for any fixed ordering each symbol (excepting the ε_0 one) can be preceded by = or < which gives 2^K different possibilities. Totally, it follows that there are $2^K\cdot$ K! different ways of possible relative credibility levels establishing in this given case.

Example 4

An example of semi-ordering of credibility levels can be presented in the form of a linear graph as shown in Fig. 2.

a/ $\varepsilon_0 = \varepsilon_4 < \varepsilon_5 < \varepsilon_7 < \varepsilon_6 < \varepsilon_{12} < \varepsilon_1 = \varepsilon_3 < \varepsilon_{10} < \varepsilon_2 = \varepsilon_9 = \varepsilon_{11} < \varepsilon_8$

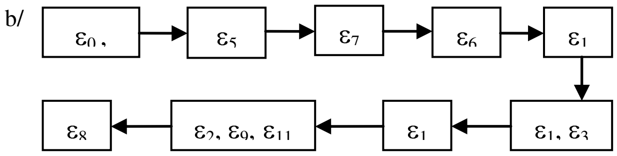


Fig. 2. Example of semi-ordering of credibility levels: a/ algebraic form, b/ graphical form.

For process extrapolation relative credibility levels of longer trajectories should be established [8]. Let us assume that it is given a non-deterministic relation **Y** describing a discrete process in the time-interval $I' \cup I''$. The credibility levels of its instances have been evaluated on the basis of past observation of the process. Therefore, **Y** can play the role of an experimental model of the process, suitable for process prediction decisions making (like training sets used in pattern recognition). For process extrapolation purposes **Y** will be presented in the form of an extended intersection of sub-relations:

$$\mathbf{Y} = \mathbf{Y}_{I'} \cap^* \mathbf{Y}_{I''} \equiv \boldsymbol{Y'} \cup^* \boldsymbol{Y''} \tag{12}$$

where $\boldsymbol{Y'}$ (a sub-process obtained by projection of $\mathbf{Y}$ on I') is considered as a model of the "past" and, similarly, $\boldsymbol{Y''}$ is used as a model of the "future". The credibility levels assigned to the instances of **Y** are without changing assigned to the corresponding instances of $\boldsymbol{Y'}$ and $\boldsymbol{Y''}$.

Similarly, conditional relations $\boldsymbol{Y'}\ |\boldsymbol{Y''}$ and $\boldsymbol{Y''}|\boldsymbol{Y'}$ can be calculated. The relation $\boldsymbol{Y''}|\boldsymbol{Y'}$ describing future trajectories of the process under the assumption that the past trajectories of the process are known is for process extrapolation of particular interest. Let $\mathbf{x} \in C'$ be a new past trajectory. Then, two cases should be considered:

1st it exists in $\boldsymbol{Y'}$ a trajectory $\mathbf{u}^{(\mathbf{x})}$ identical to $\mathbf{x}$:

$$\mathbf{u}^{(\mathbf{x})} \equiv \mathbf{x}; \tag{13}$$

2nd no such trajectory in $\boldsymbol{Y'}$ exists.

Direct one-step extrapolation. In the first case, an intersection:

$$\boldsymbol{Y''}(\mathbf{x}) = \mathbf{u}^{(\mathbf{x})} \cap^* (\boldsymbol{Y''}|\boldsymbol{Y'}) \tag{14}$$

consisting of all sub-trajectories in I'' being prolongations of **x** can be considered. This is illustrated by a graph shown in Fig. 4. The top-row of nodes represent the sub-trajectories belonging to $\boldsymbol{Y'}$ while the bottom-row corresponds to those belonging to their prolongations in the conditional relation $\boldsymbol{Y''}|\boldsymbol{Y'}$. A black node represents the condition $\mathbf{u}^{(\mathbf{x})}$ while grey nodes correspond to its possible prolongations. Moreover, to each sub-trajectory a credibility level is assigned. Therefore, a partial process consisting of the most credible sub-trajectories can be selected according to the rule:

$$\boldsymbol{z}^* = [\boldsymbol{u}^{(x)}, \boldsymbol{v}^{(p)}, u^*] \text{ where } (\boldsymbol{v}^{(p)}: e^{(p)} = max_{(q)}\{e^{(q)}\}, \boldsymbol{v}^{(q)} \in \boldsymbol{Y''}|\boldsymbol{u}^{(x)}, u^* \in \boldsymbol{v}^{(q)}) \tag{15}$$

$e^{(q)}$ being relative credibility levels assigned to the sub-trajectories $\boldsymbol{v}^{(q)}$; u^* denotes a final state of the most credible future sub-trajectory. Bold arrow in Fig. 3 represents the extension of the highest credibility level.

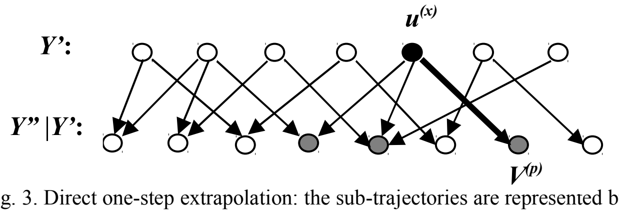


Fig. 3. Direct one-step extrapolation: the sub-trajectories are represented by the nodes.

Indirect one-step extrapolation. The 2nd case of a lack in $\boldsymbol{Y'}$ of an adequate past trajectory needs a more sophisticated approach. Solution of the problem on a general *case-based* approach consisting in finding a closely similar situation is possible. For this purpose a *relative similarity* $\rho(\boldsymbol{u}^{(i)}, \boldsymbol{u}^{(j)})$ of the pairs of elements of the sub-universe C' should be defined. Formally, it can be presented in equivalent form of a parametric relation:

$$\boldsymbol{\rho} \subseteq C' \times C' \times \boldsymbol{e} \tag{16}$$

where $\boldsymbol{e}$ is a finite set of symbols of relative credibility levels on which a semi-ordering relation σ has been imposed. In practice, the semi-ordering σ may be connected with a *similarity measure* of the pairs of sub-trajectories $(\boldsymbol{u}^{(i)}, \boldsymbol{u}^{(j)}) \in C' \times C'$.

Let $\boldsymbol{x} \in C'$ be a given past sub-trajectory. Then it can be defined a partial relation $\boldsymbol{S}_{x^*} \subset \boldsymbol{\rho}$ consisting of all instances containing the pairs $[\boldsymbol{x}, \boldsymbol{u}^{(j)}]$ and their corresponding credibility levels $e_{x,j}$. It will be taken into consideration an intersection:

$$\boldsymbol{Y'}_x = \boldsymbol{S}_{x^*} \cap^* \boldsymbol{Y'} \tag{17}$$

It evaluates the (on similarity measure based) credibility levels of the trajectories $\boldsymbol{u}^{(j)} \in \boldsymbol{Y'}$ used as models approximating the given past sub-trajectory $\boldsymbol{x}$. Then, an extrapolation of $\boldsymbol{x}$ can be reached like in the above described direct one-step extrapolation case when $\boldsymbol{Y'}$ is replaced by $\boldsymbol{Y'}_{\boldsymbol{x}}$. Analytically, the solution is given by the decision rule:

$$\begin{aligned} \boldsymbol{z}^* = [\boldsymbol{x}, \boldsymbol{u}^{(j)}, \boldsymbol{v}^{(p)}, u^*] \text{ where } (\boldsymbol{u}^{(j)}, \boldsymbol{v}^{(p)})&: e^{(j,p)} = \\ = max_{\{j,q\}} min(e_{x,j}, e_{j,q}), (\boldsymbol{x}, \boldsymbol{u}^{(j)}) &\in \boldsymbol{Y'}_{\boldsymbol{x}}, \\ \boldsymbol{v}^{(q)} \in \boldsymbol{Y''}|\boldsymbol{Y'}_x ; u^* \in \boldsymbol{v}^{(p)};\ e_{x,j}, e_{j,q} &\in \boldsymbol{e}. \end{aligned} \tag{18}$$

A scheme of this problem solution is shown in Fig. 4. It is important to remark that the most credible solution of the in-direct extrapolation problem due to the relative, non-numerical credibility levels has been found.

Direct multi-step extrapolation. In the one-step extrapolation an extrapolation range is by the length of sub-trajectories of $\boldsymbol{Y''}$ limited. However, it by a multi-step extrapolation procedure can be extended. For this purpose a sequence of equal-length overlapping time-intervals $I^{(m)}, \ldots, I^{(2)}, I^{(1)}$ will be considered, each time-interval being divided by a "present" time-instant into its "preceding" and "following" sub-interval, as shown in Fig. 5.

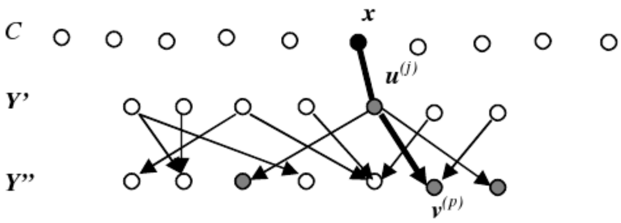


Fig.4. A scheme of in-direct one-step extrapolation problem solution.

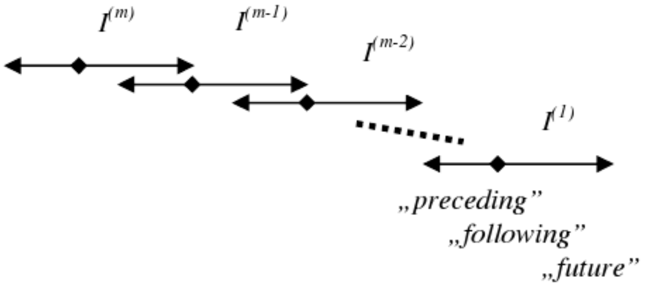


Fig. 5 Extension of process extrapolation intervals.

Let us assume that on the time-intervals the corresponding sub-processes $\boldsymbol{Y}^{(m)}, \boldsymbol{Y}^{(m-1)}, \ldots, \boldsymbol{Y}^{(1)}$ described by isomorphic relations have been defined. In each sub-process $\boldsymbol{Y}^{(\mu)}$ a "preceding" $\boldsymbol{Y}^{(\mu)}$' and "following" $\boldsymbol{Y}^{(\mu)}$'' part so that

$$\boldsymbol{Y}^{(\mu)} = \boldsymbol{Y}^{(\mu)'} \cap^* \boldsymbol{Y}^{(\mu)''} \tag{19}$$

will be distinguished. Then, an intersection:

$$\boldsymbol{Y} = \boldsymbol{Y}^{(m)} \cap^* \boldsymbol{Y}^{(m-1)} \cap^* \ldots \cap^* \boldsymbol{Y}^{(1)} \tag{20}$$

will be taken into consideration. In the process $\boldsymbol{Y}$ a sub-process $\boldsymbol{Y}^{(m)}$' as the "past" and $\boldsymbol{Y}|\boldsymbol{Y}^{(m)}$'as the "future" can be considered. Formally, solution of the multi-step extrapolation problem is similar to this of the one-step extrapolation one excepting that the "future" sub-universe becomes larger as containing more alternating sub-trajectories leading from a "present" to a "final" state. A solution of the direct multi--step extrapolation problem is given by the decision rule:

$$\begin{aligned} \boldsymbol{z}^* = [\boldsymbol{u}^{(x)}, \boldsymbol{v}^{*(p)}, u^*] \text{ where } (\boldsymbol{v}^{*(p)}: e^{(p)} = \\ = max_{(q)}\, min\{e^{(q1)}, e^{(q2)}, \ldots, e^{(qm)}\}, \boldsymbol{v}^{*(q)} \in \boldsymbol{Y''}|\boldsymbol{u}^{(x)}) \end{aligned} \tag{21}$$

Above, $\boldsymbol{v}^{*(q)}$ denotes the q-th path leading from $\boldsymbol{u}^{(x)}$, $\boldsymbol{u}^{(x)} \in \boldsymbol{Y}^{(m)}$', through a chain of connected sub-trajectories $\boldsymbol{v}^{(q1)}$, $\boldsymbol{v}^{(q2)}, \ldots, \boldsymbol{v}^{(qm)}$ to a final future state $u^* \in \boldsymbol{v}^{(qm)}$.

Example 5

Systolic pressures of a patient have been daily recorded for several weeks (a real medical record has been used as a basis of this example). The scale of pressure have been divided into intervals as follows:

D_1: 101 – 105 mmHg
D_2: 106 – 110 mmHg,
D_3: 111 – 115 mmHg,
…
D_{24}: 215 – 220 mmHg.

An observed discrete process trajectory after quantization took the form:

$D_5\ D_4\ D_5\ D_4\ D_4\ D_1\ D_6\ D_2\ D_6 D_6\ D_5\ D_7\ D_4\ D_4\ D_5\ D_6\ D_6\ D_5\ D_3\ D_6$ …

The trajectory has been used as an experimental model of the process. From the above-given sequence of data 16 sub-sequences have been selected as follows:

P_1: $D_5\ D_4\ D_5 \mid D_4\ D_4$,
P_2: $D_4\ D_5\ D_4 \mid D_4\ D_1$,
P_3: $D_5\ D_4\ D_4 \mid D_1\ D_6$,
…
P_{16}: $D_6\ D_6 D_5 \mid D_3 D_6$

where the marks | separate the (arbitrarily chosen) "past" and "future" parts of the sub-sequences. Then, there have been established formally admissible connections between the sub-sequences, like:

P_1: $D_5\ D_4\ D_5\ \mathbf{D_4\ D_4},$
P_4: $\mathbf{D_4\ D_4}\ D_1\ D_6\ D_2$
$P_1 - P_4$: $D_5\ D_4\ D_5\ \mathbf{D_4\ D_4}\ D_1\ D_6\ D_2.$

This led to a list of admissible connections shown below:

$\underline{P_1 - P_4}$, $P_1 - P_{13}$, $\underline{P_2 - P_5}$, $\underline{P_3 - P_6}$, $\underline{P_4 - P_7}$, $\underline{P_5 - P_8}$, $\underline{P_6 - P_9}$, $P_6 - P_{16}$, $\underline{P_7 - P_{10}}$,
$P_7 - P_{17}$, $\underline{P_8 - P_{11}}$, $\underline{P_9 - P_{12}}$, $P_{10} - P_4$, $\underline{P_{10} - P_{13}}$, $P_{11} - P_2$, $\underline{P_{11} - P_{14}}$, $\underline{P_{12} - P_{15}}$,
$P_{13} - P_9$, $\underline{P_{13} - P_{16}}$, $P_{14} - P_{10}$, $\underline{P_{14} - P_{17}}$, $\underline{P_{15} - P_{18}}$, $\underline{P_{16} - P_{19}}$, $\underline{P_{17} - P_{20}}$,
$P_{18} - P_4$, $P_{18} - P_{13}$, $P_{20} - P_{17}$…

To the connection their relative credibility levels e can be assigned according to the number of cases they have been really noticed. A symbolic credibility level e_1 is assigned to the connections that have been noticed and e_0 to those which only formally are possible. It is assumed that $e_0 < e_1$; the connections to which e_1 has been assigned are in the list underlined.

Let us assume that an initial sub-process $\boldsymbol{Y'} = [D_5\ D_4\ D_5\}$ has been observed. The question is: what are the most credible extrapolations of the process? Its extrapolations are:

1st step: $\underline{\boldsymbol{Y'} - P_1}= [D_5\ D_4\ D_5 \mid D_4\ D_4\}$ with credibility level e_1;

2nd step: $\underline{\boldsymbol{Y'} - P_1 - P_4} = [D_5\ D_4\ D_5 \mid D_4\ D_4\ D_1\ D_6\ D_2]$ with credibility level $e = \min(.e_1,e_1) = e_1$;

$$\underline{\boldsymbol{Y'} - P_1}–P_{13} = [D_5\ D_4\ D_5 \mid D_4\ D_4, D_5\ D_6\ \ D_6]$$

with credibility level $e = \min(.e_1,e_0) = e_0$.

In similar way, further process extrapolation can be reached •

IV. Conclusions

Human natural ability to forecast possible future events is based on experience and intuition rather than on formal logical inference. Early computer-aided forecasting systems tried to reach similar goals using formal tools like stochastic processes theory, logical inference methods, etc. However, it occurred that including less rigid, heuristic methods into forecasting procedures may also lead to their improvement. The case-based methods of processes extrapolation as well as relative credibility levels of their future states forecasting presented in this paper belong to this new approach to human thinking imitation. Presentation of a process as a relation makes consideration of not only numerical but also qualitative and multi-aspect processes possible. Credibility levels are considered as comparative, not obviously absolute numerical values. As a consequence, indication of the most credible future states of an extrapolated process is based on a simple comparative analysis instead of rigid numerical calculi. Evidently, some shortcomings with the above-presented approach, like: dependence of the extrapolation effectiveness on the representativeness of the past process observations as models of the future process behavior also exist. The methods of relative credibility levels of process instances assessment need thus deeper investigation which should be a task for future works to be undertaken..

Acknowledgment

The Author would like to thank Prof. Halina Kwasnicka and Dr Urszula Markowska-Kaczmar for their kind inspiration to write this paper as well as to the Reviewers for their substantial critical remarks.

References

[1] E. Ott, Chaos in Dynamical Systems. Cambridge University Press, 1993.

[2] T.W. Anderson, The Statistical Analysis of Time Series. John Wiley & Sons, New York, 1971.

[3] S. Rougegrez. Similarity evaluation between observed behaviours for the prediction of processes. Topics in Case-Based Reasoning. LNCS, vol. 837, Springer,1994, pp. 155-166.

[4] C. Brezinski, M. Redivo Zaglia, Extrapolation Methods. Theory and Practice. North-Holland, 1991.

[5] D. Cazorla, F. Cuartero, V. Valero, F. L. Pelayo, J. J. Pardo. Algebraic theory of probabilistic and nondeterministic processes. Journ. of Logic and Algebraic Programming. Vol. 55, No1-2, 2003, pp. 57-103.

[6] J.L. Kulikowski. *"Recognition of Composite Patterns".* Topics in Artificial Intelligence (A. Marzollo ed.). CISM Courses and Lectures No. 256. Springer Verlag, Wien, 1978, pp.169-223.

[7] J.L. Kulikowski, "Relational Approach to Structural Analysis of *Images". Machine Graphics & Vision,* vol. 1, No 1-2, 1992, pp. 299-309.

[8] J.L. Kulikowski. "Decision Making Based on Informational Variables". In: P. Grzegorzewski & al. (eds.), "Soft Methods in Probability, Statistics and Data Analysis". Physica Verlag, Heidelberg, 2002, pp. 310-320.

[9] E.P. Klement, R. Mesiar, E. Pap. Triangular Norms. Kluwer Academic Publishers, Netherlands, 2000.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Reasoning in RDFgraphic formal system with quantifiers

Alena Lukasová
Ostravská Univerzita v Ostravě,
Přírodovědecká fakulta, katedra informatiky a počítačů. 30. dubna
22, Ostrava, Czech Republic.
Email: alena.lukasova@osu.cz

Marek Vajgl
Ostravská Univerzita v Ostravě,
Přírodovědecká fakulta, katedra informatiky a počítačů. 30. dubna
22, Ostrava, Czech Republic.
Email: marek.vajgl@osu.cz

Martin Žáček
Ostravská Univerzita v Ostravě,
Přírodovědecká fakulta, katedra informatiky a počítačů. 30. dubna
22, Ostrava, Czech Republic.
Email: martin.zacek@osu.cz

***Abstract*—Both associative networks and RDF model (here we consider especially its graph version) belong to formal systems of knowledge representation based on concept-oriented paradigm. To treat properties of both of them as common properties of the systems is therefore natural. The article shows a possibility to use universal and existential quantified statements introduced prior to associative networks also within RDF graphic system and to define a RDF formal system with extended syntax and semantic that can use inference rules of associative networks. As an example solution, a logical puzzle is presented.**

I. Introduction

ABSTRACT RDF metamodel forms a framework and approach how to express a piece of knowledge by means of a directed, labeled graph that links relevant resources. All entities described by RDF expressions have to be treated as resources that are identified by resource identifiers - URIs. According to the metamodel RDF graph notation RDF statement consists of two labeled nodes (subject and object) linked together by a labeled edge (predicate). All of the labels of subject or predicate are represented by names as URIs, in the case of object it is also possible to use literal value. RDF in its second part RDFS also has an ability to define vocabularies (new terms) for practical use in RDF statements to specify kinds or classes of resources with specific attributes.

The graph view is the easiest possible and easy-to-understand visual explanations of RDF statements.

A similar idea as in the case of RDF occurred several years ago at the beginning of knowledge representation approach by means of association (semantic) networks. The authors of the idea tried to express semantics of knowledge about a concrete topic by a certain context framework represented graphically in the form of a binary labeled network (see for example [1]).

In relation to the first order logic associative networks as well as RDF graphs have been built on atomic vectors - elementary network statements, represented in the first order predicate logic by binary predicates:
⟨ predicate_symbol ⟩ (⟨attribute_1⟩, ⟨attribute_2⟩), or graphically:

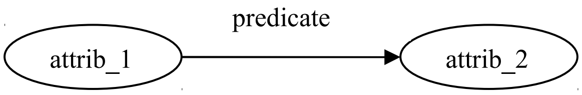


Fig.1 Associative network

In our paper [2], we have treated the RDF modeling as a special case of knowledge representation in associative networks (without functions). This fact brought a possibility to use notification and inference methods of associative networks also in the frame of RDF modelling. Moreover it has given a possibility to see RDF as a formal system.

The formal system of RDF(S) ought to be defined by its language, special axioms used to represent a knowledge base, and inference rules as a ground for creating special theories by means of formal proofs. The system ought to be presented as a formal system corresponding to predicate logic.

II. Extended Language Syntax of the RDF(S) Formal System

Our definition of RDF(S) language corresponds to the document W3C [3]. Authors of the document define names (node labels) as URIs or literals or *blank nodes* expressing the fact that there exists a URI reference making the statement of the triple true.

Graphs without blank nodes are ground RDF graphs.

According to [3], the RDF language does not contain a mechanism to represent universal or existential quantification. However, first order logic formulas can be transformed to special clausal form, where all individual variables (previously bounded to existential quantification) are eliminated by skolemization, and in this form (without quantification symbols) marked with special characters as existential terms. Universally quantified variables are (represented without quantification) conceived as variables of a universal character.

For the sake of expressing inference rules in a manner corresponding to the clausal form of the first order logic we add a metalanguage extention to the original RDF graphic language:

1. We extend the set of node names of the graphic RDF language by:

978-83-60810-27-9/09/$25.00 © 2010 IEEE

a. existential metasymbols (strings with the @ at the beginning) that give a possibility within a statement to express an existence of a name in subject or object label that satisfies the corresponding triple. (The original definition of the RDF language recommendation [3] do not suppose the necessity to distinguish existential symbols in different statements or different nodes and solves the problem universally by means of blank nodes only);

b. universal variable metasymbols (strings with a capital at the beginning) that give a possibility within a graph or its special part to express its universal properties.

2. We introduce a new kinds of RDF graphs that can express conditions "if -then". So we use within knowledge bases similarly as in association networks graphs the following kinds of networks:

   a. Unconditional networks:

   i. *ground* representing facts, containing purely constant node names, without variable or existential metasymbols

   ii. *universal/existential* with an occurrence of a universal/existential metasymbol

   b. *Conditions* (*rules*) - represent rules: q if p_1, p_2 ,...p_n with antecedent p_1, p_2 ,...p_n and consequent q. Edges in an antecedent part of the rule are represented via dash-line, consequent is represented via solid line, as usual. Conditions can have more than one atom in antecedent, but exactly one atom in consequent. Again, they can be divided into:

   i. ground

   ii. universal/existential

Network with a node label containing universal/existential metasymbol is a universal/existential network; otherwise it is taken as a ground network.

III. 3 Semantics of the Extended RDF(S) Language

Name in RDF(S) is a URI-reference or a literal (possibly ordered to a URI-reference). A set of used names of a language defines its vocabulary.

As the RDF graphs are described by RDF syntax triples, we have to consider an interpretation of the RDF language relative to a corresponding vocabulary, it means relative to a set of names (identifiers) - a set of node's and arcs' labels.

Resources corresponding to the URI-reference set of a vocabulary form a *universe of discourse* IR of an interpretation I.

To be able to use interpretation rules of a language, we need to have a *structure* of the interpretation including the universe of discourse IR that orders in the model theoretic semantics of a language:

- an object of the universe of discourse to each of the name in the model,
- a truth value to each of the statement of the model.

To define interpretation of our extended language RDF(S), we follow a two-part process corresponding to that one of syntax definition of the RDF(S) language:

- rdf-interpretation (rdfs-interpretation) concerning interpretation of the rdf vocabulary (rdfs vocabulary) that orders an object of the universe of discourse to each of the names in the model;
- rules of interpretation as methods how to derive truth values of basic statements (event. obtained from universal/existential statements) and consequently a truth value of the whole RDF graph.

Interpretation rules according to [6] (slightly modified and extended):

A simple interpretation ***I*** of a vocabulary **V** is defined by the structure of interpretation ***I*** as follows:

1. A non-empty set IR of resources, called the domain or universe of ***I***.
2. A set IP $\subseteq$ IR - the set of resources of IR corresponding to properties.
3. A mapping IEXT from IP into the powerset of IR x IR i.e. the set of sets of pairs <x,y> with x and y in IR.
4. A mapping IS from URI references in V into IR.
5. A mapping IL from typed literals in V into IR.
6. A distinguished subset LV of IR, called the set of literal values, which contains all the plain literals in V.

Points 4. - 6. concern interpretation of names. In this cases interpretation values are elements of the universe of discourse. The interpretation assigns to each URI reference its corresponding resource, to literals it assigns them.

The mapping of the point 3 concerns interpretation of a property (predicate) of a ground triple E:

If E is a ground triple (⟨subject⟩, ⟨predicate⟩, ⟨object⟩), then ***I***(E) = true if ⟨subject⟩, ⟨predicate⟩ and ⟨object⟩ are in V, ***I***(⟨predicate⟩) is in IP and the pair (***I***(⟨subject⟩), ***I***(⟨object⟩)) is in IEXT(***I***(⟨predicate⟩)), otherwise ***I***(E)= false.

It means in this case an interpretation I denotes a truth-value to a ground triple E: If E is a ground graph then ***I***(E) = false iff ***I***(E′) = false for some triple E′ in E, otherwise ***I***(E) = true.

Interpretation also has to specify the truth-value of a graph containing nodes with existential symbol labels.

If ***I*** is interpretation and ***A*** is a mapping from the set of nodes labeled by existential symbols to the universe IR of ***I*** then the interpretation [***I***+***A***](E) orders to every triple E containing an existential symbol label a ground triple via the mapping ***A*** and orders to the triple or to the whole graph a truth value as stated above.

Triples having universal metasymbols as node names give us a possibility to treat them as universal variables, to make valuations of the variables and then to order a truth value to the resulting ground triple (graph) as stated above.

Conditions representing rules q if p_1, p_2 ,...p_n hold universally.

Example 1 - representation of condition (with quantifiers)

“Cheetah belongs to order the Carnivore if it eats meat.”

We can rewrite the sentence as a universal conditional network. “Every animal belongs to the order of Carnivore if it eats meat.”

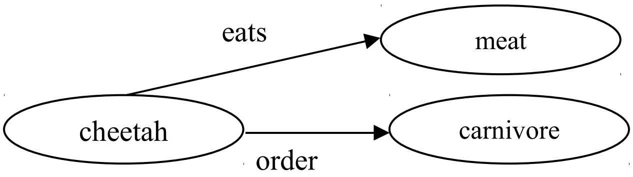


Fig.2 Unconditional network

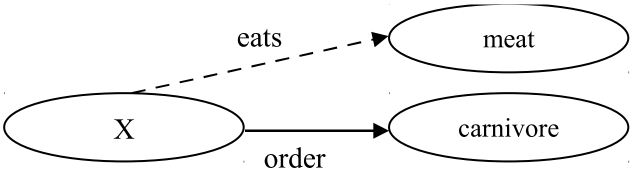


Fig.3 Conditional network

Using existential symbols we can express the sentence “Animal belongs to the order of Carnivore if there exists a meal that is meat and the animal eats it.”

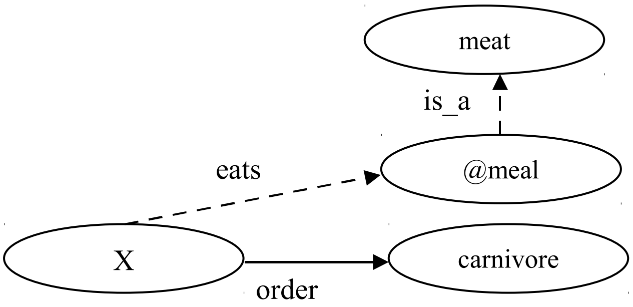


Fig.4 Extended network

IV. Inference in RDF

Deduction in RDF similarly to that in an associative network is a process of filling triples or terms in labels of the nodes from a *particular network* into a *main network* using one of the following rules:

Uniform substitution rule

a) If all the edges of a particular network also appear in the main network containing universal symbols as labels of their connected nodes, then the labels of nodes of the particular network can be substituted uniformly to the corresponding labels of nodes of the main network labeled by universal symbols.

b) If all the edges of a particular network also appear in the main network containing existential symbols as labels of their connected nodes and a maping *A* from the set of nodes labeled by existential symbols to the universe IR of ***I*** has been found then substitute uniformly corresponding elements of IR for existential symbols via the maping ***A***.

Transfer rule

If all dashed edges of the antecedent of a particular network appear (as solid) in the main network, and moreover if the labels of the corresponding nodes are the same then the solid vector of consequent can be added into the main network.

Negation of a triple as a special condition

To negate the triple E of RDF a special statement is used. It stands for the false consequent in the implication with a true antecedent. Due to no existence of a definition for a contradiction triple, a new special symbol of network has been created, called falsum, (notation ⊗), which is false in all interpretations.

Conditions, which consequent is falsum, are called negative facts.

Figure no. 8 represents negative fact „It is not true that dragonfly lives in Iceland. “

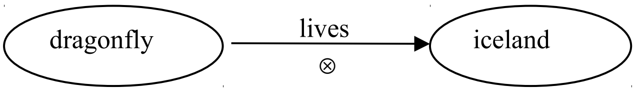


Fig.5 Negated associative network

Example 2 (logical puzzle)

We have 4 animals that live somewhere else and eat something else. Moreover:

1. The cheetah lives in Africa.
2. An animal that eats the flying insect lives in Europe.
3. The falcon doesn’t live in the Atlantic.
4. The dragonfly doesn’t hunt small fish.
5. The cheetah eats the mostly mammals.
6. The swordfish doesn’t live in Europe.
7. The falcon doesn’t eat the flying insects.
8. The dragonfly doesn’t live in the Iceland.
9. An animal that lives in the Atlantic doesn’t eat little birds.

Where does the falcon live and what does it eat?

Universe of discourse IR consists of all the resources used within solving the puzzle. Form of a list of pairs (graph label, URI of corresponding resource) is shown in tab 1.

Tab.1
URIs

mammal	http://dbpedia.org/resource/Mammal
bird	http://dbpedia.org/resource/Bird
fish	http://dbpedia.org/resource/Fish
insect	http://dbpedia.org/resource/Insect
cheetah	http://dbpedia.org/resource/Cheetah
falcon	http://dbpedia.org/resource/Peregrine_Falcon
swordfish	http://dbpedia.org/resource/Swordfish
dragonfly	http://dbpedia.org/resource/Dragonfly
africa	http://dbpedia.org/resource/Africa
iceland	http://dbpedia.org/resource/Iceland
atlantic	http://dbpedia.org/resource/Atlantic_Ocean
europe	http://dbpedia.org/resource/Europe
eat	http://en.wikipedia.org/wiki/Eat
live	http://en.wikipedia.org/wiki/Living

At the beginning we can imagine the main network graph consisting of four connected subgraphs with mutualy different existential node labels (fig. 5).

Next, we can define particular networks representing sentences 1 - 9 (as all the subgraphs are recognizable, the existential labels can be omitted) (fig 6).

Searching a common solution (model) of the main network and all of the nine participating particular networks consists in sequential binding of the label names occurring in particular networks (names of objects of the universe of discourse) to a set of existential nodes symbols (here blank nodes) in the main network until no contradiction is obtained and the proper mapping has been found.

Partially, rules 1. and 5. (similarly the case 4. and 8. and the case 3. and 7.) with the common cheetah label give a possibility to find a mapping within two subgraphs and make a corresponding substitution of labels (fig. 7, 8 and 9).

We can rewrite rules with negation (here for example 11 and 12) by RDF:bag. As an example, for a subgraph "dragonfly" holds a universal (within the puzzle) condition (fig. 10 and 11).

By applying these steps, we come to the conclusion, who lives where and what eats (fig. 12).

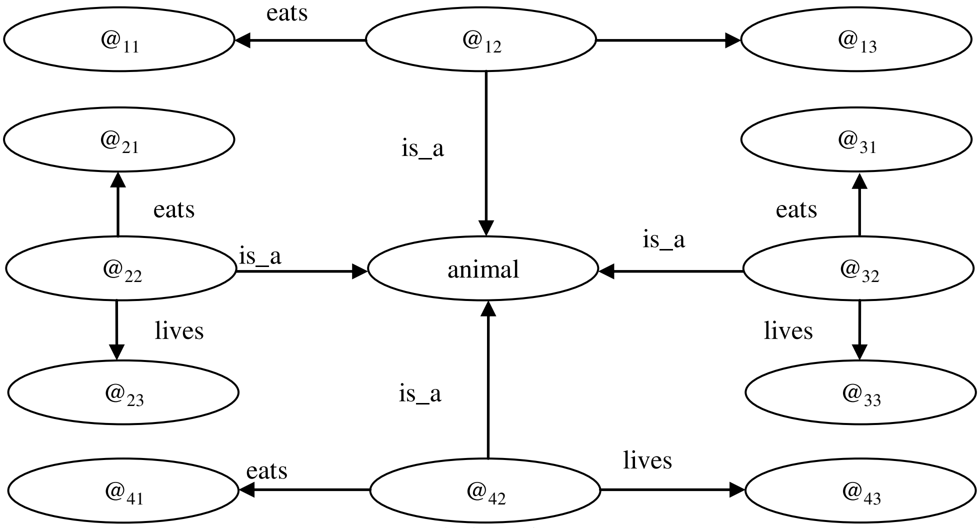


Fig.5 Base fact

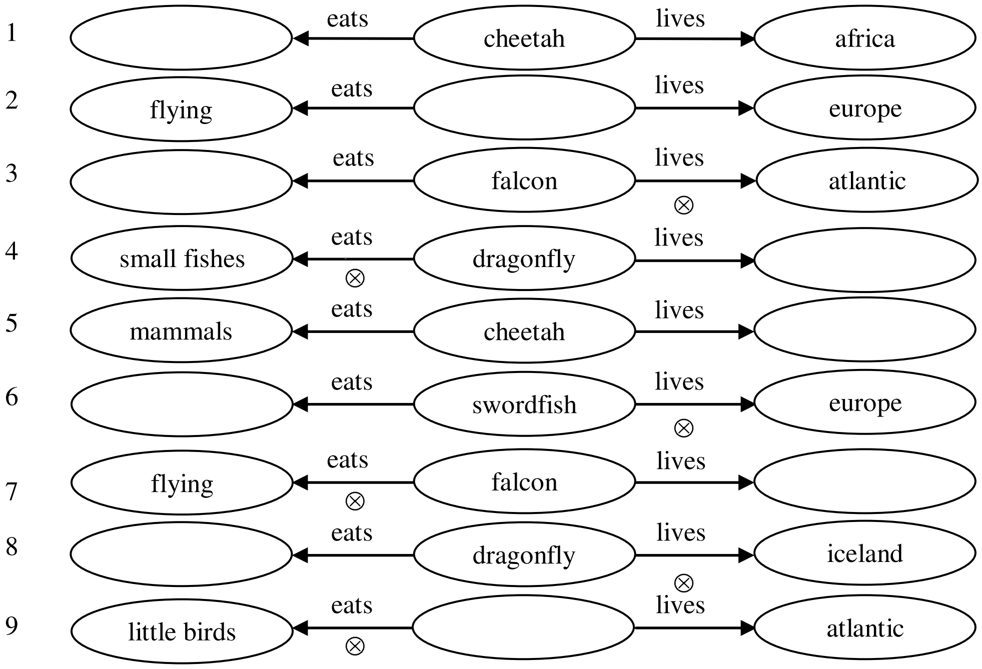


Fig.6 Extending facts

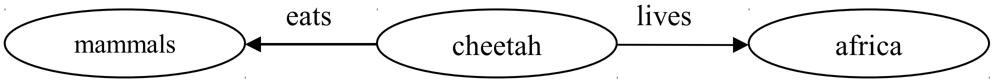


Fig.7 Rules over networks 1 and 5.

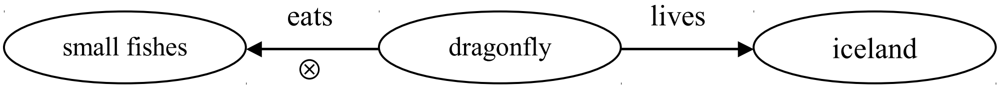


Fig.8 Rules over network 4 and 8.

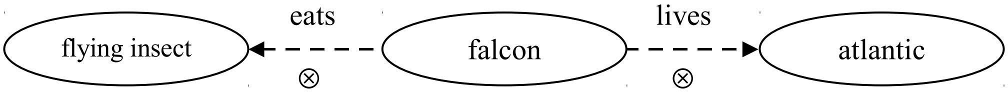


Fig.9 Rules over network 3 and 7.

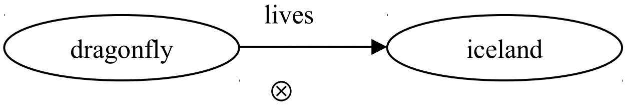


Fig.10 Negative facts

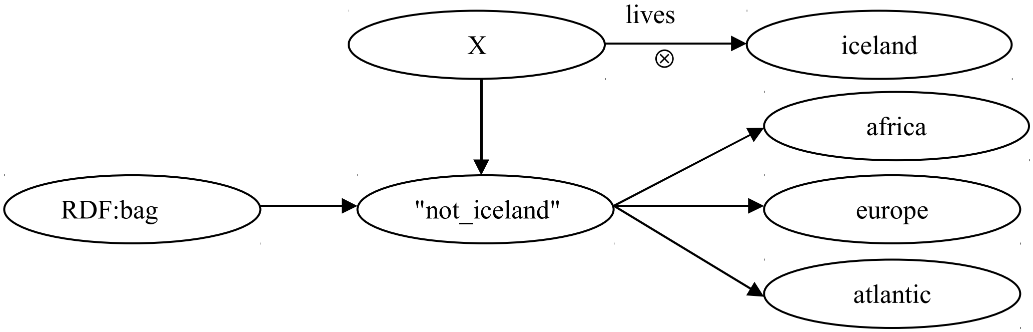


Fig.11 Negative facts

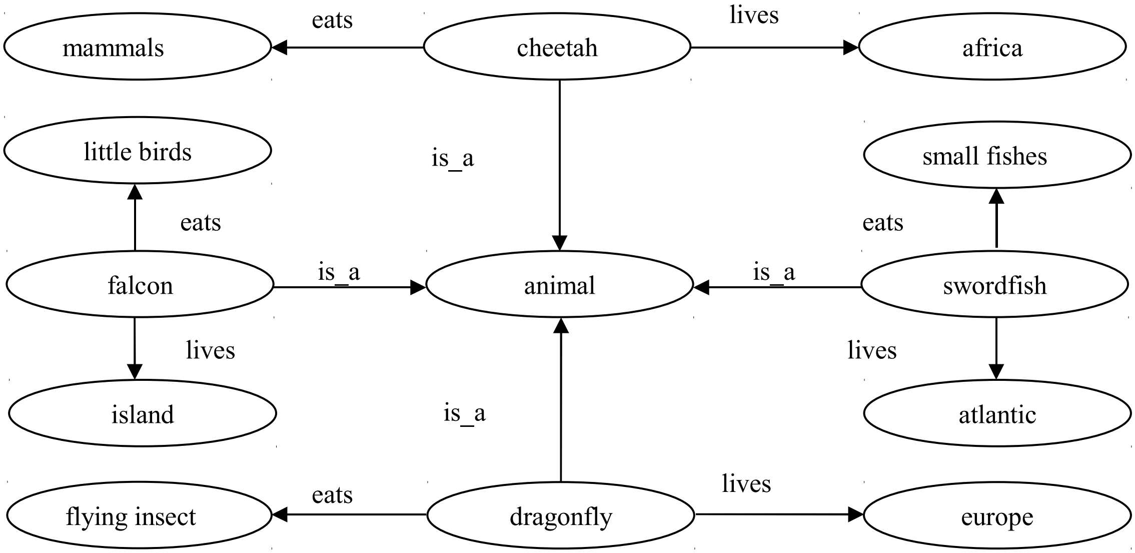


Fig.12 Extended network

V. Conclusion

We can treat the RDF modeling as a special case of knowledge representation in associative networks (without functions). This fact offers us a possibility to use the notification and inference rules of associative networks also in the frame of RDF modeling. Reasoning, possibly supported by the graphical version of representation, then becomes more understandable and easier to use than that of rewriting RDF models by description logic tools (OWL language) because of using their inference mechanisms. However, some issues like how negative and positive vectors are matched together are not fully formalized. Although article presents a mechanism (and an example) how can be semantic network used for inference over RDF models via graphical way affording better understandability, especially for human.

References

[1] Lukasová, A.: Reprezentace znalostí v asociativních sítích.Proc. Znalosti 2001 (in Czech).

[2] Vajgl, M., Lukasová, A.: RDF model as Associative Network. Datakon 2009, ISBN 978-80-245-1568-7.

[3] W3C: Resource description Framework (RDF). RDF Primer http://www.w3.org/TR/2004/REC-rdf-primer-20040210/

[4] W3C: RDF Vocabulary Description Language 1.0: RDF Schema, 2004 http://www.w3.org/TR/rdf-schema/

[5] W3C: Status for Resource Description Framework (RDF) Model and Syntax Specification, Recommendation 1999, http://w3.org/TR/REC-rdf-syntax.

[6] W3C: RDF Semantics. Recommendation 10 February 2004. http://www.w3.org/TR/2004/REC-rdf-mt-20040210/

Proceedings of the International Multiconference on Computer Science and Information Technology pp. 73–79

ISBN 978-83-60810-27-9
ISSN 1896-7094

Coevolutionary Algorithm For Rule Induction

Paweł B. Myszkowski
Wrocław University of Technology,
Wyb. Wyspiańskiego 27, 51-370 Wrocław, Poland
Email: pawel.myszkowski@pwr.wroc.pl

Abstract—**This paper describes our last research results in the field of evolutionary algorithms for rule extraction applied to classification (and image annotation). We focus on the data mining classification task and we propose evolutionary algorithm for rule extraction. Presented approach is based on binary classical genetic algorithm with representation of 'if-then' rules and we propose two specialized genetic operators. We want to show that some search space reduction techniques make possible to get solution comparable to others from literature. To present our method ability of discovering the set of rules with high F-score we tested our approach on four benchmark datasets and ImageCLEF competition dataset.**

Keywords: **data mining, rule extraction, evolutionary algorithms, image annotation**

I. Introduction

The size of datasets is growing constantly and as cannot be analysed it by human being so we use automating process, so-called Knowledge Discovery in Databases (KDD) which is a part of Machine Learning domain. The most interesting, from this paper point of view, is its one stage – data mining (DM). The data mining is an interdisciplinary field and its essence is knowledge acquisition from large amount of data. As our data might contain useful hidden and implicit the knowledge, the extracted knowledge can be successfully used in very important real-world domains such as image annotation.

As Evolutionary Algorithms (EA) are metaheuristics that search the solution space and can be easily applied for data mining tasks, such as clustering, prediction, classification and rule induction (paper [8] is a great survey of EA applications to KDD). Rule discovering (or rule induction) is most studied data mining task and its main goal is to build model of given data that describes it with possible best accuracy. Such model can be based on intuitive 'if-then' rules where: if-part (antecedent) attribute conditions and then-part (consequent) contain predicted class value (label). The classifier quality (accuracy of the gained model) can be tested on unseen data and measured by prediction error value.

EA is group of metaheuristics inspired by nature (Darwinian evolution theory) where the fittest individuals have better chance to survive and EA is widely used in data mining tasks (e.g. [2][3][5][15][20]). EA codes the problem solution as an individual and operates its features with aid of genetic operators (usually by mutation and crossover). Quality of individual is given as fitness function and its better value gives the higher probability of getting an offspring. In rule discovering task, the main motivation is to discover the rules with high predictive accuracy value. In literature we can find some approaches based on natural evolution that extends simple EA (e.g. classical genetic algorithm [9][12]) by some additional elements. For instance, commonly used if-then form of rule can be a single individual (Michigan approach) or included in ruleset form of individual (Pittsburgh approach). Mostly, there is used the Pittsburgh approach as it takes into consideration a rule interaction and is more natural and intuitive.

Interesting EA based method, so-called Genetic Programming (GP) can be found in [3], where individual is represented by logic tree that corresponds to logical expression such as rule: “If attrib$_0$>5 and (attrib$_3$=2,5 or atrrib$_2$<1,3) then class$_3$” to describe attributes' conditions of given class in chest pain diagnosis. The rule is presented as decision tree, where internal nodes are operators and leaf nodes represents attributes and corresponding condition values.

The multipopulation EA is based on the natural phenomena of coevolution (CoEvolutionary Algorithm, CoEA), where fitness function evaluation of few populations runs effectively in parallel way. Such method is proposed in [15], where individuals are selected from a few populations by cooperative selection pressure by choosing the best, previous fitness or classical selection method. There is used a collaboration pool size (from each population a group of the candidate individuals is selected) or a collaboration credit assignment (based on a selection of *n* individuals, where its optimistic, the average or pessimistic version is used). Another CoEA (presented in [20]) operates on populations that correspond to ruleset of *n* rules (where *n* is a parameter) linked by token competition as a form of the niching method. Paper [17] describes approach that takes into consideration distributed genetic algorithm for rule extraction task, where is presented positive influence of dynamic data partitioning distribution model to classifier accuracy.

There is also another strong trend in EA applications in DM – usage of specialized genetic operators. In Pittsburgh approach to classification task it is very important that individual representation consists of a complete classifier where rules cooperate and recombination operator causing its separation may make worse its classification accuracy. In [20] above problem is discussed and there is proposed a symbiotic combination operator which is a kind of heuristic that ana-

This work is partially financed form the Ministry of Science and Higher Education Republic of Poland resources in 2008-2010 years as a Poland-Singapore joint research project 65/N-SINGAPORE/2007/0.

lyzes results of changes in newly created individual. Another type of genetic operator specialization can be the usage of some hill-climbing algorithms to improve individuals (as candidate solution) by making “minor” modifications: if it causes fitter individual, given change is accepted. As a matter of fact, this causes the Baldwin effect [12]. Also, in evolution process ruleset can be modified in pruning procedure (e.g. [5]) - is optimized by removing unused/invalid attribute condition according to information gain measure value and/or examines results by some small changes in the ruleset. Also, we can find hybridization as a quite strong trend, where main motivation of such propositions is to build approach and link advantages of connected methods.

The remainder of this paper is organized as follows. Details of problem definition and our approach for rule discovering task is presented in section 2. Research methodology, used benchmark dataset and results of experiments are presented in section 3. Finally, section 4 presents conclusions and future research directions.

II. CAREX: Coevolutionary Algorithm for Rule EXtraction

Proposed method uses standard EA schema and starts a learning process with initial population (usually created randomly) and next individuals of current population are evaluated: each individual receives a fitness function value that corresponds to quality of proposition of given problem solution. In next step EA checks if stop conditions are not met: usually it is limit of generations and the best individual fitness value is acceptable (success). If stop criteria is not met EA runs the selection procedure that defines a seed of the new generation; next it is a communication between individuals (by crossover operator) and the independent trial (by mutation operator). The whole process repeats until some stopping condition is met. The crucial issue in evolutionary based method is the definition of individual representation schema, genetic operators (mutation and crossover) and evaluation function form, that give information about the individual fitness function value. In this section above elements are described.

A. Representation schema

In CAREX approach we decided to construct individual representation schema as simple as possible to get reduction of solution search space size. This is gained by coding of arguments value in binary representation (usually 2x8 bits per arguments) and only two logical operators: IN and NOT_IN. This methodology makes possible to use simple genetic algorithm, as we wanted to show that there is no need of EA extension to get solution comparable to others based on literature. Also we shown that there is possibility to build a specialized genetic operators.

We considered Michigan and Pittsburgh [12] approaches and finally CAREX system implements both, but in our research we use only the Pittsburgh model to keep all rules interactions in one individual. Therefore individual is represented as the set of rules (*ruleset*) that can assign instance to one class (if the task is to find completely one class description, so-called one-class-model) or to get model of all classes presented in dataset the individual consists of rules connected to class identifiers (all-class-model) as follows:

$$RuleSet := \{Rule_0, \ldots, Rule_n\}$$

Each rule is represented as the set of commonly used if-then type rules IF <*attributes_conditions*> THEN <*class_identifier*> as follows:

$$Rule_i := \ IF\ A_0 \text{ and} \ldots \text{and } A_n\ THEN \text{ class}_j$$

Where *class$_j$* symbol represents given class identifier, and $A_i, i \in \{0, \ldots, n\}$ represents a numeric range for a condition for *i*-th attribute as follows:

$$A_i := attribute_i\ operator\ (a,b)$$

where bellow *operator* can be:

$$\text{IN}(a,b) \rightarrow a < attribute_i < b \quad \text{or}$$

$$\text{NOT\_IN}(a,b) \rightarrow attribute_i < a \text{ or } attribute_i > b$$

where $a<b$ and $a,b \in \mathbb{R}$. Above representation is strictly based on conditions combination for selected attributes. We decided to use only two operators to keep individual representation as simple as possible. For instance, rule is:

$$IF\ attribute_0\ \text{IN}(0.1, 0.5)\ \text{and}$$
$$attribute_2\ \text{NOT\_IN}(1.0, 1.2)\ THEN\ class_2$$

Above rule describes all instances with values from range <0,1; 0,5> for attribute$_0$. Another condition takes into consideration attribute$_2$, where its value cannot be in range <1,0; 1,2>. Data described by conjunction of conditions are proposed to label with class$_2$.

In CAREX we decided to use binary vector representation thus we are allowed to use classical binary genetic operators to manipulate individual's particles. To avoid a drastic change of attribute value we use a Gray code. Also each attribute is extended by one enabled/disabled bit. Before CAREX starts, the dataset is preprocessed: instances are analyzed to recognize domains for all attributes. Then, each attribute domain is mapped into binary vector. That allows to keep each individual valid and there is no need to waste extra CPU time for repairing or removing invalid attribute values.

In proposed representation we use only “AND” logical operator, therefore EA to describe some set of instances as two separate conditions using ruleset as two connected rules. Indeed, the relation between these rules is logical disjunction, and indeed there exist a sort of rules coevolution phenomena.

B. Fitness Function

Generally, in EA, fitness function evaluation is very critical issue. Its definition decides about shape of solution landscape and must be defined very carefully. As rule extraction problem corresponds to data mining and our individual is a ruleset we can use commonly used classifier measure. In classification the main goal is prediction of the value $c_i \in C$ (class) analyzing values of attributes x_i of given instance $x_i = \{x_i^0, \ldots, x_i^n\}$ where $x_i^n \in X$ defines solution landscape. Thus classification task is based on explore set of $(x_1, c_1), \ldots, (x_n, c_m)$ to build model $m(x_i): X \rightarrow C$ that labels unseen instance $x_k \in X$. Evaluation of rule is connected to its quality as classifier. In such context of data mining domain, the terms true positives (*tp*), true negatives (*tn*), false positives (*fp*) and false negatives (*fn*) we can define *recall* :

$$recall = \frac{tp}{tp+fp} \quad (1)$$

and *precison* measure as follows:

$$precision = \frac{tp}{tp+fn} \quad (2)$$

The *recall* value tells only if given rule labels instances in proper way, *precision* informs if rule covers all labeled data by proper class identifier. Above formulas say a lot about rule classification quality but there are two separate values. Although EA should use only one value to evaluate rule in literature (e.g. [3]) we can find some combinations of these values. However, we decided to use measure based on modified van Rijsbergen's effectiveness measure (F*score*) [18], than can be used also in data mining, as it combines *precision* and *recall*:

$$Fscore = \frac{1}{\frac{\alpha}{precision} + \frac{1-\alpha}{recall}} \quad (3)$$

where α can give a predominance of one of two elements, but we established its value on 0,5 to keep two elements equal. If *Fscore* value is near 1 means that evaluated rule has high quality as it corresponds to maximizing problem. Bellow *Fscore* measure is very useful as fitness function form, but for comparison of gained results in literature is used other measure of predictive *accuracy*, as a rule quality measure, defined as follows:

$$accuracy = \frac{tp+tn}{tp+fp+fn+tn} \quad (4)$$

where *tp/tn/fp/fn* correspond respectively to true positive/true negative/false positive/false negative to denote accuracy of classification in given set of instances.

We do not use accuracy formula as fitness function because of more practical features of *Fscore.* As individual is a ruleset the fitness function value is calculated as the average value of *Fscore* of all existing in dataset classes. Such form of fitness function occurs some distortions specially when dataset is dominated by one class and others have small representation. In dominated datasets we can use stratified crossover mechanism to reduce such problem.

C. Genetic operators

Our individual is represented by binary vector which can be operated by classic binary operators in simple way. Random modification of selected bit works as mutation (***SM***, Simple Mutation) and inserts new information into chromosome. To deliver communication in population in CAREX we developed one-point crossover (***OX***) that links two individuals to build the new one as combination of their genes. As all individuals have the same size, there is no situation of invalid individual creation using the random cut position. The high efficiency of basic set of genetic operators (*SM* and *OX*) in CAREX encouraged us to construct some specialized operators. We developed and tested two: Directed Mutation (*DM*) and Best Class Crossover (*BCX*).

Directed Mutation (DM)

This operator is not semi blind as simple mutation *(SM)* and it tries to direct evolution process by increasing the mutation probability if given rule has low quality (fitness function value is low). Given *Rule$_i$* mutation probability is given according to below formula:

$$P'_m = P_m * (2 - Fit^{class_i}(Rule_i)) \quad (5)$$

where P_m is constant mutation probability given to whole *EA.* The $Fit^{class_i}(Rule_i)$ formula part gives a value of fitness value of given *class$_i$* classification accuracy. The motivation is to change rules in class that has worst quality value (near 0,0), but it is almost absent when value is near 1,0 (the best quality). Of course, such directed mutation selects rules to modification but if such direction is too hard it could stuck the whole learning process in local optima. To avoid such situation we establish an increasing parameter as maximal value to double P_m value.

Best Class Crossover (BCX)

The standard crossover operator (such as *OX*) does not take into consideration specific problem knowledge. It just randomly cuts individuals and randomly generates a new individual. Our main motivation was to build a new crossover operator that uses specific domain knowledge and links two individuals more effectively. We inspired slightly the *BCX* operator (presented in [13] which, in its base form, is used for graph coloring problem, we developed the other version of *BCX* (Best Class Crossover), which is presented as pseudo code on Błąd: Nie znaleziono źródła odwołania.

The BCX operator takes two RuleSet P_1 and P_2 as parental individuals and returns the newly generated as result. The offspring is compilation of two parental individuals based on analyse which a parental *RuleSet* has better accuracy of given class description c_i – its rules are included in newly generated offspring individual. Such operation is repeated for each class independently.

The operator is specific form of communication between individuals where each of them gives the best part of selected rules and it is very useful, but its frequent usage is rather

```
function crossover_BCX ( RuleSet $P_1$,
                         RuleSet $P_2$ )
  begin
  RuleSet $Off := \emptyset$
     foreach class $c_i$
       begin
              if $(Fit^{c_i}(P_1^{c_i}) > Fit^{c_i}(P_2^{c_i}))$
                  then $Off := Off \cup P_1^{c_i}$
                  else $Off := Off \cup P_2^{c_i}$
       end
  return $Off$
  end
```

Fig 1. BCX operator pseudocode

risky. As it is determinate, it can cause loss of diversity in population and premature convergence indeed. Such operator we use not more than P_x<0,1.

D. CAREX programming platform speedup

Our experiments needed programming platform and we decided to use Java language. To make experiments less time consuming we developed:

- data buffering, all data are stored in RAM memory not in hard drive – it makes data query more effective and less time consuming,
- data indexing based on binary search that gives us more effectively access to our data,
- data sequence covering based on attributes condition cascade techniques that give evaluation function of each rule less time consuming,
- evaluation cache, that stores in RAM memory rules that were already evaluated and in evolution process are without any changes. Thanks this mechanism time for evaluations is significantly reduced.

Such low level programming techniques (not connected to EA conception directly) make experiments less time consuming. It is worth mentioning because of stochastic character of EA, where experiments must be repeated many times.

III. Computational experiments and results

Evaluation of learning method is important to compare results against other methods. This can be done experimentally. We developed in Java a research environment that supports learning and rule validating process. To evaluate predictive classification ability of developed model we split data into train and test data using commonly used crossvalidation method.

First, there are used train data to generate *RuleSet* by CAREX to get possible high accuracy (*Fscore* is used as fitness function). Learning process based on evolution runs using selection, mutation and crossover operators. For evaluation accuracy of gained rules there is used train dataset, but when evolution process is finished test dataset is used to validate predictive accuracy of generated *RuleSet*. To minimalize a random elements influence on our research we use 5x2 crossvalidation (according to research presented in [7]). Also to reduce influence of stochastic aspect of EA to our research the whole evolution process is repeated 10 times to each crossvalidation fold.

A. Used datasets

To show CAREX application we decided to use benchmark data set from UCI data Repository [21]. From set of 187 datasets we selected four connected to important real-world domains such as biology, medicine, chemical industry, food industry or health care, as follows:

- **iris**: 150 instances, 4 attributes, 3 classes (distribution: 50/50/50),
- (pima indians) **diabetes**: 768 inst., 8 attributes, 2 classes (distrib.: 500/246),
- **glass**: 214 instances, 9 attributes, 6 classes (distribution: 70/17/76/13/9/29)
- **wine**: 178 instances, 13 attributes, 3 classes (distribution: 59/71/48)

First dataset (Fisher's iris flower) is benchmark for each new classification algorithm. The diabetes dataset includes medical records of medical review if the patient shows signs of diabetes according to the World Health Organization criteria. The third dataset consists of glass identification in chemical industry data. The last one (wine) contains results of chemical analysis of wines grown in the same region in Italy.

B. Experimental CAREX

We experimented to establish CAREX algorithm optimal parameters values. The most influential parameter is the probability of mutation (see its influence on average predictive accuracy presented on Fig 1). Our experiments showed that CAREX results are the most stable and the best in P_m=0,01. Probability of crossover P_x=0,2 but our first research result showed that crossover has minor influence on CAREX results.

Another very important CAREX issue is population size and number of generations. We developed several experiments series to establish its optimal values. Results of such experiments are showed in Fig 2, where influence of number of births (as product of population size and number of generations) to average accuracy of developed model in iris dataset is presented. We used population size and generations number from set {10,20,50,100,200 and 1000}. The Fig 3 shows that the optimal value of births is between 10 and 20 thousands, which is connected to population size 100 and 200-400 generations.

The higher values of births make CAREX less effective – it is more time consuming learning process and we do not have the significant increase of accuracy value.

Basically, we tested two types of selection methods: tournament selection and roulette wheel selection. The best results were achieved in tournament selection, where individuals are selected without replacement to the pool size of 2 or 10 individuals. The roulette wheel method causes smaller selection pressure and makes learning process less effective. In our experiments we do not use elitism parameter.

C. *Effectiveness of CAREX*

This part of paper presents experiments connected to population size and genetic operators usage influence on CAREX effectiveness. Evaluation of CAREX approach in the context of four benchmark dataset is given. Used values of EA parameters are presented in Table 1.

Table 1. CAREX parameters used in experiments

parameter name	IRIS	DIABETES	GLASS	WINE
Population size	200	20	200	100
Number of generations	200	200	500	200
Prob. of crossover [OX]	0,2	0,2	0,2	0,2
Prob. of mutation [DM]	0,01	0,01	0,01	0,01
Selection method	10	2	10	10
Number of rules	5	10	30	15

In CAREX an individual is represented by set of rules and this number strictly depends on used dataset: usually we use about 3-5 rules per class. As different datasets have various CAREX parameters requirements are presented in Table 1. The population size in our approach usually equals to 100-500 individuals. Number of generations gives evolution process the "time" to work out a solution but we experimentally tested that there is no need to extend it to not more than 1000 generations. Another important aspect of EA is selection – we experimentally developed a tournament selection of 2 or 10 individuals. A mutation operator (*Pm*) rate is experimentally set to 0,01. Stop condition is met if individual that has *Fsc* equals to 1,0 (success) or number of generation is exceed (fail).

Corresponding to literature we compare results of CAREX with other methods (see Table 2) - we selected EA based methods (e.g. CORE [20], CCE [15], ESIA [11]) and classical C4.5 [16] algorithm. For each dataset we presented results of each selected method (as it is given in literature): the average predictive accuracy, standard deviation value and maximal value of observed accuracy. Unfortunately, not for all dataset the results are given in the literature.

We can observe that CAREX achieves the best accuracy for **iris** dataset. Also comparable results gain CORE method and EMA-AIS. It is worth to mention that difference is very small (near to statistical error value). For **diabetes** dataset we can see that CAREX does not achieve the best accuracy (but comparable to classical C4.5 algorithm). It is outperformed by CORE and EMA-AIS method. It is also worth mentioning that EMA in its basic form gives comparable results to CAREX, while EMA-AIS is the hybrid of EMA with AIS and gives potentially better solution. Comparison of predictive accuracy value in **glass** dataset to other methods shows that CAREX results can successfully compete with other methods. Unfortunately, stochastic element of CAREX causes relatively large value of standard deviations (8,75%). That value is very puzzling, because ESIA approach uses 5-fold cross-validation and standard deviation equals to 0,03%. Dataset **wine** results presented in the above table show that given dataset is no problematic for CAREX. We compared our results with two other approaches, and CAREX outperforms classic C4.5 and gives results comparable to SEA method.

C. *ImageCLEF Photo Annotation competition dataset – first results*

In previous section we presented CAREX method applied to benchmark UCL Repository datasets. Improved high CAREX efficiency, we applied it to practical problem. We

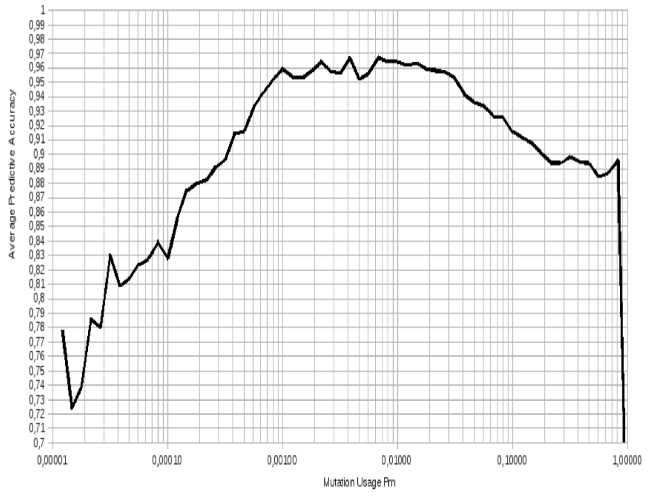


Fig 2. CAREX mutation prob. value' influence on average predictive accuracy [iris dataset]

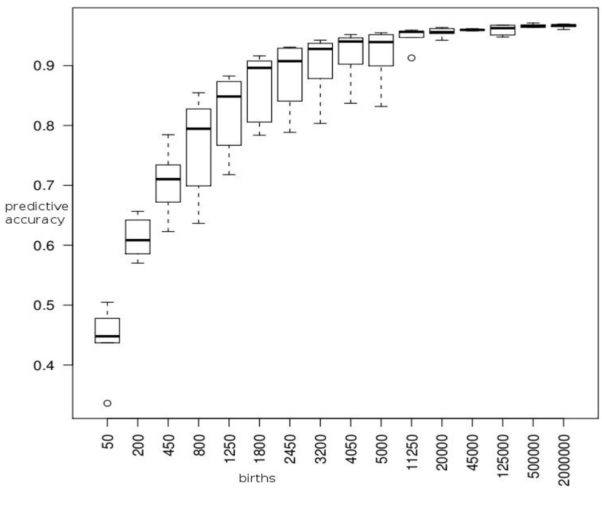


Fig 3. CAREX - number of births and its influence on average predictive accuracy [iris dataset]

Table 2. Performance comparison for the benchmark dataset.
Average predicative accuracy [%], standard deviation and best value of accuracy are given [%].

	IRIS	DIABETES	GLASS	WINE
CAREX	**96,67±1,5 (99,1)**	73,89±1,0 (76,43)	**75,49±8,75 (83,54)**	**93,93±3,68 (100)**
CORE [20]	**96,61±2,35 (100)**	**75,34±2,3 (80,15)**	n.u.	n.u.
C4.5 [16]	93,67±3,73 (100)	73,13±2,5 (77,39)	67,72±12,27 (?)	89,03±7,55 (?)
CCE [15]	95,40±3,28 (100)	n.u.	n.u.	n.u.
ESIA [11]	95,33±3,0 (?)	70,18±0,21 (?)	72,43±0,03 (?)	n.u.
EMA [2]*	94,59±3,9 (100)	73,23±2,13 (77,08)	n.u.	n.u.
EMA-AIS [2]*	**97,02±0,83 (100)**	**75,23±1,4 (77,6)**	n.u.	n.u.
SEA [10]	95,57±? (?)	n.u.	0,68±? (?)	**94,59±7,55 (?)**

decided to generate rules in photo annotation problem – and we used benchmark dataset from ImageCLEF conquest [14]. Automatic Image/photo annotation problem [4] needs method for multiclass labeling images, as problem concerns a number of classes and large size of datasets. There is many approaches in literature based on decision trees, suport vector machine (SVM) and others (survey can be found in [1]).

The ImageCLEF conquest dataset consists of 8000 images labeled by 93 classes (words). We do not use any segmentation methods and we use 18 global image Tamura [19] features such as coarseness, contrast or directionality. We run CAREX method using 10 individuals to run 100 generations to get one word annotation. The example of such run is presented on Fig 4, where is shown the best *Fscore*, average *Fscore*, worst *Fscore* and best Accuracy in given evolution process. Evolutionary Algorithm searches the whole solution space effectively and even so small population (only 10 individuals) makes possible to get optimal solution.

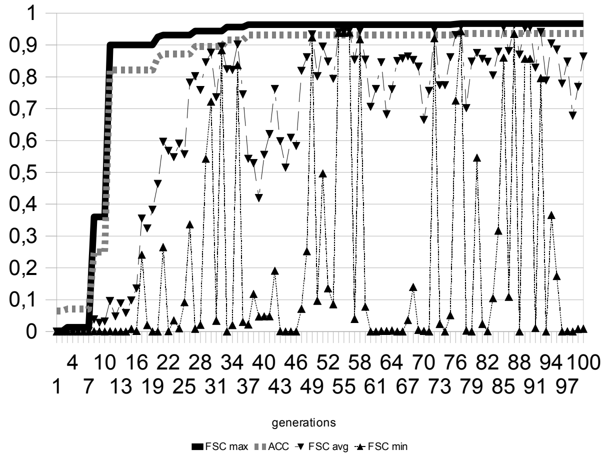


Fig 4. Example of CAREX run of Photo ImageCLEF word 'Neutral_Illumination'. CAREX parameters: 2 Rules, popsize=10, generations=100. P_m=0.01. P_x=0.2

Some initial experiments were done and effectiveness of presented CAREX method is very promising. First results were presented in Table 3, where the best of 10 word were shown. The average *Fscore* for 10 chosen words equals to 74,5%. However, these words are frequent and the average F-score value in dataset equals 26%. Example of 2 rules generated on word 'Neutral_Illumination' with very high value of *Fscore*:

Table 3. CAREX results for 10 best words for ImageCLEF images dataset

word	records	Fscore	Precision	Recall	Accuracy
Visual_Arts	3346	0,590	0,419	1,000	0,420
No_Visual_Time	3271	0,590	0,429	0,946	0,462
Cute	3909	0,657	0,490	0,998	0,491
Outdoor	4172	0,697	0,560	0,923	0,581
Day	4198	0,710	0,570	0,941	0,597
Natural	4593	0,730	0,575	0,999	0,576
No_Blur	5240	0,792	0,656	0,999	0,656
No_Persons	5467	0,812	0,684	1,000	0,684
No_Visual_Season	6646	0,908	0,831	1,000	0,831
Neutral_Illumination	7483	0,968	0,939	1,000	0,938
	average	0,745			

```
IF      a4 IN<58,128;341,378>
    AND  a7 NOT_IN<170,012;266,089>
    AND  a8 NOT_IN<291,938;312,497>
    AND  a13 NOT_IN<169,923;265,895>
    AND  a14 IN<178,128;345,282>
    AND  a15 IN<19,201;218,386>
THEN class = 'Neutral_Illumination'
  [Rec=0,013 Prec=0,981 Acc=0,077 Fsc=0,027]
IF      a7 NOT_IN<586,874;603,571>
THEN class = 'Neutral_Illumination'
  [Rec=1,000 Prec=0,935 Acc=0,935 Fsc=0,967]
```

The CAREX in the first results is very promising but also needs some extra ImageCLEF experiments connected to specialized genetic operators, another set of features (not only 18 Tamura features), segmentation influence to CAREX effectiveness and time consuming EA run as extra tests. Another positive aspect of CAREX usage is very intuitive rule representation that allows to understand the annotation description by the human.

IV. Conclusions and further work

The presented method CAREX is based on EA and simple binary solution representation. We wanted to show that binary based EA can be effective in searching a large search space in data mining tasks. We presented results of performance based on four benchmark databases. The CAREX results are compared to other methods and show that CAREX is able to compete successfully in rule induction/classification task. Also, in our opinion CAREX has great potential in this area of research. However, experiments results showed some problems, especially connected to stochastic aspects of presented approach – in our opinion the standard deviation value of results is too high to be accepted in real-world applications. We presented first experiment results on real world of image annotation problem using the ImageCLEF dataset. Results of developed experiments encourage to continue work on CAREX.

We plan further research into two main directions. First, fitness function modification by adding some niching methods to make it more competitive. Other way we see in specialization of genetic operators, including some ruleset improving methods such as hill climbing, local search type or simple pruning rules procedure and discovering potential labeling conflicts in the whole ruleset.

V. References

[1] Alham N. K., Li M., Hammoud S., Qi H. "Evaluating Machine Learning Techniques for Automatic Image Annotations" Proc. of the 6th Inter. Conf. on Fuzzy Syst. and Know. Discovery. vol 07, pp: 245-249, 2009

[2] Ang J. H., Tan K. C., Mamun A. A., *An evolutionary memetic algorithm for rule extraction*, Expert Systems with Applications 37, pp.1302-1315, 2010.

[3] Bojarczuk C., Lopes H., Freitas A., *Genetic programming for knowledge discovery in chest pain diagnosis,* IEEE Eng. Med.Mag 2000:19(4):38-44, 2000.

[4] Broda B., Kwasnicka H., Paradowski M., Stanek M.: MAGMA - efficient method for image annotation in low dimensional feature space based on Multivariate Gaussian Models. IMCSIT 2009 proceedings 131-138, 2009.

[5] Cattral R., Oppacher F., Graham Lee K. J., *Techniques for Evolutionary Rule Discovery in Data Mining*, IEEE Congress on Evolution. Comp., Norway 2009.

[6] Carneiro G., Chan A.B., Moreno P.J., Vasconcelos N., "Supervised Learning of Semantic Classes for Image Annotation and Retrieval," IEEE Trans. on Pattern Analysis and Machine Intelligence 29(3), pp. 394-410, 2007

[7] Dietterich T. G., *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithm*, Neural Comp. 10, pp.1895–1923, 1998.

[8] Freitas A. A. *A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery,* Advantages in Evolutionary Computing: theory and applications, pp.819-845, Spinger-Verlag NY, 2003.

[9] Goldberg D., *Genetic algorithms in search, optimization and machine learning*, Addision-Wes., 1989.

[10] Halavati R., Souraki S.B., Esfandiar P., Lotfi S., *Rule Based Classifier Generation using Symbiotic Evolutionary Algorithm*, ICTAI, vol. 1, pp.458-464, 19th IEEE Inter. Conf. on Tools with AI, 2007.

[11] Liu J. J., Kwok J. T. *An extended genetic rule induction algorithm*, Proceedings of the Congress on Evolutionary Computation (CEC), pp.458-463, La Jolla, California (USA), 2000.

[12] Michalewicz Z., *Genetic Algorithms+Data Structures =Evolution Programs,* Springer-Verlag, 1994.

[13] Myszkowski P. B., *Solving Scheduling Problems by Evolutionary Algorithms for Graph Coloring Problem*. Metaheuristics for Scheduling in Industrial and Manufacturing App, pp.145-167, Springer -Verlag 2008.

[14] ImageCLEF Photo Annotation competition dataset: http://www.imageclef.org/2010/PhotoAnnotation

[15] Stoen C., *Various Collaborator Selection Pressures for Cooperative Coevolution for Classification*, International Conference of Artificial Intelligence and Digital Communications, AIDC 2006.

[16] Quinlan J. R. *Bagging, Boosting, and C4.5*, Proc. of the 13th National Conf. on AI, pp. 725-730, 1996.

[17] Rodriguez M., Escalante D.M., Peregrin A., *Efficient Distributed Genetic Algorithm for Rule Extraction*, Applied Soft Computing. (accepted, Jan 13th, 2010).

[18] van Rijsbergen, C. J. (1979). *Information Retrieval* (2nd ed.). Butterworth. 1979.

[19] Tamura H., Mori S., Yamawaki T. "Textrual Features Corrensponding to Visual Perception", IEEE Transactions on Systems, MAN and Cybernetics 8(6), pp.460-473, 1978.

[20] Tan K. C., Yu Q., Ang J. H., *A coevolutionary algorithm for rules discovery in data mining*, Inter. Jour. of Systems Science, Vol.37, No.12,pp.835-864, 2006.

[21] UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/)

ISBN 978-83-60810-27-9
ISSN 1896-7094

Evolutionary Algorithm in Forex trade strategy generation

Paweł B. Myszkowski
Wrocław University of Technology,
Wyb. Wyspiańskiego 27, 51-370 Wrocław, Poland
Email: pawel.myszkowski@pwr.wroc.pl

Adam Bicz
Wrocław University of Technology,
Wyb. Wyspiańskiego 27, 51-370 Wrocław, Poland
Email: cyklop@gmail.com

***Abstract*—This paper shows an evolutionary algorithm application to generate profitable strategies to trade futures contracts on foreign exchange market (Forex). Strategy model in approach is based on two decision trees, responsible for taking the decisions of opening long or short positions on Euro/US Dollar currency pair. Trees take into consideration only technical analysis indicators, which are connected by logic operators to identify border values of these indicators for taking profitable decision(s). We have tested the efficiency of presented approach on learning and test time-frames of various characteristics.**

***Keywords:* financial data mining, evolutionary algorithm, genetic programming, decision tree induction, trade strategy, Forex**

I. Introduction

IN THE literature we can find many approaches to the automatic trading and many attempts to use artificial intelligence methods to trade on stock markets. A good summation on large part of the work already done in this matter is [2], where the list of surveyed markets, potential input data and modeling methods were shown. On financial markets automatic trade systems are widely used (especially by great investment funds), but the rules of taking the decisions are not publicly known. We don't know how far they are human controlled, and how far autonomic they are in their decisions. There is a lot of artificial neural networks (ANN), evolutionary algorithms (EA) or genetic programming (GP) usages as appropriate methods to search rules and strategies that could use ineffectiveness of the markets to earn money. How far they are well developing on current, not historical data, is an open question.

We have decided to investigate the usability of EA to generate rules, which will trade on foreign exchange market (Forex). Forex is a very interesting market because of its liquidity and open hours. It's closed only on weekends, normally operating 24 hours per day [7]. It means that (sometimes huge) price gaps between trading days, what is a normal behavior in stock markets, are here rarely observed. Thanks to high liquidity we can always sell and buy any amounts of assets at actual price without the risk that we will strongly affect the price or run out of contrary orders (thus being unable to sell or buy at reasonable prices), at least if we are individual investors and not great investment funds. Liquidity means constant movements, which means constant opportunities to make profitable trades. Thanks to high leverage and very small transaction costs it's possible to generate high profits even on very small price movements (such as 0.1% changes). Unfortunately, the leverage means the risk of taking high losses or even losing whole invested capital.

A. Related works

In the literature there are many interesting approaches, which take into consideration an automating profitable trade strategy generation.

In paper [16] is described GP to build decision trees. There are two types of decision trees, based on moving averages values and filters – decision is being taken after stock value rising up over last maximum or falling down below last minimum for specified percent value. Decision trees evaluated by GP can use arithmetical operations, functions, logical operators (incl. negation), conditional operators (IF-THEN), numerical and logical constants. However, its search space is being very big. Also in [16] daily intervals are used, where we used 10 minutes intervals. It means that the transactions have a much longer investment time horizon. Such approach has successfully generated many profitable strategies on various currency pairs, at limited risk. The risk value is calculated with beta indicator[1], which was compared with its value for some benchmark portfolios based on various stock market indicators. What's interesting, rules generated for one currency pair were much worse, when used on others.

GP can be used also for trade strategies generation on Warsaw Stock Exchange were investigated [15]. Based on the decision trees stored in GP individual a portfolio up to 10 stocks is build. The content of the portfolio vary in time. After doing several experiments authors stated that the profits earned by automatically generated strategies exceed those generated by benchmark (WIG stock indicator as buy&hold strategy profit). On decreasing time-frames no tremendous profits were generated, but substantial losses were avoided too and according to decreasing character of used time-frame such results are acceptable.

In work [13] EA is used to optimize values for predefined filters. Authors optimize and test filters on data from Australian Stock Exchange. They find out that EA finds profitable strategies more effectively than greedy algorithm. This conclusion is important if we want to use this solution for real time usage.

Paper [18] shows that there is no direct correlation between the distance from the learning to testing time-frame

[1] measures the correlation between an investment's value and movements in the overall market

and the efficiency of gained strategy. There is another interesting conclusion that no positive results of removing strategies, which behave worse on some validating time-frames, where found. It means that if we will remove strategies, because they are not as good as the others on one validating frame, we probably can lose one of the best strategies for another time-frame.

In work [9] EA usefulness to generation set of rules is investigated, which decides about the moments to buy or sell stocks on the Paris Stock Exchange (France). The individual there is represented not as a tree, but a sequence of bits, where each bit responses for usage of technical analysis indicators. Such representation allows EA to search the solution space very effectively. As there were shown, strategies generated in such way give better results than benchmark buy&hold strategy. What's important, there wasn't found any proof that some stocks are easier in prediction. Also, that constant generating of new rules on current data can improve the results, which is contrary to the conclusion from [18]. Generated rules were profitable only on those stocks on which they were learned, and there weren't found any sign of some indicators being preferred over others.

Another EA usage were shown in [1], where task was to find technical trading rules on Standard and Poor's composite stock index. Rules generated by EA have form of decision trees, where the nodes can be logical operators, functions (avg/max/min), arithmetic operations, comparisons or constants. Training data were given from years 1963-69, and test data from the years 1970-1989 and other periods since 1929. The rules generated excess returns only on the first test period. On periods before 1963, including the transaction costs, they generated losses.

In [12] method searches for similarities in the history of stock value movement to predict its values. EA is used to find as good as possible analogies between some latest intervals and those taken from history of given stock. Such historical data are preprocessed, where outliers are removed to make the predictions more accurate. There was proven that filtering outliers and replacing them with moving average or just average values from the nearest neighbors gives positive results. Prediction results of volume values are much worse, mostly because of some huge random fluctuations.

The stock trading predictions system was build in [17] . As predictions there were used ANN, dynamic time-frames and case based reasoning technique. The ANN was used to predict buy/sell points for interesting stocks, previously chosen by some pre-defined rules, e.g. monthly sales, daily trading volumes, 5 day average volumes. Such method generated significant revenues on test data, even for stocks in downtrend.

In [20] there is another interesting ANN usage, where the task is to predict "next" values of Warsaw Index Futures. Technical analysis techniques – such as channel breakout, Bollinger Bands, Momentum Oscillators, Moving Averages – are used in combination of ANN predictions and without them, as competitive strategies. Technical analysis usage with ANN forecast makes possible to generate profits even for strategies, which normally were not profitable at all.

In [10] consists in Evolution Strategies usage to portfolio optimization problem. The main goal was defined to minimize the risk at fixed level of the expected return. Risk was measured as a semi-variance of returns (similar to variance, but emphasizing the "downside risk" - *actual return - expected return* but only if it's smaller than 0). Artificial "experts" were built by the EA, using some of predefined technical analysis rules (chosen by the EA algorithm). The portfolio of stocks was chosen at the beginning, and artificial "experts" were able to reduce or increase the share of each stock in the portfolio. There is stated, that the results were much better than random portfolio shares (buy&hold strategy), but worse than market index. Worth noticing, that buy&hold is here understood as keeping initial (random) portfolio shares unchanged, and market index is market value weighted (bigger companies has bigger share in the index). They stated that additional research is needed.

Work [8] uses a hybrid approach based on various types of ANN and EA to detect interesting patterns in stock market, where data came from the daily Korea Stock Price index 200. EA were applied to optimization of time delays and network architectural factors. The various types of ANN were connected into a multilayer feed-forward network, where time series were the input. Also there were tested and compared time delay ANN, adaptive time delay ANN, recurrent ANN and stated that their integrated approach gives better results than standard ANN.

Another interesting approach was used in [19]. A data mart was created and data mining techniques applied to predict future stock prices using historical values. A traditional grey prediction[2] model was expanded by fuzzy logic model. Data for each stock were pre-classified in groups using the "time" attribute. On the test data, the average deviation from the predicted highest stock price and the actual price was less than 9%, which authors interpreted as pretty accurate, as "predictions are feasible".

Another interesting approach presents [4], where Genetic Network Programming (GNP) was used for building stock trading model. Build network consists of judgment and processing nodes. Judgment nodes use technical analysis indicators and candlestick chart formations to decide which next node should be processed, and processing nodes take actions such as buying/selling stocks. Training and test data were taken from Tokyo Stock Exchange (Japan). As results are very promising, is suggested that it was achieved due to the representation of solutions as graph structures, which allows memorizing past actions sequences more effectively.

In our work we generate trade strategies as BUY/SELL decision trees, which are responsible for opening long or short positions on the EUR/USD currency pair on Forex market.

This paper is organized as follows. Section II describes details of proposed EA based approach: representation schema, genetic operators, selection method and the form of the fitness function. Research to find as good as possible EA parameters is shown in section III. Experiments are present-

[2] grey system theory requires limited amount of data to estimate the behavior of unknown systems [5]

ed in section IV, where are defined used dataset, research methodology and gained results. Section V shows possible further research and issues that have been skipped in this work. Section VI consists of conclusions and short summary of this work.

II. Proposed Approach

Our approach is based on GP [11] and is slightly inspired by [15]. In our work EA generates trade strategy based on BUY/SELL decision trees that consists in technical analysis indicators that are connected by logical operators. We decided to work with EUR/USD currency pair, because it's the most popular and liquid pair as it generates 27% of the total volume on Forex [3].

A. Individual representation

Each individual is represented as two decision trees, where one is responsible for BUY signals, and one for SELL signals recognition. Each of the decision trees is actually a trading rule. It consists of logical functions binding leafs, which are technical analysis indicators.

Used technical indicators are some variants of Simple Moving Averages (SMA), Weighted Moving Averages (WMA), Displaced Moving Averages (DMA), Relative Strength Index (RSI), Moving Average Convergence-Divergence (MACD), Rate Of Change indicators (ROC), volume value and volatility for specified period values. They are widely described in [14] as very useful tools for the investors to determine trends, reversal points, to identify interesting buy/sell points. They are very popular among investors of any kind, but can be used in sometimes very complex ways, where the same indicator value can be interpreted by various investors in a different way.

In decision tree each node can be logical (and/or) or terminal node in form *(Indicator [<, >] Value).* Example tree can be presented as follows:

```
  ((ISMINOF<11) or ((ROC (5)<0.234) and
(MACD (26,12)<-1.155))) and (VOLUME<833)
```

Example of decision tree is shown in Figure 1.

B. Selection method

We used a combination of tournament and roulette method. We select randomly 5 (value is set experimentally) individuals, whom we rate and sort in descending order. We chose one individual with probabilities 5/15, 4/15, 3/15, 2/15, 1/15. After selection mutation and crossover generates an individual.

C. Genetic operators

In our work mutation and crossover operators are used.

Mutation operator works on one individual and gives some changes in its genome. Mutation is stochastic based, some methods of decision trees:

- leaf mutation, (I) where a random value is chosen from the possible value set for given indicator (60% chance), (II) mutation changes inequality sign from ">" to "<" and in another direction (16% probability) and last one (III) changes the indicator type (24% chance)
- mutation for logical nodes, that changes "and" into "or" and in another direction (50% chance)
- mutation of tree size: (I) that reduces of given tree to left or right child (50% chance), (II) split given leaf into two leaves connected with „or" or „and" node. The second node is generated randomly (40% chance for each child, only if not at maximal height) and (III) switches left and right child (10% chance)

Crossover operator links two individuals and switches their buy/sell decision trees. The probability usage of crossover was experimentally set to 3%.

D. Fitness function

We have chosen a fitness function that is closely related to the real use of strategies in the stock market, which is the possibility of such strategy to generate gains. It must include transaction costs, because ignoring them can result in over reactive strategies, generating losses if transaction costs are taken into consideration.

Fitness =

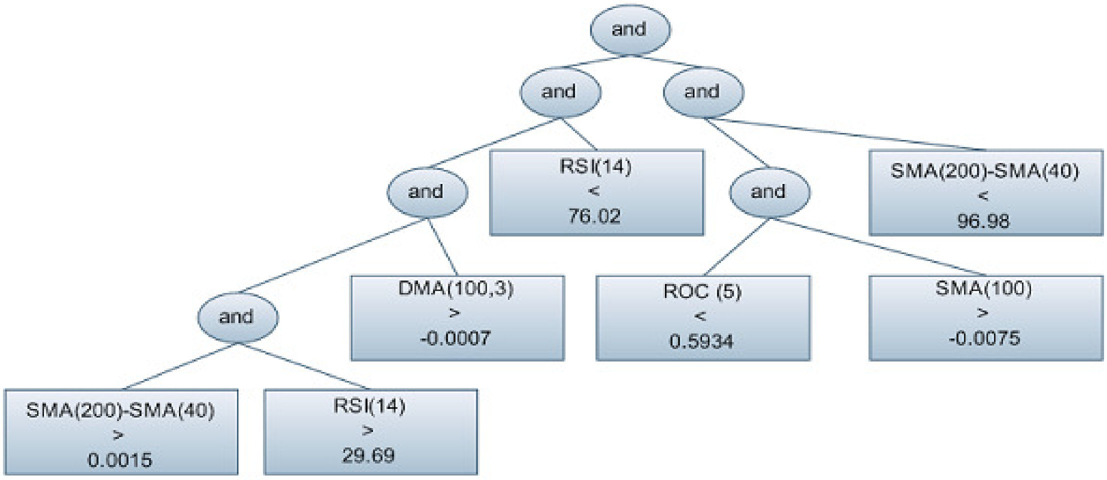


Fig 1: Example of buy decision tree. The presented decision tree as rule: *(*(((SMA(200)-SMA(40)>0.00150) and (RSI(14)>29.69711)) and (DMA(100,3) >-0.00073)) and (RSI(14)<76.02417)) and (((SMA(100)>-0.00757) and (ROC (5)<0.59348)) and (SMA(200)-SMA(40)<96.98830))

```
Sum of all profits/losses
+ Transaction count * Transaction cost
– MAX (0, 4 - Transaction count) * 8
```

Our fitness function form needs stock simulation, playing with the chosen strategy on the learning set data, and taking the result in points (adjusted by transaction costs as provision that equals 4 points). The last component in given function is responsible for faster learning to make more transactions rather than trying to buy/sell and hold. The 4 constant means, that if the strategy has made at least 4 transactions, there will be no punishment. For each transaction less than 4, it gets a punishment of 8 points (which is 2 times greater than transaction costs, which equals 4 points). It means that the punishment for the simplest buy&hold strategy is 24 points as the maximal value.

III. Research on EA parameters

All parameters of EA were tested to find the best values. Searching for EA parameters is a multidimensional problem. Additionally best parameters vary for each of the learning frames, so our results are only some approximation and trade-off between many possibilities. Results were obtained from repeated 200 runs under the same conditions, which generated 200 strategies (1 for each run). Average profit inc. provisions was taken to decide which parameter value is most promising.

Example of such parameter is *maximal tree height.* Some tests (e.g. L1 time-frame, see Table I) have shown, that best results can be obtained with the maximal height equal 5 or 6. Too high trees mean probably too big search space, however too small trees are too simple to represent the complexity of price movements. Maximal height means, that tree can be lower, but never higher as specified height (if mutation could make the tree higher, it will be stopped). Our experiments have shown that the optimal values for transaction provision is 4 points, probability of mutation 72%, and probability of crossover 3%.

In Table II we have shown the averaged profits of strategies in population of *population size* parameter. We can see, that populations greater and smaller than 200 give worse results as the optimal 200, by 300 even much worse.

Strategy profits are measured on test data, not on training data. Stop condition (*generations*) amount in all tests is fixed and equal 150.

IV. Experiments

In our tests we have generated a very well developing strategy (verified on test data), which we will be describe in details in this section.

A. Used datasets

The EA operates on time-frames consisting of about 1400 of 10 minutes intervals, which is about 2 weeks of currency pair historical values [6]. To obtain the fitness of each rule pair (BUY/SELL decision) we virtually play stock with these rules, opening and closing positions on historical data (learning periods in the learning phase, testing periods in testing phase). Example of used datasets (time frames) in learning or validation process (cross-validation was applied for testing):

Falling:
L1 (2009-12-11 10:50, 2009-12-24 15:10) intervals: 1412
L2 (2010-01-12 19:40, 2010-01-26 00:00) intervals: 1412
...
L8 (2007-05-15 19:20, 2007-05-29 00:00) intervals: 1412
Neutral:
N1 (2009-09-24 00:00, 2009-10-07 00:00) intervals: 1386
N2 (2009-11-16 15:50, 2009-11-27 21:50) intervals: 1428
...
N6 (2009-11-02 15:50, 2009-11-13 21:50) intervals: 1428
Rising:
H1 (2009-08-30 23:10, 2009-09-11 00:00) intervals: 1397
H2 (2009-10-07 00:00, 2009-10-20 00:00) intervals: 1386
...
H9 (2009-06-23 04:20, 2009-07-07 00:00) intervals: 1412
Long time-frame:
LONG1 (2008-09-01 15:50, 2008-10-06 02:40) int.: 3593

There were used 24 time-frames for learning and testing process, 3 additional only for testing process. Charts for time-frames of selected L1, L2, L8 and H1 are shown in Fig 2. We can see, that in each of the time-frames a one directional price movement (so-called *trend*) could be noticed,

Table I. Results of testing of 200 strategies produced with various max tree height parameter

Max. tree height	2	3	4	5	6	7	8	9	10
Avg. profit	1028	933	1155,9	**1221,6**	**1290,7**	1073	1094,2	870	985,6
Avg. profit (incl. prov)	599,2	506,1	728,9	**806**	**866,8**	633,4	663	430	541

Table II. Results of experiments with various population size parameter

Population size (individuals)	25	50	100	150	200	250	300
Avg. profit	1000	800	1153	778	**1309**	1060	769
Avg. profit (incl. prov)	590	378	678	338	**878**	614	282

with various short term deviations from the trend line. If the time-frame has more such big-enough short term trends, it's more interesting for our algorithm, because it gives an opportunity to learn when to switch from long to short position. We can see, that time-frame L2 gave very few opportunities to earn more than a simple strategy "sell at the beginning and wait" (sell&hold strategy) (if we consider transaction costs). Time-frame L1 was much more interesting, because there were many rapid movements which resulted in corrective movements.

Simulating FOREX stock strategy can meet three types of situations in the market with long, short position or out of the market. We admitted to keep always 1 position open. The profits are measured in points (or so-called *pips*), which are the fourth number after the point, for example value 1,2548 means 8 pips, instead of percent difference. It is standard measure used in future markets, because each investor can use other leverage and so the percent difference varies, even if they open identical transactions.

B. *Example of strategy*

Example of BUY decision tree is presented on Figure 1. It's a decision tree taken from a strategy which was best behaving on the test data. Transactions of this strategy on H5 dataset are shown in Table III. Such rule is easy to analyze, because for example it expects, that RSI indicator (for 14 intervals) has higher value than 30, and lower than 76, which are similar to values mentioned in the literature [14], 30 and 70. It means that the algorithm try to avoid extreme values (peaks). The important part of the rule is that SMA for 200 days must be at least 15 points higher than the SMA for 40 days, so it buys if the prices are relatively low, but at the same time not extremely falling down (already mentioned RSI > 30). Other parts of the rule are either only small optimizations or are of no meaning for the strategy, at least on the test data. Rule describing the SELL decision tree:

```
( (DMA(100,3)>-0.00082) or ((SMA(200)
>0.00238) or (SMA(200)>0.00349)))
and (WMA(200)<-0.00067)
```

To understand how the SELL rule works, we can see, that it sells when either the DMA with parameters 100 and 3 (average on 100 intervals, delayed by 3 intervals) is higher than the actual price minus 8 points or the SMA of 200 intervals is higher than price +23 points and in both situations the WMA is lower that actual price minus 6 points. It's hard to judge if this rule is reasonable, because it's not so obvious. The last part using WMA means, that we will sell only if the price is a bit better that the average price in the past (200 intervals) – it ensures, that we will not sell if the price is too cheap.

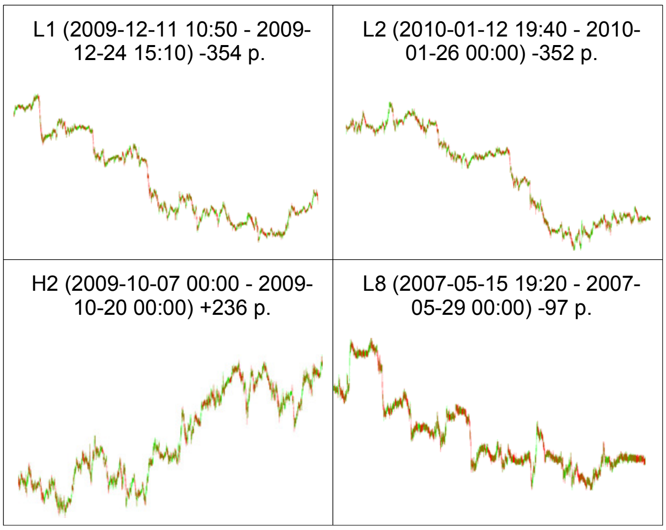


Fig 2. Example charts of the learning sets character. Time frames L1 and L8 are interesting for learning set than L2, which has fewer possibilities for position reversal.

C. *Tests*

In our research test (results are presented in Table IV and Table V) we are running our EA on all learning sets 200 times to get the best strategy for each run. After that we have generated 200 strategies to test their profit on 3 data sets.

To describe its genealogy, strategies learned on a specific time-frame are specified as the name of training time-frame and a name of the test data-set in brackets, for example "L1 (T1)" means strategies learned on time frame *L1* and tested on *T1* data set. Used data sets (*T1*, *T2*, *T3*) of time-frames are specified as follows (in brackets are the results of using *buy&hold* and *sell&hold* strategies, on each time frame from the dataset separately, and then adding the results):

T1: 24 time-frames (*sell&hold* strategy including provisions: 731 points, *buy&hold* strategy: -827 points)

T2: 23 time-frames (*sell&hold* strategy including provisions: -1051 points, *buy&hold* strategy: 959 points)

Table III. Table of transactions on h6 time-frame of the best found strategy (profits are given without provision cost)

Transaction type	Opened	Closed	Opening price	Closing price	Profit [in points]
LONG	2009-05-28 11:50	2009-06-01 00:00	1.3855	1.4101	**246**
SHORT	2009-06-01 00:10	2009-06-02 07:30	1.4101	1.4114	**-12**
LONG	2009-06-02 07:40	2009-06-03 03:00	1.4121	1.4287	**166**
SHORT	2009-06-03 03:10	2009-06-03 13:30	1.4282	1.4202	**80**
LONG	2009-06-03 13:40	2009-06-04 18:40	1.4174	1.4181	**6**
SHORT	2009-06-04 19:50	2009-06-05 14:40	1.4181	1.4023	**158**
LONG	2009-06-05 14:50	2009-06-10 00:00	1.4006	1.4069	**63**

T3: 24 time-frames (*sell&hold* strategy including provisions: -1227 points, *buy&hold* strategy: 1131 points)

For reasons of clarity: we were testing our strategies on data sets, which consist of 23-24 time-frames. Total amount of used time-frames is 27. Time-frames are independent, short time periods (shown in Fig 2), and data sets *T1*, *T2* and *T3* are collections of time-periods of various kind (rising, falling), but adding the price changes, causes rising (*T2*, *T3*) or falling (*T1*) type. We have used 3 data-sets instead of one, to show how the generated strategies behave on rising and falling data-sets. Using only one data-set does not allow us to do this.

In the traditional stock market we can always compare results of our strategy with a simple *buy&hold* strategy, which means simply buying at the beginning and selling at the end, which is the simplest way to invest. Growth of the market is the expected direction, as we all expect that the economy will rise in a long term.

Differently from the stock market on FOREX we have no real benchmark strategy – *buy&hold* strategy is exactly as good as *sell&hold* (sell at the beginning, buy at the end, such strategy is possible when we consider futures market), because the expected market direction isn't known. Such situation occurs because we always consider pair of currencies, for example EUR/USD, which we could as well switch to USD/EUR to get the ideal opposite of all price movements. That's the reason why we are showing the results of both strategies – *buy&hold* and *sell&hold*. Of course, provision costs are included, which arise when we open a transaction at the beginning of each time frame and close it at the end. The transaction costs are 2 point per transaction (1 for opening and 1 for closing), which is reasonable value on EUR/USD pair. We used transaction costs equal 4 only in the learning approach, to get more steady strategies.

In Table IV and Table V are given some experiments results. Benchmark *(buy&hold)* and *(sell&hold)* columns are the same values as those for datasets, adjusted by the learning time frame which is always taken out of the consideration. Positive *buy&hold* means that the prices on the test data was mainly raising, negative means the opposite. It's including provisions. Profits are presented in points rather than percentage, because we always have one position

Table IV. Results of testing of 200 strategies produced on 5 rising time-frames and one long time-frame on 3 data sets

Learned on (tested on)	H1 (T1)	H1 (T3)	H9 (H1)	H2 (T1)	H2 (T3)	LONG1 (H1)	SUM
Benchmark (buy&hold) incl. provision	-1115	908	-1028	-1126	897	1131	-333
Benchmark (sell&hold) incl. provision	1023	-1000	936	1034	-989	-1227	-223
Avg. profit **not** incl. provision	217	172	-289	-824	1238	-220	294
Avg. profit incl. provision	-67	-70	-680	-1341	763	-451	-1846
Avg. profit with transaction balancing incl. provision	-30	-29	-604	-1267	825	-413	-1518
Combined strategy including provision	**-960**	17	-259	**-403**	1398	-455	-662
Standard deviation (from profit incl. provision)	1013	934	796	890	716	705	-
Max (from profit incl. provision)	2574	1993	2074	1856	2758	1814	-
Min (from profit incl. provision)	-2938	-2644	-2896	-3256	-1827	-2532	-
Median (from profit incl. provision)	-161	-217,5	-777	-1112	843	-386	-

Table V. Results of testing of 200 strategies produced on 5 falling and 3 neutral time-frames on 3 data sets

Learned on (tested on)	L1 (T1)	L1 (T2)	L2 (H1)	L8 (T1)	L8 (T3)	N1 (H1)	N2 (H1)	N6 (H1)	SUM
Benchmark (buy&hold) incl. provision	-536	**959**	-538	-793	1230	-816	-920	-1032	-2446
Benchmark (sell&hold) incl. provision	**444**	-1051	446	701	-1322	724	828	940	1710
Avg. profit **not** incl. provision	2246	2037	-156	**1547**	1585	189	409	142	7999
Avg. profit incl. provision	**1381**	**1295**	**-688**	**752**	790	-565	135	40	3140
Avg. profit with trans. balancing incl. provision	1493	1381	-542	**847**	919	-462	184	51	3871
Combined strategy including provision	1533	1125	**592**	1354	1458	-95	-157	**379**	**6189**
Standard deviation (from profit incl. provision)	1245	972	1102	1500	999	1019	968	729	-
Max (from profit incl. provision)	2457	2843	3215	2596	3062	2062	3312	2968	-
Min (from profit incl. provision)	-4425	-3196	-3022	-4323	-3660	-3761	-2556	-2391	-
Median (from profit incl. provision)	**1810**	**1362**	-851	1174	946	-563	99	-7	-

opened, so earned points mean always the same real profit (10 dollars per point). That means, that consecutive profits add linearly, not exponentially. To get exponential results we have to open variable amount of contracts. It's difficult to give the results in percentage, because we need to know the leverage of the potential investor.

Average profit is the profit earned by 200 strategies generated on present learning set, ignoring transaction costs. *Avg. profit incl. provisions* are average results adjusted by transaction costs (2 points per transaction). *Avg. profit with transaction balancing (incl. provision)* is an average profit per strategy generated by playing with all strategies simultaneously, and balancing together buy and sell transactions opened on the same interval. This value is always between avg. profit and avg. profit including provision. *Combined strategy (incl. provision)* is a result of a strategy being built from all 200 strategies generated by EA. We run all strategies simultaneously, but open a transaction only if at least 60% want to open it and agree for the same direction. We are always playing with maximally 1 opened position. *Standard deviation, max, min* and *median* are some statistical functions run on all profits including provision for all 200 strategies.

We can see that strategies learned on time-frames *L1* and *L8* (see Table V) give extraordinary profits on both rising as falling data sets. For example if one has been very bullish on all time frames from second data set he would have earned 959 points, but average strategy (learned on *L1*) has gained 1295 points. The median of all strategies is high: 1362 points. Results on first data set are even better – bearish *sell&hold* strategy would have earned 444 points, where average generated (on *L1*) strategy would have earned 1381 points. Each point is valued as 10 dollars gain for the investor with one position opened, so it means $13.810 gain per one standard contract (so called *lot*), which is a very good result.

Other interesting result is the value of avg. profit with transaction balancing vs. without balancing. If we run all strategies simultaneously and balance trades which are contrary to each other we can spare 12-13% on transactional costs. It means for example that instead of earning 752 points we can earn 847 points (see Table V, it would be 1547 points without transactional costs) on one data set using strategies generated on *L8*. Unfortunately we must be ready to have so many opened contracts as many strategies we use, which limits the use of this strategy to either a limited choice of strategies or investors with great amounts of cash allowing opening 100-200 contracts.

The ultimate solution to the already mentioned problem is a combined strategy. We take our 200 EA generated strategies and build one strategy from it. The rule is easy, if 60% of all strategies show buy signal, we buy. If 60% of all strategies generate sell signal, we sell. In this case uncertain strategies (what happens pretty often) act against opening transactions. Interesting results can be obtained too, if we decide to leave the market if the uncertainty rate is higher than 90-98%.

Such combined strategies have one main advantage: they are single strategies. We can open just one contract, and still get the knowledge from all strategies and get the same or even better results as the average of all these strategies. In fact the results of such combined strategy are nearly always better as avg. profit for all strategies (for *L1* tested on second dataset the result was a little bit worse – 1125 vs. 1295 points, but still comparable, the same for *LONG1* (see Table IV), only on *H1* (*T1*) the results for combined strategy were much worse than average, -960 vs. -67 points).

In the Figure 3 we can see a schematic representation of our combined strategy mechanism. In the lower part we can see how many strategies were bullish (light bars) and bearish (dark bars). They are placed on each other. If the amount of bullish/bearish transactions crosses the decision line, a new position is opened or the old one closed. We can move this line higher or lower, so we can possibly ignore some signals or not. Studies have shown that optimal values depend on the learning set and vary between 42-62%.

We can see that strategies generated on *L2*, and *H2* (*T1*) were behaving worse, than even a wrong direction of buy/sell and hold strategy. It’s alarming, because it means that if we choose wrong learning set, the results can be really bad, although not much worse than keeping a wrong position opened. We cannot forget that it’s an expected result. There are some time-frames which give a great opportunity for our EA algorithm to extract some reasonable rules and some that are not. We have to choose interesting time-frames by hand and by using some additional testing.

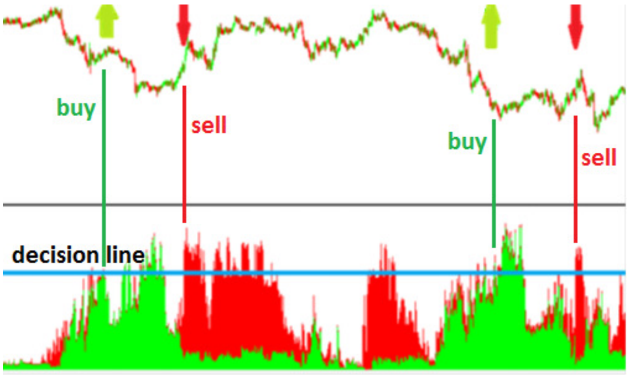


Fig 3: Combined strategy as an additional indicator. Chart below the actual price chart shows the distribution of bullish and bearish strategies (at each interval of time).

Such interesting time-frames are definitely *L1* and *L8* (shown in Table 2). Frames, that are generating some strategies which are very neutral are, not surprisingly, “neutral” time-frames *N1*, *N2*, *N6*. They would be profitable if we would ignore transactional costs. Interesting time-frame is *H2* – it produces profitable strategies but only for growing market. Obviously we wouldn’t use such strategies because of heavy (but a bit lower than wrong *buy&hold* strategy) losses generated by those on falling market.

We have run a “long term” test on a 05-01-2007 - 01-05-2007 time-frame. Results for *buy&hold* on this time-frame would be 549 points, and for *sell&hold* gives -553 points.

Results for a strategy learned on *L8* are: the average profit equals 324 points (including provision: 178 points). The profit gained by transaction balancing equals 191 points and combined strategy gained 280.

Results are not as good as one could expect, but there are some important things to mention - we have learned our EA on a falling time-frame, and it’s a rising one. If we would just sell at the beginning and buy at the end, we would lose 553 points and we have earned 280 points which is over a half of 549 points earned by a *buy&hold* strategy.

Also, we have tested *H2* combined strategy on the same time-frame. *H2* has gained 650 points, which is better than *buy&hold*. It’s not surprising, it’s a rising time-frame and it was thought on a rising time-frame too. What’s interesting it’s the fact, that it has made some profitable short trades (3 out of 4). It has proven to detect some local heights.

To summarize the results, our approach allows generating profitable strategies if we consider profits from more than 20 test time-frames. It was impossible to obtain a strategy, that was always gaining money independent of the actual time-frame, but when adding profits and losses from many time-frames, the results are definitely positive. It's important to choose a right learning time-frame – not each of them contains enough information to learn. Choosing combined strategy is nearly always a good idea, allowing us to open just one contract and still profit as an average strategy, without the risk connected with arbitrarily chosen strategy from all generated, which could possibly result in a very poor behavior (see *Min* column in Table IV or Table V). The most important – it's possible to trade with our strategies without human intervention and generate gains.

V. Further work

The matter of risk of our automatically generated strategies was ignored in this work, so as the capital management. In further studies they need additional research, because without those it cannot be used in any long term trading system, were not only random, high capital losses are not allowed, but also the amount of opened positions must vary in a way that uses the invested capital optimally. In future market such as Forex without proper capital management a so called “margin call” can happen, which means that in case of losses an additional deposit of cash must be made, or the position will be closed and (nearly) all the money lost.

Another case that needs more attention is so called “take-profit” (*tp*) and “stop-loss” (*sl*) orders. They were ignored in this work, but they are widely used by most of the investors, especially on Forex market. They allow us to profit from very short moments of euphoria on the market, where the price peaks only for seconds (*tp*) or to save our capital if we want to limit our losses on single transaction (*sl*). We could possibly allow the genetic algorithm to set this orders to either fixed values (for example 100 points from the transaction price) or values constantly adjusted by decision trees.

VI. Conclusion

We think that although our automating generated strategies are still too unstable and unreliable to use them in real trading systems, they can be used as an additional help for an experienced investor. Especially in the form of additional indicator, which shows in real time how many strategies are for and how many against our decisions. If we combine strategies from rising and falling learning time-frames our indicator is even more interesting, revealing us the knowledge of hundreds of automatically generated strategies in a very accessible form (which needs some practice though). However, reliability of such an indicator still needs further studies. We are planing to verify our method results comparing with another methods on the same data.

References

[1] Allen F., Karjalainen R., “Using genetic algorithms to find technical trading rules”, Journal of Financial Econ. 51(2), pp.245-271, 1999.

[2] Atsalakis G. S., Valavanis K. P., "Surveying stock market forecasting techniques - Part II: Soft computing methods", Expert Systems with Applications: An Inter. Journal Arch 36 (3), pp. 5932-5941, 2009.

[3] Bank for international settlements, “Foreign exchange and derivatives market activity in 2007”, Triennial Central Bank Survey of Foreign Exchange and Derivatives Market Activity, 2007.

[4] Chen Y. , Mabu S., Shimada K., Hirasawa K., “A genetic network programming with learning approach for enhanced stock trading model”, Expert Systems with App., 36(10), pp. 12537-12546, 2009

[5] Deng J.-L.,“Control problems of grey systems”, Systems & Control Letters, 1(5), pp. 288-294, 1982

[6] FOREX historical data for EUR/USD pair http://www.forexrate.co.uk/forexhistoricaldata.php

[7] FOREX trading hours http://www.gftforex.com/forex/forex-trading-hours.asp

[8] Kim H., Shin K., “A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets”, Applied Soft Computing, v.7 n.2, p.569-576, 2007

[9] Korczak J., Roger P., “Stock Timing using Genetic Algorithms”, [in:] Jour. of Stoch. Models in Business and Indust., 18, pp.121-134, 2002.

[10] Korczak, J., Lipinski, P., Roger, P., "Evolution Strategy in Portfolio Optimization", Artificial Evolution, ed. P. Collet, LNCS 2310, Springer, 2002, pp.156-167.

[11] Koza J. R., "Genetic Programming: On the Programming of Computers by Means of Natural Selection", MIT Press, 1992

[12] Kucharski A., „Algorytmy genetyczne w prognozowaniu danych giełdowych - usuwanie obserwacji nietypowych”, Badania Operacyjne i Decyzje, pp. 35-45, 1/2008 (*in polish*).

[13] Lin L., Cao L., Wang J., Zhang C., “The Applications of Genetic Algorithms in Stock Market Data Mining Optimisation”, Zanasi, A., Ebecken, N. F. F. (eds.) Data Mining V. WIT Press (2004).

[14] Murphy J., “Technical Analysis of the Financial Markets” , 1999.

[15] Myszkowski P. B., Rachwalski Ł., „Trading rule discovery in Warsaw Stock Exchange using coevolutionary algorithms”, 4th Inter. Symp. Advances in AI and App., Mrągowo (Poland), pp.81-88, 2009.

[16] Neely C., Weller P., Dittmar R., “Is Technical Analysis in the Foreign Exchange Market Profitable? A Genetic Programming Approach”, Jour. of Finan. and Quantitative Analysis 32 (4), pp. 405–426, 1997.

[17] Pei-Chann C., Chen-Hao L., Jun-Lin L., Chin-Yuan F., Celeste S.P. Ng, "A neural network with a case based dynamic window for stock trading prediction", Expert Syst. with App. 36, pp.6889-6898, 2009.

[18] Thomas J., Sycara K., “The Importance of Simplicity and Validation in GP for Data Mining in Financial Data”, AAAI Press, 1999.

[19] Wang Y.-F., “Predicting stock price using fuzzy grey prediction system”. Experts Systems with Applications. V22. 33-39, 2002

[20] Witkowska D., Marcinkiewicz E., "Construction and Evaluation of Trading Systems: Warsaw Index Futures", Inter. Advances in Econ.Research 11 (1), pp. 83-92, 2005.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Emotion-based Image Retrieval—an Artificial Neural Network Approach

Katarzyna Agnieszka Olkiewicz
Institute of Informatics
Wroclaw University of Technology
Wroclaw Wyb. Wyspianskiego 27, Poland
157627@student.pwr.wroc.pl

Urszula Markowska-Kaczmar
Institute of Informatics
Wroclaw University of Technology
Wroclaw Wyb. Wyspianskiego 27, Poland
Urszula.Markowska-Kaczmar@pwr.wroc.pl

***Abstract*—Human emotions can provide an essential clue in searching images in an image database. The paper presents our approach to content based image retrieval systems which takes into account its emotional content. The goal of the research presented in this paper is to examine possibilities of use of an artificial neural network for labeling images with emotional keywords based on visual features only and examine an influence of used emotion filter on process of similar images retrieval. The performed experiments have shown that use of the emotion filter increases performance of the system for around 10 percent. points**

***Index Terms*—Artificial neural network, feature selection, similarity measures, emotion recognition, image retrieval, relevance feedback.**

I. Introduction

IN RECENT years an increase of computer storage capacity and Internet resources can be observed. Fast development of new image and video technologies and easy access to sophisticated forms of information demand constantly improving searching and processing tools. Existing methods of text documents retrieval give satisfying results, so now research is focused on images retrieval. Finding the right set of images in a base containing thousands of them is still a challenging task. Few working methods were created and developed to solve the issue. The first category of approaches is based on textual annotations. It assumes that every image in the database has a label describing its content. Systems, which use only annotations, are nothing more than text-based searchers.

Another way of dealing with the same problem is based on observation that textual labels are not always available. Content based image retrieval (CBIR) systems assume that many features useful during searching process can be extracted from the image itself. In the approach looking for similar images may be reduced to measuring a visual distance between them. Many of the systems use color information; as an example we can point the paper [1], where authors created images retrieval system based on color-spatial information. The main difference between both approaches is the type of similarity they can find. Textual searchers are capable to find semantic similarity, also named similarity of ideas (for example tiger in summer and tiger in winter) and content based searchers return visually similar images, even if they present different ideas.

CBIR systems look for similar images, but criteria of similarity are not explicitly defined. They can take into account image coloring, objects included in it, its category (for instance *outside* or *inside*) or its emotion (also called mood or feeling). The last one, depending on interpretation, can be seen as emotional content of a picture itself or an impression it makes on a human. In the paper we consider both definitions as equivalent. These systems are called EBIR (Emotion Based Image Retrieval) and they are a subcategory of CBIR ones. The term EBIR was introduced in the paper [2].

The most of research in the area is focused on assigning image mood on the basis of of eyes and lips arrangement, because the studies concentrate on images containing faces. In the current version of our research we assumed that emotional content is characterized by image coloristic, texture and objects represented by edges, and the information can be used in similar images retrieval process. An extension of this list can contain faces or other objects and symbols which can have an influence on the image affect.

When talking about emotions, we can not skip two important topics: subjectivity and the emotion classification. As stated in the paper [3], different emotions can appear in a subject while looking at the same picture, depending on a person and its current emotional state. But what we are looking for is not a system perfectly matching images and emotions. Our far reaching aim is to build a system, which can in an effective way support a searching process and increase a number of relevant pictures returned by any given query. The goal of the research presented in this paper is to examine possibilities of use of an artificial neural network for labeling images with emotional keywords based on visual features only and examine an influence of used emotion filter on process of similar images retrieval. Advantages of such approach is easiness adjustment to any kind of pictures and emotional preferences. Neural networks are machine learning techniques well known because of their noise resistance, which is very desirable feature in this application.

The paper is organized as follows: in the section II various approaches to image emotional content recognition described in a domain literature are presented. In the section III a general overview of the system is presented, together with a description of used visual descriptors and measurement of the image similarity. The constructed neural network is presented and a note about image databases used for learning and testing is added. In the section IV results of performed experiments are presented and an analysis of the results is given. Finally, in the section V, a conclusion and further work directions are proposed.

II. Related works

Broadly speaking there are three main methods of acquiring emotional information from pictures: labels' analysis, face expression's analysis and visual content analysis. The first method is based on textual descriptions of pictures and dictionaries of emotional terms. An example of such approach is presented in the paper [4]. The second method is used only to find emotions in pictures of human face and further applied for example in human-robot interactions. Analysis of faces are presented in the paper [5]. The last method assumes no information about pictures. Extraction of visual features is based only on properties like color and texture. The method was implemented in some systems, for example in the one presented in the paper [6].

A problem connected with EBIR systems is connected to sets of emotions considered by their authors. Many classifications of emotions exist; that is why it is difficult to compare them. The simplest set, presented in the paper [7], contains positive-negative categories. In [4] the basic emotion set is as follows: happiness, sadness, anger, fear and disgust. In the paper [5] surprise has been added to the above set. Authors of the paper [8] removed disgust from the set, but added neutral emotion and hate.

Another way of classification of images is based on adjectives describing more objective attributes of a picture, like a warm-cold, static-dynamic, heavy-light set, presented in [6]. Authors of the paper [9] developed the concept and created the following set: exhilarated-depressive, warm-cool, happy-sad, light-heavy, hard-soft, brilliant-gloomy, lively-tedious, magnificent-modest, vibrant-desolate, showy-elegant, clear-fuzzy, fanciful-realistic. Some other proposals are: Kobayashi's words (used for example in the paper [2]) and space of valence-arousal-control describing emotions, presented in the paper [3].

Let us recall that for learning rules of matching visual features to emotions some solutions were also developed. The most common are: regression [9], neural networks [5] [8] [10] and genetic algorithms [10]. Our system does not use any rules for classification; it is not a hybrid system also.

III. NECR – Neural-based Emotional Content Retrieval system

As we have mentioned above, the research investigates the feasibility of use of visual features for the retrieval of emotional content of images and tests feasibility of training ANN to accomplish classification task. To achieve this goal, a prototype system has been designed and implemented. The next subsection presents an idea of our approach.

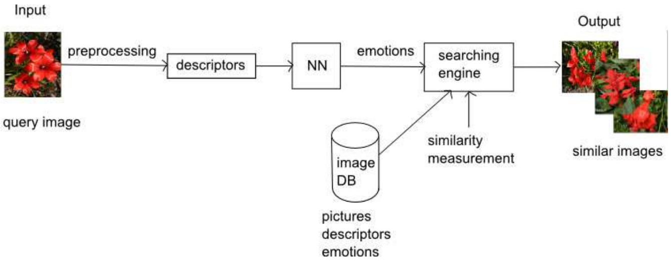


Fig. 1. Schema of the system

A. Idea

A general idea of the system is presented in Fig. 1. The system consists of a database of images, neural network, searching engine and interface to communicate with a user. All images in the database need to be preprocessed in order to find their visual feature descriptors, which refer to coloristic, texture and edges in pictures. We assume that the system is able to recognize an emotional content of images on the basis of classification method. Classification is performed by a supervised trained neural network. A learning set for the network was prepared manually, by assigning class labels to images from the database.

In our system in order to test an influence of the visual feature descriptors on an ability to recognize the emotional content of images and to find similar images, we have considered three various groups of emotion classification:

- positive-negative with neutral option,
- groups of adjectives:
  - warm, cold, neutral,
  - dynamic, static, neutral,
  - heavy, light, neutral,
  - artificial, natural; to distinguish between photos and hand-made pictures,
- 5 basic emotions (happiness, sadness, anger, disgust and fear).

After the training process the neural network is ready to assign emotions to pictures; one emotion from each category, what makes 6 labels for each picture. However, before any classification can take place, images need to be preprocessed. As a result of this step, visual descriptors are calculated and stored in the database, together with pictures. The network uses values of descriptors in classification process, and assigned labels are also stored in the database. The first stage of system's work is presented in Fig. 2.

Searching engine takes information about the pictures from the database and about the query image, calculated on an ongoing basis. As a result of the engine's work, 12 the most similar images are returned. The user can accept results or run the program again, with a modified query. The new query contains of an original picture and these of returned 12, which the user has marked as appropriate. The process can

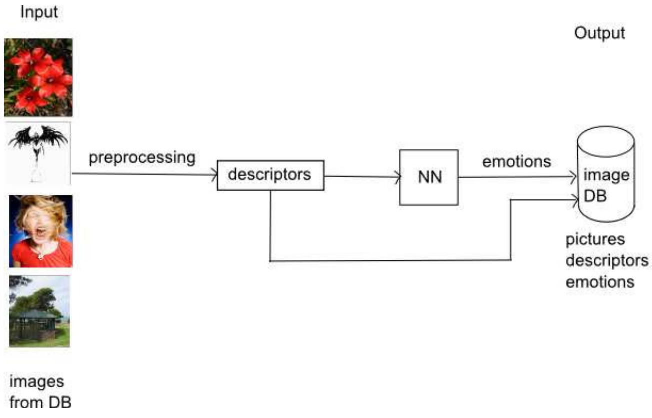


Fig. 2. Preparation of data

be repeated many times if needed. In a multi-images query, for each query image the most similar pictures are found and then a common list is built, as an average of distances between query images and images from the database.

Visual descriptors are calculated and emotional classification is made only once for each database; it means that if a user does not change the database, the program will run much faster. Because a query image can be of any kind, descriptors for it are always calculated, even if the picture belongs to the database. There is no option of retraining the network in the program.

B. Visual descriptors

Extracting information from a picture is a challenging task. Descriptors need to meet performance, reliability and accuracy criteria. Standard MPEG-7 defines some descriptors, which can be used for similar images retrieval (from the Internet article [11]). Some of the proposed there descriptors were used already in image retrieval systems [6]. They allow acquiring information about colors, edges and textures. In the system, three of them are used: Edge Histogram, Scalable Color Descriptor and Color Layout Descriptor. We base on implementations published in [12]. Additionally, two commonly available custom descriptors are used: CEDD and FCTH (described in [13] [14] [15]). They combine information about colors and edges or textures respectively.

Edge Histogram returns 80 numbers representing quantity of edges: 16 regions x 5 directions of edges (vertical, horizontal, 2 diagonals and without direction). We added global number of edges for each category to let the network to easily label pictures with dominating edges direction. Scalable Color Descriptor divides color space into 256 colors and calculates percentage of a picture covered with that color. Color Layout Descriptor divides picture into 64 regions and chooses a dominant color for each region. It allows us to obtain spatial-color information. CEDD (Color and Edge Directivity Descriptor) divides a picture into 1600 regions. 144 numbers are obtained as count of regions for each combination of 24 colors and 6 types of edges. FCTH (Fuzzy Color and Texture Histogram) works similar as CEDD, but in place of 6 categories of edges, it uses 8 categories of textures, what gives 192 numbers representing each picture.

The purpose of using so many descriptors is to acquire as much information about a picture as possible and as a result - to train the network efficiently. Of course, balance between amount of information collected by the system and processing time has to be found.

C. Neural network

The multi-layered perceptron neural network is used for emotional image classification on the basis of its visual descriptors, because it is universal, easy to construct and it performs well. Neural networks can distinguish between very similar input vectors and are immune to redundant or noisy information. We wanted to make classification of input images as consistent as possible, but it is not possible to judge few thousands of pictures in the same way. There is no theoretical model matching visual content of a picture to its emotional content. Neural networks have the ability to find schemas and rules even in such extreme environment.

After the preprocessing stage every image from the base is represented by its visual features vector **v**. The first elements of the vector **v** refer to SCD, the next to CLD, EH and the last two to CEDD and FCTH. In other words, for *i*-th image in database vector $\mathbf{v}^i$ is composed of 5 component vectors (eq.1)

$$\mathbf{v}^i = [\mathbf{v}_{SCD}, \mathbf{v}_{CLD}, \mathbf{v}_{EH}, \mathbf{v}_{CEDD}, \mathbf{v}_{FCTH}] \tag{1}$$

The query image is processed in the same way and is also described by its visual features vector $\mathbf{v}^q$.

Vector **v** is an input for the neural network. Its length is equal 869, so the number of inputs of neural network is also equal 869. It is worth pointing out that values of each element in vector **v** are scaled in the range (0-1). In the output layer we have 19 neurons. They encode 18 different emotions belonging to 6 categories. An answer of the output neuron equals to 1 indicates presence of a particular emotion. Only one emotion from each of 6 sets given in the section III-A can be present, so from all output neurons representing a category the one with the highest activation is chosen and its value is set to 1. For all others within the same category 0 is set.

The network contains three layers: input, hidden and output. All output neurons are connected with all hidden ones; 128 hidden neurons are connected with input ones in a way allowing better feature and pattern recognition. It means that hidden neurons are responsible for discovering only one feature. The schema of the network is presented in Fig. 3. For clarity reasons, only one set of connections between hidden and output neurons is shown.

It is visible that hidden neurons have their unique role in the classification process and are responsible for detecting only one kind of feature. Such specialized structure of the network was inspired by authors of the paper [16]. Because of limited set of connections between input and hidden layers (the network is not fully connected), learning process takes considerably less time. More complex structures with two hidden layers or more hidden neurons in already existing layer

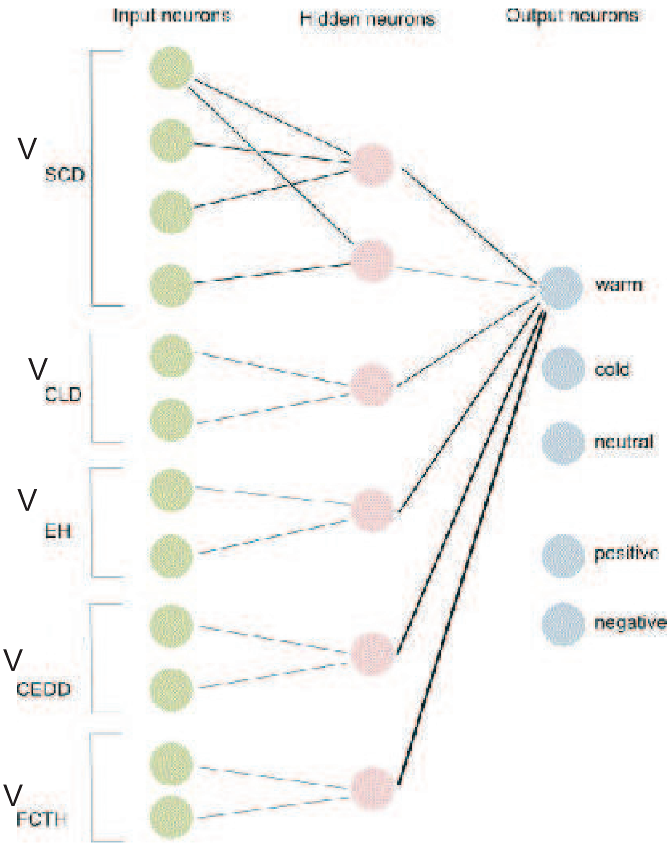


Fig. 3. Schema of the network. Only one set of connections between hidden and output neurons is shown

were considered as well. But, with concern about speed of images' classification and retrieval processes, we decided to use a simpler model.

After processing by the neural network each i-th image is represented by two vectors: vector of visual descriptors $\mathbf{v}^i$ and vector of emotions $\mathbf{e}^i$.

D. Similarity of images

To measure similarity between a query image and i-th image in the database, the distance between them is calculated. In some experiments we take only visual similarity, in other experiments we take both visual and emotional similarities (both vectors $\mathbf{v}$ and $\mathbf{e}$ were considered in this case). Let us focus on vector $\mathbf{v}$ first. The distance is separately assigned for each component vector $\mathbf{v}_{SCD}$, $\mathbf{v}_{CLD}$, $\mathbf{v}_{EH}$, $\mathbf{v}_{CEDD}$ and $\mathbf{v}_{FCTH}$. It is weighted and summed as in eq. 2.

$$d' = w_{SCD} \cdot d_{SCD} + w_{CLD} \cdot d_{CLD} + w_{EH} \cdot d_{EH} + \\ + w_{CEDD} \cdot d_{CEDD} + w_{FCTH} \cdot d_{FCTH} \quad (2)$$

Where w with an index denotes a weight of a given part of a distance component. The final distance d between query image and i-th image in the base is a weighted average. It is expressed by eq. 3.

$$d = \frac{d'}{w_{SCD} + w_{EH} + w_{CLD} + w_{CEDD} + w_{FCTH}} \quad (3)$$

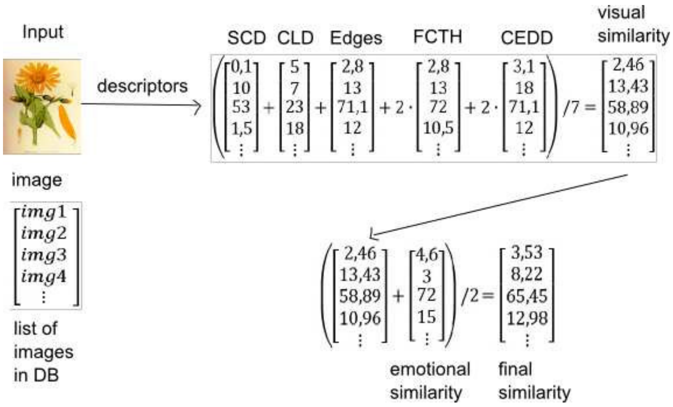


Fig. 4. An example of calculation of distance between a query images and images from the database

The way of distance computation was inspired by the paper [15], where the detailed description of the method can be found. To measure the distance on the basis of the part $\mathbf{v}_{CLD}$ the method was modified to deal with the three values referring to the three components of a color. The distance is transformed into the range (0-100). In particular 0 means the same image. Fig. 4 shows an example of visual distance calculation between a query image and each of images in the database. For the query image the similarity vector to each image in the base is obtained.

In the performed experiments weights w_{FCTH} and w_{CEDD} were set to 2, because these descriptors have the best individual retrieval scores. Remaining weights were equal to 1.

The second component in evaluation of images similarity takes into account emotional aspect and is based on the vector $\mathbf{e}$. For every matching label, 1 is added to a temporal result and then the final number is casted on the range 0-100, with 0 denoting maximal similarity. The query image is described by a vector of emotional similarities to each database image. Finally, both results (visual and emotional) are added and divided by 2. This is the final answer of the system. Whole method is illustrated by Fig. 4.

In a case with multiple query images, an average from all rankings is taken. Twelve images from the database with the smallest values are presented to the user. A case with multiple query images is presented in Fig. 5.

IV. Experimental study

To evaluate performance of our system and effectiveness of the similar images retrieval method, we performed some experiments. We assessed performance of the neural network (correct emotions assignment) and accuracy of retrieval results independently, with concern about various factors which can influence the performance.

The testing set in these experiments consists of 42 images, labeled manually and checked for consistency with labels given by the network. We tested the network trained on two different learning sets and we compared results. Details are

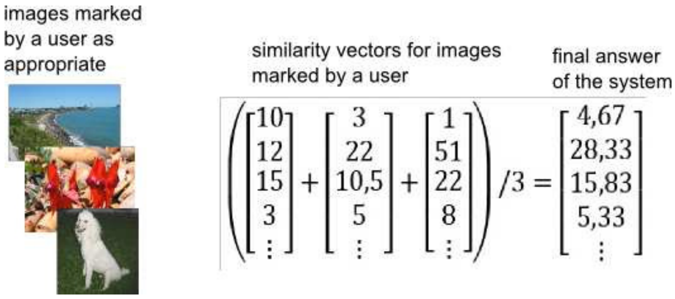

images marked by a user as appropriate
similarity vectors for images marked by a user
final answer of the system
$\left(\begin{bmatrix}10\\12\\15\\3\\\vdots\end{bmatrix}+\begin{bmatrix}3\\22\\10,5\\5\\\vdots\end{bmatrix}+\begin{bmatrix}1\\51\\22\\8\\\vdots\end{bmatrix}\right)/3=\begin{bmatrix}4,67\\28,33\\15,83\\5,33\\\vdots\end{bmatrix}$

Fig. 5. An example of finding similar images to a multiple query

TABLE I
CROSS-VALIDATION TESTS FOR THE NEURAL NETWORK

Subject	Accuracy	Deviation
Percent of correctly assigned (CA) labels	64.4	2.15
Percent of CA labels for warm-cold	80	2.32
Percent of CA labels for light-heavy	62.4	4.03
Percent of CA labels for dynamic-static	67.6	6.15
Percent of CA labels for artificial-natural	82	3.6
Percent of CA labels for positive-negative	55	4.1
Percent of CA labels for basic emotions	52	4.26

presented in subsection IV-A. We also did cross-validation tests.

The second part of these tests, dedicated to overall system performance analysis is more complex. We tested the system against many factors: various query images, image databases, learning sets and finally we evaluated difference in performance given by an emotions recognition module. Details are presented in the following subsections.

A. Datasets

Few image sets were created for learning and testing purposes. Because the system is supposed to support emotion based image retrieval, construction of sets was made with high consideration of an emotional content of pictures, especially for learning sets creation. The images in learning set were selected in a way which provides a fair representation of variously labeled pictures (the learning set consists of pictures labeled by every emotion from the set of 18 emotions). Fig. 6 presents the number of representative images in LS3 belonging to the particular emotions' categories.

First learning set (LS1) was intended to support good distinction between warm-cold, heavy-light and positive-negative categories and it consists of 893 pictures. It contains mainly landscape pictures, so expressing dynamism or anger is not possible there. Second learning set (LS2) was intended to support these categories, which are not supported in the first one: basic emotions, dynamic-static and artificial-natural and is built from 636 images. It contains images returned by searching engine like Flickr and Google for emotional keywords queries. But, the neural network trained on this set can not classify correctly any general images (for example landscapes), so third one (LS3) was made from 1456 pictures. It contains pictures from previous two sets, to support all classifications.

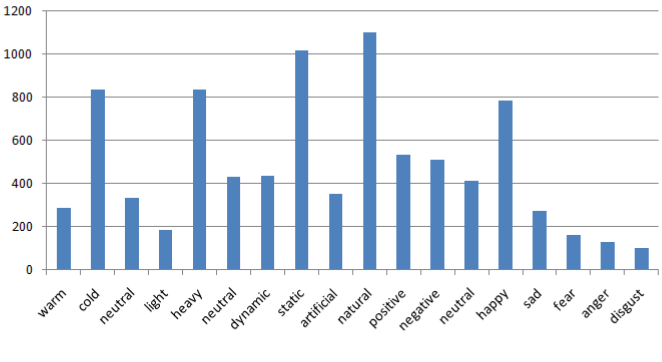


Fig. 6. Number of representatives of emotions in LS3

Three image sets are used in experiments, to evaluate performance of the system. All of them contain various pictures, belonging to different categories. We tried to balance quantity of representatives of every category. The first set (DB1) contains 2096 images, mostly landscapes. The second set (DB2) contains 1456 images, mostly emotionally rich and artificial ones. The third set (DB3) contains 1612 images, mostly natural ones and photos of people.

B. Evaluation of neural network performance

The network was trained with back-propagation method. The following values of parameters were set: learning rate 0.1, number of epochs 500, momentum 0.6, sigmoid unipolar activation function and error tolerance 0.1. For every learning set the network is trained only once and after that it is used in experiments.

Performance of the neural network was checked in two independent tests: by 5-cross-validation method and on a testing set of images different from learning sets. Cross-validation was performed with use of LS3 data set. The results are presented in Table I.

It is visible that performance of the network depends heavily on subsets chosen for learning and testing (the standard deviation can be as high as 6.15). But high classification score for one category has its drawback - lower scores for other categories: the network trained on the 3rd subset classified correctly 78% of pictures according to dynamic-static category had lower classification score for all other categories.

To determine performance of the network in an unknown environment, 42 different from learning sets pictures were chosen and classified by the network. Then, an automatic classification was compared with a manual one and results are shown in Table II.

In the test the learning sets LS1 and LS3 were used. The learning set LS2 was build only from pictures returned as results for emotional keywords queries and a network trained on it would not be able to determine a category of emotion properly 1-4 (rows 4-7 in Table II).

TABLE II
COMPARISON OF PERFORMANCE OF THE NEURAL NETWORK TRAINED WITH USE OF 2 TRAINING SETS

	Subject	Set LS1	Set LS3
1	Percent of correctly classified images	8	17
2	Percent of images with 1 wrong label	22	37
3	*Percent of correctly assigned (CA) labels*	*64*	*73*
4	Percent of CA labels for warm-cold	78	87
5	Percent of CA labels for light-heavy	62	74
6	Percent of CA labels for dynamic-static	70	69
7	Percent of CA labels for artificial-natural	70	83
8	Percent of CA labels for positive-negative	51	64
9	Percent of CA labels for basic emotions	49	60

Percentage of correctly assigned labels is used as measurement of system's efficiency because more common measures like recall and precision can not be used here. The system has to return 12 pictures in every run, so there is no possibility to define a set of false positives (even if some pictures score less than others, they are still present in results as complement to true positives). Moreover, if more than 12 images in the database are similar to the query image, the system has no possibility to show them all as a result.

As it can be seen in Table II, the network trained on a more general learning set (LS3) performs better than the one trained on less general one (LS1). The most problematic categories are basic emotions and positive-negative. It proves that emotional content of pictures can not be fully expressed only with chosen by us visual descriptors.

The network was trained two times on learning set LS3 (starting from random values of weights) and answers of the network from both trials were compared. Only in 17% of cases both networks were wrong and most of these mistakes were connected to basic emotions, which were not possible to be discovered without semantic knowledge about the picture. In 20% of cases one of the networks was wrong.

In most cases a network trained on the whole set LS3 performed better than the one trained on 80% of the set, even though test pictures here differed more than in the previous experiment. For dynamic-static, artificial-natural and positive-negative categories some subsets from the previous experiments scored higher than the network in the current one (trained on the whole set LS3). It can be explained in two ways: test images in the second experiment were more difficult to be classified and random division of the 3rd set favored different categories in different subsets.

C. Different image sets

Three different sets of pictures (DB1, DB2 and DB3) were created in order to test retrieval performance of the system. Results of experiments are presented in Table III. We are interested in number of runs (queries) needed to find all similar images from the sets. Three numbers, separated by commas, in every cell denote three sets. The network trained on the third learning set was used in the section.

TABLE III
PERFORMANCE OF THE SYSTEM AGAINST DIFFERENT QUERIES AND SETS. THREE NUMBERS, SEPARATED BY COMMAS, IN EVERY CELL DENOTE RESULTS REFERRING TO THREE SETS

Picture	N_{sr}	N_{pr}	N_{Runs}
black-white	2, 2, 3	2, 2, 1	1, 1, 1
red flower	10, 4, 10	5, 1, 5	5, 2, 4
lagoon, mountain	4, 4, 5	4, 4, 1	1, 2, 1
tropical forest	9, 11, 6	3, 4, 3	1, 3, 2
iceberg	8, 8, 2	6, 7, 0	2, 2, 0
sunset	12, 15, 5	10, 12, 4	4, 7, 1
red, shouting man	1, 6, 1	1, 6, 1	1, 1, 1
grey-scale	2, 7,-	1, 2, -	1, 1, -
worm	-, 6, -	-, 3, -	-, 2, -
boxing fight	-, 7, -	-, 6, -	-, 2, -

In Table III N_{sr} refers to the number of pictures in the set, which are similar to the query image. N_{pr} refers to the number of relevant pictures returned by the system and N_{Runs} refers to the number of searching trials the system had to perform to retrieve such results. Three numbers separated by commas in every cell denote results for every set: the first number refers to DB1, the second to DB2 and the third to DB3.

Some problems are shown here: color quantization and difficulty in finding precisely described set in hundreds of very similar pictures. Still, characteristic images are easy to find and overall results are very good. In many cases one query is enough to find the whole set, in others rerunning the program allows to receive better results. Images containing worms and boxing fights were present only in one set, so for others "-" is placed in Table III. The set DB3 contains pictures similar semantically to query images, but not visually, that is why retrieval results are worse than for the other two sets.

D. Emotions' filter

Emotion filter is a tool which uses vector **e** to produce final similarity score between two pictures as shown in Fig. 4. Without it, only vector **v** is used. To evaluate an input of an emotion filter to the final result, the same tests as in the subsection IV-B were run, but without calculating the vector of emotional distance between pictures. Results are presented in Table IV.

It is clear that emotions are important in the image retrieval process and improve results of traditional CBIR systems. In the EBIR system, more adequate pictures are found and it is done faster. Moreover, it can be noticed that the number of not relevant images (for example green building returned for tropical forest query) decreases when emotions' filter was used. Quality of results is higher for the system with the filter, what supports our theory.

To evaluate influence of the emotional filter, we created a metrics of efficiency E, expressed by eq. 4.

$$E = \frac{\frac{N_{pr}}{N_{sr}}}{1 + 0.05 \cdot (N_{Runs} - 1)} \cdot 100\% \tag{4}$$

TABLE IV
PERFORMANCE OF THE SYSTEM WITHOUT EMOTIONS' FILTER. THREE NUMBERS, SEPARATED BY COMMAS, IN EVERY CELL DENOTE RESULTS REFERRING TO THREE SETS

Picture	N_{sr}	N_{pr}	N_{Runs}
black-white	2, 2, 3	2, 2, 0	1, 1, 0
red flower	10, 4, 10	4, 0, 6	8, 0, 6
lagoon, mountain	4, 4, 5	4, 4, 4	3, 2, 5
tropical forest	9, 11, 6	3, 4, 0	1, 3, 0
iceberg	8, 8, 2	6, 7, 0	2, 2, 0
sunset	12, 15, 5	6, 6, 4	3, 3, 1
red, shouting man	1, 6, 1	1, 6, 1	1, 1, 1
grey-scale	2, 7, -	0, 2, -	0, 2, -
worm	-, 6, -	-, 1, -	-, 1, -
boxing fight	-, 7, -	-, 5, -	-, 2, -

where:
N_{pr} – number of pictures returned,
N_{sr} – number of pictures that should be returned,
N_{Runs}– number of runs. This metrics describes accuracy in relation to the number of runs. In the case with use of emotion filter E equals to 71%, 67% and 47% for sets DB1, DB2 and DB3 respectively. In the case without emotions filter E is equal to 59%, 57% and 42% for the same sets. Average decrease in performance is 9 percent points. The biggest differences in performance for various pictures are 31 percent points for a worm, 27 percent points for a grey-scale image and 19 percent points for a sunset. A lagoon picture scored 12 percent points better without emotions filter, but it is the only exception.

Detailed comparison between the results presented in two tables is illustrated in Fig. 7. Further conclusions are given in the subsection IV-E. Comparison between Tables III and IV shows that decrease in quality of results for the case without emotions filter is 17% and speed decrease is equal to 17%. Additionally, in a case with use of emotions filter, only in two situations no similar images were retrieved, but in the case without the filter – five times.

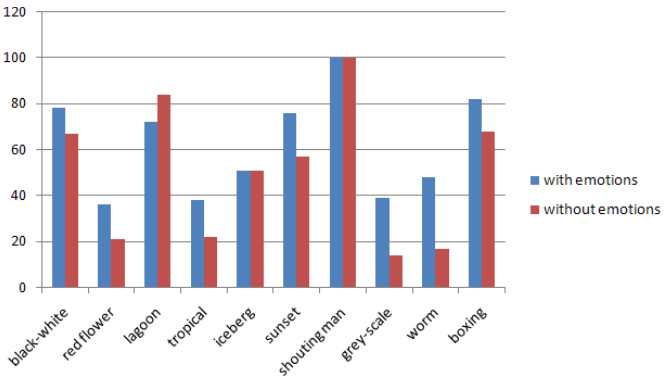


Fig. 7. Value of metrics E for different sets and pictures

TABLE V
PERFORMANCE OF THE SYSTEM AGAINST DIFFERENT LEARNING SETS

Picture	N_{sr}	N_{pr}	N_{Runs}
black-white	2	2, 2	1, 2
red flower	4	1, 0	2, 0
lagoon, mountain	4	4, 4	2, 1
tropical forest	11	4, 4	3, 3
iceberg	8	7, 7	2, 1
sunset	15	12, 7	7, 3
red, shouting man	6	6, 6	1, 2
grey-scale	7	2, 2	1, 1
worm	6	3, 1	2, 1
boxing fight	7	6, 6	2, 3

E. *Different learning sets*

Two learning sets were tested here: LS1 and LS3. Retrieval performance was checked in the same way as in previous sections (but only the DB2 set was used). Here numbers in cells denotes two learning sets. The first number belongs to the third set and the second one to the first set. Results can be found in Table V.

It can be seen that learning set influences retrieval results, so it should be chosen with high consideration about databases with which it will work or, in case when a working environment of the system is not known, learning set should be universal and should contain all kinds of pictures. Still, learning sets influence less overall system performance than lack of the emotion filter.

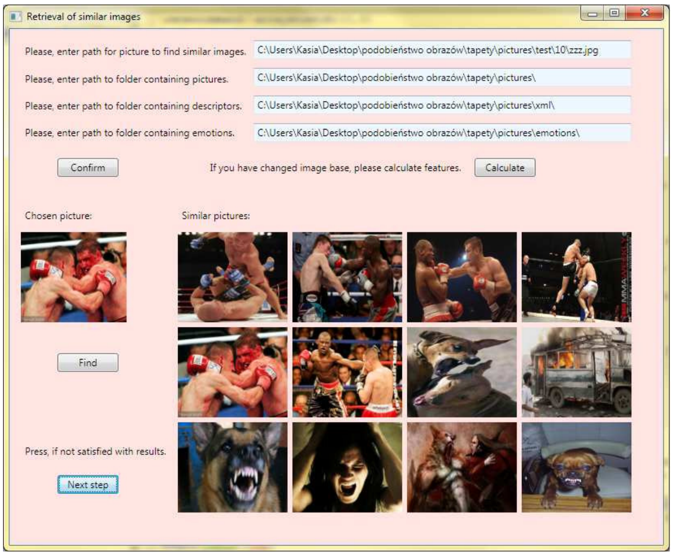
V. CONCLUSION

Our system is capable of finding similar images in a database with relatively high accuracy. Use of the emotion filter increases performance of the system for around 10 percent points. Experiments showed that average retrieval rate depends on many factors: a database, a query image, number of similar images in the database and a training set of the neural network. Although a user not always receives satisfying results during the first run of the searching engine, in most cases, after few runs they are satisfying.

Interface of the application and results returned by the system for a query image (boxing fight) are presented in Fig. 8.

Further improvements to the system are considered. To increase accuracy of the results, a module for face detection and analyzing face expression can be added. More work is needed to develop the system in a way allowing it to analyze existing textual descriptions of images and other meta-data. More accurate and informative descriptors can be also created. Another idea is to build a system containing two or more neural networks and use them as an ensemble classifier.

To fully evaluate the results obtained with the neural network in future we plan to apply another classifier instead. Bayesian models, linear models, decision trees and K-NN methods are concerned.



Please, enter path for picture to find similar images. C:\Users\Kasia\Desktop\podobieństwo obrazów\tapety\pictures\test\10\zzz.jpg

Please, enter path to folder containing pictures. C:\Users\Kasia\Desktop\podobieństwo obrazów\tapety\pictures\

Please, enter path to folder containing descriptors. C:\Users\Kasia\Desktop\podobieństwo obrazów\tapety\pictures\xml\

Please, enter path to folder containing emotions. C:\Users\Kasia\Desktop\podobieństwo obrazów\tapety\pictures\emotions\

If you have changed image base, please calculate features.

Press, if not satisfied with results.

Fig. 8. An example of program's run

VI. Acknowledgement

This work is partially financed from the Ministry of Science and Higher Education Republic of Poland resources in 2008 2010 years as a Poland-Singapore joint research project 65/N–SINGAPORE/ 2007/0.

References

[1] Y. Jo and K. Um, "A signature representation and indexing scheme of color-spatial information for similar image retrieval," *IEEE Conference on Web Information Systems Engineering*, vol. 1, pp. 384–392, 2000.

[2] Y. Kim, Y. Shin, Y. Kim, E. Kim, and H. Shin, "Ebir: Emotion-based image retrieval," in *Digest of Technical Papers International Conference on Consumer Electronics*, 2009, pp. 1–2.

[3] A. Hanjalic, "Extracting moods from pictures and sound," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 90–100, 2006.

[4] S. Schmidt and W. G. Stock, "Collective indexing of emotions in images. a study in emotional information retrieval," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 5, 2009.

[5] F. Siraj, N. Yusoff, and L. Kee, "Emotion classification using neural network," in *International Conference on Computing & Informatics*, 2006, pp. 1–7.

[6] E.-Y. Park and Y.-W. Lee, "Emotion-based image retrieval using multiple-queries and consistency feedback," in *6th IEEE International Conference on Industrial Informatics*, 2008.

[7] Q. Zhang and M. Lee, "Emotion recognition in natural scene images based on brain activity and gist," in *IEEE World Congress on Computational Intelligence*, June 2008.

[8] Y. Guo and H. Gao, "Emotion recognition system in images based on fuzzy neural network and HMM," in *5th IEEE International Conference on Cognitive Informatics*, 2006, pp. 73–78.

[9] W. Wang, Y. Yu, and S. Jiang, "Image retrieval by emotional semantics: a study of emotional space and feature extraction," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, 2006, pp. 3534–3539.

[10] Y. Sun, Z. Li, and C. Tang, "An evolving neural network for authentic emotion classification," in *5th International Conference on Natural Computation*, 2009, pp. 109–113.

[11] (2010) Standard mpeg-7. [Online]. Available: http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm

[12] (2010) Implementation of visual desciptors descrobed in standard mpeg-7. [Online]. Available: http://savvash.blogspot.com/2007/10/here-acm-multimedia-2007.html

[13] S. Chatzichristofis and Y. Boutalis, "Cedd: Color and edge directivity descriptor - a compact descriptor for image indexing and retrieval," in *6th Interntional Conference in Advanced Research on Computer Vision Systems*, 2008.

[14] S. Chatzichristofis and B. Yiannis, "Fcth: Fuzzy color and texture histogram, a low level feature for accurate image retrieval," *Ninth International Workshop on In Image Analysis for Multimedia Interactive Services*, pp. 191–196, 2008.

[15] (2010) Implementation of descriptors cedd and fcth. [Online]. Available: http://savvash.blogspot.com/2008/05/cedd-and-fcth-are-now-open.html

[16] H. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, 1998.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 97–104

ISBN 978-83-60810-27-9
ISSN 1896-7094

Automatic Visual Object Formation using Image Fragment Matching

Mariusz Paradowski
Wroclaw University of Technology
Institute of Informatics, Poland
Nanyang Technological University
School of Computer Engineering, Singapore

Andrzej Śluzek
Nanyang Technological University
School of Computer Engineering, Singapore
Nicolaus Copernicus University
Faculty of Physics, Astronomy and Informatics, Poland

***Abstract*—Low–level vision approaches, such as local image features, are an important component of bottom–up machine vision solutions. They are able to effectively identify local visual similarities between fragments of underlying physical objects. Such vision approaches are used to build a learning system capable to form meaningful visual objects out of unlabelled collections of images. By capturing similar fragments of images, the underlying physical objects are extracted and their visual appearances are generalized. This leads to formation of visual objects, which (typically) represent specific underlying physical objects in a form of automatically extracted multiple template images.**

I. Introduction

Image understanding, although generally considered the highest level of machine vision applications, provides useful information for low–level image processing tasks. For example, noise removal, image segmentation, etc. can be performed more effectively if the presence of certain contents is known or assumed in the processed images. On the other hand, the results of low–level operations are usually indispensable to detect such contents in unknown images. This contradiction is apparently one of the most challenging issues in advanced applications of machine vision.

A similar, and equally challenging, problem exists in *content-based visual information retrieval* (CBVIR). Two operations can be relatively easily done for a given image, namely (1) image annotation (manually performed by a human) and (2) low–level feature extraction (performed automatically by dedicated algorithms). However, a *semantic gap* [1] exists between these two operations, i.e. it is usually very difficult to relate image features to semantically meaningful image tags.

The paper proposes a technique that handles the above contradictions from the low–level perspective. In general, we attempt to automatically build image semantics using purely visual characteristics of the images. The content of images is assumed unknown and random (although they may be collected from a certain "world"). The first step of the proposed approach, i.e. the automatic formation of *visual objects* [2] (which typically correspond to physical objects depicted in the images) is discussed in the paper in detail. Such *visual objects* are built as groups (clusters) of *prototypes* [2] containing multiple similar fragments identified in the available collection (database) of images. If the database is a representative description of the "world", we identify typical components and, thus, provide a certain level of understanding of that "world". The presented results are development of preliminary ideas highlighted in [3].

The similarity between image fragments is determined using sets of locally computed feature vectors, i.e. the proposed method falls into the *local* category. Each feature vector describes a small fragment of the image (usually an elliptical or circular *keypoint*). To calculate the keypoints, we use popular, widely discussed approaches such as *Harris-Affine* [4] and *SIFT* [5], [6].

Detection and clusterization of similar fragments in the image database is executed in four steps: (1) image pre-retrieval, (2) image fragment matching, (3) formation of *prototypes* from similar fragments, (4) formation of *visual objects* from prototypes. The first two steps are responsible for localization of similar image fragments within the database. Image pre–retrieval is introduced solely for a higher efficiency of the method, while image fragment matching is the key operation in finding similar image fragments.

In the remaining two steps, relations are established between similar image fragments identified in the first two steps. *Prototypes* are formed based on intersections of image fragments located within the same image. These multiple intersecting fragments come from matching of the same image with other images. The last step merges prototypes found in different images into *visual objects*.

The general idea of the proposed method is illustrated in Fig. 1 and detailed explanations of the underlying algorithms are provided in Section II.

Altogether, the automatic visual object formation is a grouping algorithm. Its input is a collection (database) of unlabelled images. The output is a set of groups (visual objects), where each group consists of image fragments that have been found mutually similar. The number of output visual objects is determined fully automatically based on the visual properties of database images.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

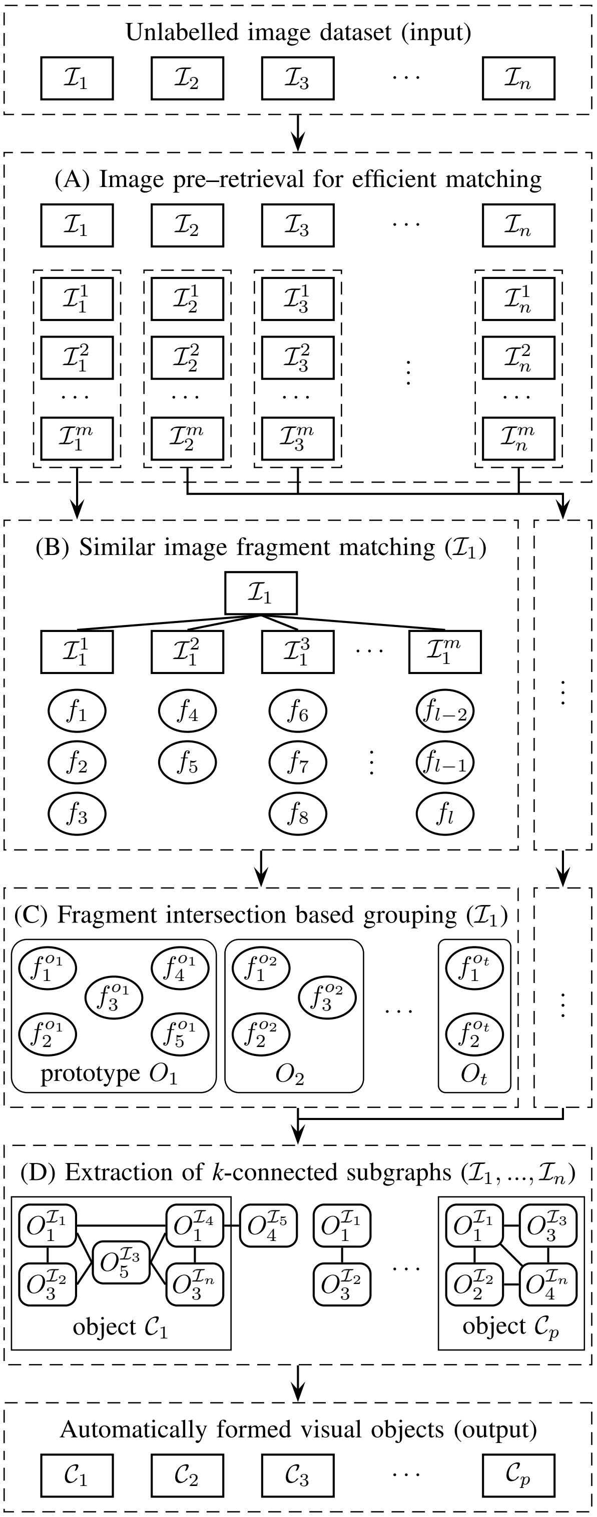

Unlabelled image dataset (input)
$\mathcal{I}_1$ $\mathcal{I}_2$ $\mathcal{I}_3$ ... $\mathcal{I}_n$
(A) Image pre–retrieval for efficient matching
$\mathcal{I}_1^1$ $\mathcal{I}_1^2$ ... $\mathcal{I}_1^m$
$\mathcal{I}_n^1$ $\mathcal{I}_n^2$ ... $\mathcal{I}_n^m$
(B) Similar image fragment matching ($\mathcal{I}_1$)
f_1 f_2 f_3 ... f_{l-2} f_{l-1} f_l
(C) Fragment intersection based grouping ($\mathcal{I}_1$)
prototype O_1
O_2
O_t
(D) Extraction of k-connected subgraphs ($\mathcal{I}_1,...,\mathcal{I}_n$)
object $\mathcal{C}_1$
object $\mathcal{C}_p$
Automatically formed visual objects (output)
$\mathcal{C}_1$ $\mathcal{C}_2$ $\mathcal{C}_3$ ... $\mathcal{C}_p$

Fig. 1. Steps of the proposed automatic visual object formation method; (A) image pre–retrieval to limit number of matches, (B) image fragment matching to extract similar fragments in different images, (C) formation of prototypes out of intersecting fragments in the same image, (D) formation of objects out of prototypes from different images.

II. Visual object formation

A. *Efficient image pre–retrieval (step A)*

Given a collection $\{\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_n\}$ of n unlabelled images, our objective is to identify the presence of similar fragments within these images. Therefore, for a collection of n images we have to match up to n^2 image pairs. This can be a time–consuming operation even for a small database. Given a pair of images $\mathcal{I}$ and $\mathcal{J}$, similar fragments are identified using an *image fragment matching* method (described in Section II-B). The results of such a matching are accurate, but the operation is computationally intensive. Therefore, we have introduced an efficient image pre–retrieval scheme so that the system can be applied to image collections of relatively large sizes. The proposed pre–retrieval mechanism is one of the novelties presented in this paper.

For a query image $\mathcal{I}_x : x \in \{1, ..., n\}$, we attempt to identify the most relevant (candidate) images $\{\mathcal{I}_x^1, \mathcal{I}_x^2, ..., \mathcal{I}_x^m\} : m \ll n$ from the whole database $\{\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_n\}$. Detection of fragments similar to unspecified fragments of $\mathcal{I}_x$ query is attempted within those candidate images only.

The candidate images are identified using a specialized *similarity function* $s(\mathcal{I}, \mathcal{J})$ defined for a pair of images $\mathcal{I}$ and $\mathcal{J}$. The similarity function is called *image topology similarity*. The function can be converted (if necessary) into an image–distance function using a variety of approaches. Unlike many classic image similarity measures, the proposed similarity function is able to identify images that are very different but contain similar fragments. The idea of the proposed function is conceptually similar to the previously proposed topological image fragment matching algorithm [7], i.e. it is based on pairs of matched keypoints. The main difference is that in here we use *weaker topological constrains*.

The proposed similarity function is defined as follows: First, we obtain matched pairs of keypoints $P(\mathcal{I}, \mathcal{J})$ between the input images (details to be discussed later). Having a non-empty set of such pairs P, we check the topological constrains for each keypoint pair $(p_I, p_J) \in P$. The topological constrains are verified within spatial neighbourhoods $\mathcal{N}(p_I)$ and $\mathcal{N}(p_J)$ of both keypoints from the pair. The spatial neighbourhood is defined as a set of r keypoints being the nearest neighbours in terms of image coordinates (Euclidean distance d) of a given keypoint. The neighbourhoods are found off–line and the nearest neighbours can be cached for each keypoint (and quickly retrieved on demand). Formally, the spatial neighbourhood $\mathcal{N}(p_X)$ for a keypoint p_X from an image (set of keypoints) $\mathcal{X}$ is equal to:

$$\mathcal{N}(p_X) = \arg\min_{N \in \mathcal{X}} \sum_{n \in N} d(n, p_X), \quad |N| = r. \tag{1}$$

Given $\mathcal{N}(p_I)$ and $\mathcal{N}(p_J)$ neighbourhoods of p_I and p_J keypoints, we check *how many* pairs of matched keypoints $P(\mathcal{I}, \mathcal{J})$ can be found within these neighbourhoods. The larger number of found keypoint pairs, the more credible (topologically) is the selected keypoint pair $(p_I, p_J) \in P$. We can now define a *topological verification function* $t(\mathcal{I}, \mathcal{J}, p_I, p_J)$ for a pair of keypoints $(p_I, p_J) \in P$ as the normalized number (r is the neighbourhood size) of matched pairs found in the neighbourhoods ($\times$ stands for Cartesian product):

$$t(\mathcal{I},\mathcal{J},p_I,p_J) = \frac{1}{r}\Big|\big[\mathcal{N}(p_I)\times\mathcal{N}(p_J)\big]\cap P(\mathcal{I},\mathcal{J})\Big|. \quad (2)$$

An illustrative example is shown in Fig. 2.

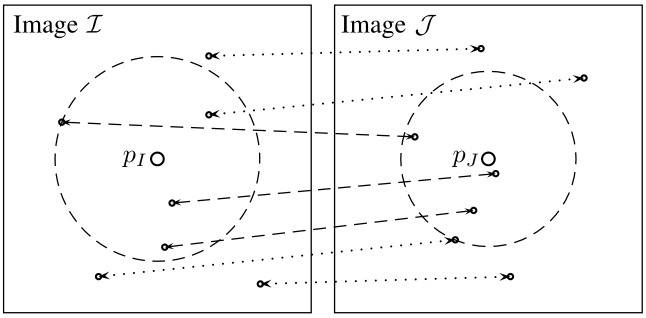


Fig. 2. Illustration of the topological constrains. Only three (dashed lines) out of $r = 4$ neighbours are matched keypoint pairs, $t(\mathcal{I},\mathcal{J},p_I,p_J) = \frac{3}{4}$.

The *similarity function* $s(\mathcal{I},\mathcal{J})$ between two images $\mathcal{I}$ and $\mathcal{J}$ is defined using the *topological verification function* $t(\mathcal{I},\mathcal{J},p_I,p_J)$ for all matched keypoint pairs $P(\mathcal{I},\mathcal{J})$ of both images (Eq. 3). The normalization factor $P^{max}(\mathcal{I},\mathcal{J})$ is the maximum possible number of matched keypoint pairs generated by the matching routine, given images $\mathcal{I}$ and $\mathcal{J}$.

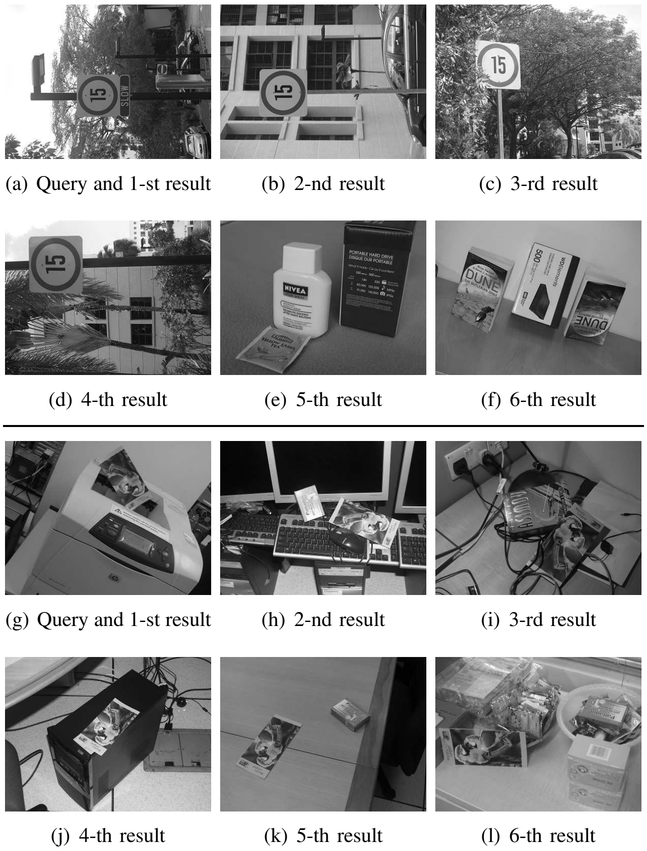
$$s(\mathcal{I},\mathcal{J}) = \sum_{(p_I,p_J)\in P(\mathcal{I},\mathcal{J})} \frac{t(\mathcal{I},\mathcal{J},p_I,p_J)}{P^{max}(\mathcal{I},\mathcal{J})}. \quad (3)$$

An important property of the proposed *similarity function* is that it can detect the presence of similar fragments (even very small ones) in images with very complex and diversified backgrounds. The function is also computationally effective; it requires only $O(pr)$ operations, where p is the number of keypoint pairs $P(\mathcal{I},\mathcal{J})$ and r is the size of the neighbourhood (the costs of keypoint matching are not included into the complexity of the function). Additionally, it is not sensitive to a certain level of inaccuracies in keypoint matching.

Efficiency of keypoint matching is, obviously, another important aspect of the algorithm. The classic approach (e.g. *coherent pairs* method, i.e. *one-to-one* matching) is extremely slow and takes as much as $O(fp^2)$, where f is the length of keypoint descriptor vectors. This makes the classic (exact) approach not applicable to the pre–retrieval step, and we need to search for an approximate keypoint matching approach. There is a variety of approximate nearest neighbour algorithms e.g. [8], [9]. We have implemented a method which has been experimentally found time–efficient (although we did not compare its performances against the available alternatives). Exemplary pre–retrieval results are shown in Fig. 3.

B. Image fragment matching (step B)

The key factor in automatic visual object formation is a reliable detection of multiple similar fragments in a collection images (without any prior knowledge about the image contents). The assumption regarding the complete lack of prior knowledge is very important, because the system has to explore and learn unknown environments.


(a) Query and 1-st result (b) 2-nd result (c) 3-rd result

(d) 4-th result (e) 5-th result (f) 6-th result

(g) Query and 1-st result (h) 2-nd result (i) 3-rd result

(j) 4-th result (k) 5-th result (l) 6-th result

Fig. 3. Examples of pre–retrieval based on *image topology similarity* function $s(\mathcal{I},\mathcal{J})$. The proposed similarity function is able to detect a presence of small similar fragments in images with very complex and diversified backgrounds.

Let us now formalize the similar fragments detection routine. Given a set of images $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_n\}$ we would like to detect all existing similarities within the set. To make the process more efficient, for each image $\mathcal{I}_x : x \in \{1, ..., n\}$, we filter out only a subset of most similar images $\{\mathcal{I}_x^1, \mathcal{I}_x^2, ..., \mathcal{I}_x^m\} : m \ll n$ (see Section II-A). Thus, there are nm possible image pairs to be checked. For a single pair of images, a reliable detection of similar fragments can be solved by a *image fragment matching* method. The fragment matching method generates a set of image fragment pairs, first element of each pair represents a fragment on the first image, second element of the pair represents the similar fragment on the second image. When all similar images in the database are matched, the resulting set (union of sets from all pairs) contains all similarities found within the database.

Image fragment matching the most important and the most time consuming operation of the whole approach. We use two such methods, namely: *geometric* and *topological* keypoint matching. We will shortly present both of them, now.

1) The geometrical method with triangles: The objective of geometric image fragment matching is to reconstruct a set of *affine* transformations relating similar planar surfaces present on both input images. The transformations are reconstructed from triangle pairs built over both images using pairs of matched keypoints. Affine transformations are decomposed into elementary transformations (rotations, translations and scales). A six–dimensional histogram (affine transformations have six degrees of freedom) of all transformations is built for a pair of images. A 2D subspace (two elementary rotations) of an exemplary histogram for a selected pair of images is visualized in Fig. 4. A non–parametric approach is used to find peaks of the histogram, which represent dominant affine transformations relating both images (i.e. relating similar fragments in the images). Sets of triangle pairs which contribute to these peaks form the outlines (convex hulls) of similar fragments shared by both images. Further details can be found in [10], [11].

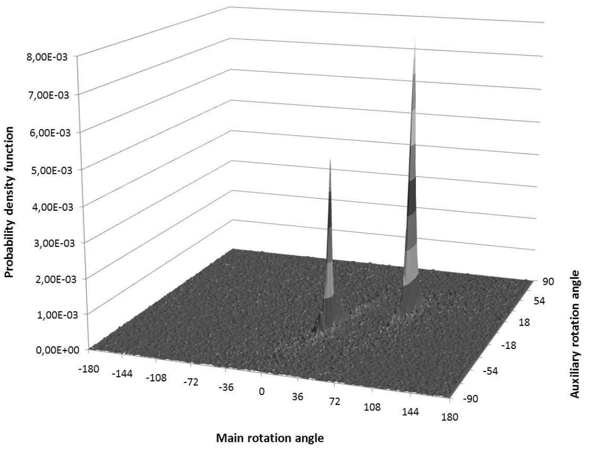


Fig. 4. Histogram of two rotations extracted from affine transformations [11]. Two peaks are visible, they represent two similar fragments.

2) The topological method: An alternative approach is the topological image fragment matching. Instead of recreating exact geometrical transformations between fragments of two images, this method focuses only on image topology. A *topological constrain* is introduced which, in general, reliably represents shape distortions of physical objects. We assume that neighbouring keypoints have to obey this constrain to be considered a similar fragment. Locally similar fragments are, therefore, extracted according to the results of topological constrain verification. The proposed topological constrain is the *matching order of vector orientations connecting keypoints from a selected pair to the neighbouring keypoints*. The constrain for a given keypoint pair is illustrated in Fig. 5. The topological method is more flexible than the geometric one; it is able to detect non-planar and deformed fragments of similar objects. However, the detection is less accurate in terms of generated fragment outlines (also represented as convex hulls). Further details on the topological method can be found in [7].

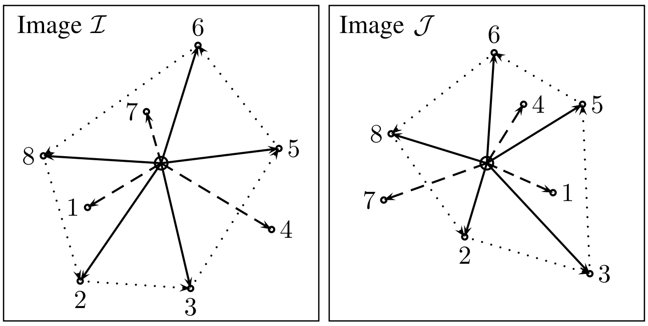


Fig. 5. Topological image matching concept. For each pair of matched keypoints, the largest subset of orientation-ordered neighbors is found. An exemplary subset of size 5 is shown.

3) Performances of image fragment matching: Reliable visual object formation is possible only if image fragment matching is performed with a high quality. In fact, any *false positive* matching error is very problematic for the future processing based on graph analysis. Such errors result in false connections within the graph and may result in incorrect contents of visual objects. The proposed routines are somewhat resistant to such errors, but this resistance is rather weak. On the other hand, *false negative* fragment matching errors are much less problematic. In case of missing graph connections, some visual objects may not be formed correctly. To solve this problem, it is usually enough to pre–retrieve more images or to deliver more images which would contain the corresponding physical objects. However, both of these solutions increase the computational costs of the method.

TABLE I
AVERAGE RECALL AND PRECISION FOR THE TEST DATASET.

Detector *Descriptor*	HarAff SIFT	HarAff GLOH	HarAff Mom.	SURF SURF	MSER SIFT
Method	Geometrical method				
Prec. (area)	0.96	0.96	0.97	0.90	0.95
Recall (area)	0.64	0.50	0.47	0.49	0.53
F-m. (area)	0.77	0.66	0.64	0.63	0.68
Prec. (obj.)	0.97	0.97	0.97	0.98	0.94
Recall (obj.)	0.81	0.71	0.70	0.61	0.68
F-m. (obj.)	0.88	0.82	0.81	0.75	0.79
Method	Topological method				
Prec. (area)	0.64	0.62	0.78	0.50	0.71
Recall (area)	0.79	0.74	0.64	0.70	0.63
F-m. (area)	0.71	0.67	0.70	0.59	0.67
Prec. (obj.)	0.98	0.97	0.99	0.97	0.98
Recall (obj.)	0.92	0.88	0.86	0.79	0.78
F-m. (obj.)	0.95	0.92	0.92	0.87	0.87

The achieved matching results (the geometric and topological approaches) on the processed database are shown in Table I. Two measurement modes are used: in the first one (object-wise) we check if the similar fragments are matched, in the second one (area-wise) we measure how accurately the shapes of matched fragments are outlined. The geometric method is more precise in terms of area measurement, due to very strict mathematical foundations. The topological method is less precise, but it is able to find more matching fragments.

In Fig. 6 we show two exemplary cases of similar fragment matching using the geometric approach.

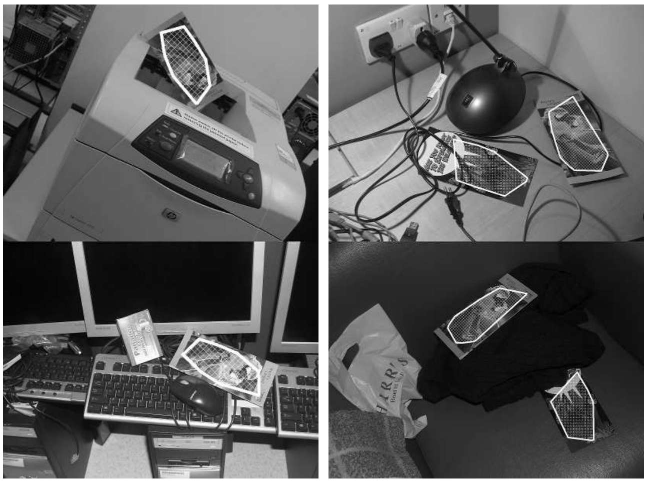


Fig. 6. Examples of image fragment matching using geometrical method

C. Automatic formation of prototypes (step C)

Image fragment matching provides us with a set of image fragment pairs. While image fragments within a pair are related (similar) there is not direct visual relation between fragments belonging to different pairs (even if detected in the same image). We need, nevertheless, to establish such relations because some of those fragments may represent the same physical object; the first step is to build relations between fragments extracted within the same image.

As shown in Fig. 1 a single image $\mathcal{I}_x : x \in \{1, ..., n\}$ is matched (i.e. it shares similar fragments) with a subset of images from the database. These fragments depicts physical objects present in the corresponding pairs of images. If we consider physical objects located in a single image $\mathcal{I}_x$, there might be several image fragments depicting each of these objects (they come from different matching processes against the same fragments of image $\mathcal{I}_x$). Such fragments should be very similar in shape, size and location (they are on the same image and approximate the same underlying physical object). Therefore, we assume that similar fragments represent the same physical object of image $\mathcal{I}_x$ and, thus, such groups of similar fragments are called *prototypes*; this is an important concept in the proposed approach. In fact, *prototypes* are intermediate structures required for form **visual objects**.

The process of prototypes construction is a grouping problem. To extract prototypes in a single image $\mathcal{I}_x$ we analyse intersections of image fragments. The larger is the relative size of the intersection, the higher chance that both fragments represent the same physical object. Two fragments are merged (to form a prototype) if: (1) both have similar sizes (areas), (2) the intersection of fragments is relatively large, i.e.

$$\min\left(\frac{c_1}{c_2}, \frac{c_2}{c_1}\right) > t_R, \quad \frac{c_I}{\min(c_1, c_2)} > t_I, \qquad (4)$$

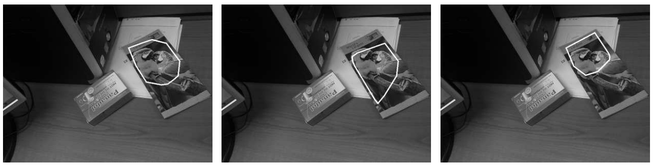


Fig. 7. Different image fragments (convex hulls) belonging to the same *prototype*. They come from matching of a single image with other images from the collection.

where: c_1 – area of the first fragment, c_2 – area of the second fragment, c_I – area of fragment intersection, t_R – area ratio threshold, t_T – intersection area ratio threshold.

Prototypes are formed from multiple fragments using a graph analysis technique. Each fragment is represented by a single graph node. Two graph nodes are connected by an edge if the underlying fragments satisfy the above merging criterion (Eq. 4). At least two fragments are necessary to form a prototype. Given the graph representation, prototype construction can be simply solved using graph connected component search. Various fragments of the same prototype formed in an exemplary image are given in Fig. 7.

D. Automatic formation of visual objects (step D)

Each prototype depicts (usually) a physical object located within a single image. Our ultimate objective is, however, to establish relations between prototypes form all database images, i.e. to form groups of prototypes that are referred to as *visual objects*.

Fortunately, the connections between images are already established in a form of similar fragment pairs (see Section II-B and Fig. 1). Since similar fragments are matched using a high–precision matching process (see Table I) the generated inter–image connections are mostly correct (this is a fundamental requirement for the visual object formation).

Formally, the formation of visual object is another graph–based grouping problem. We simply build a graph representing the above–mentioned connections between images. Each prototype $O_y \in y = \{1, ..., q\}$ is represented by a single graph node (note that prototypes are groups of image fragments from the same image). Each image fragment is matched with a similar fragment in another database image. Therefore, graph edges are created between nodes (prototypes) according to the matches between image fragments. Each prototype has a number multiple outgoing edges, equal to the number of similar fragments within this prototype. However, because the precision of similar fragment matching is still below 100%, a verification mechanism has to be put in place. The proposed verification mechanism is based on the analysis of graph *k-edge-connectivity*. A graph is *k-edge-connected* if it remains connected when less than k edges are deleted from the graph. In other words, a new prototype can be added to an existing visual object *if and only if* it is connected with at least k prototypes from the visual object. Such an approach is effective in eliminating random matching errors, because they usually form only a single connection to another

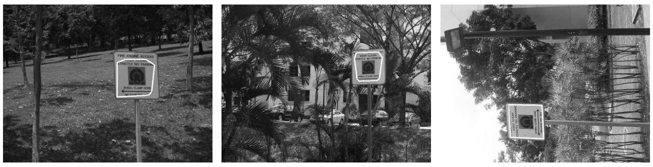


Fig. 8. Visual objects are formed out of many prototypes present on different images.

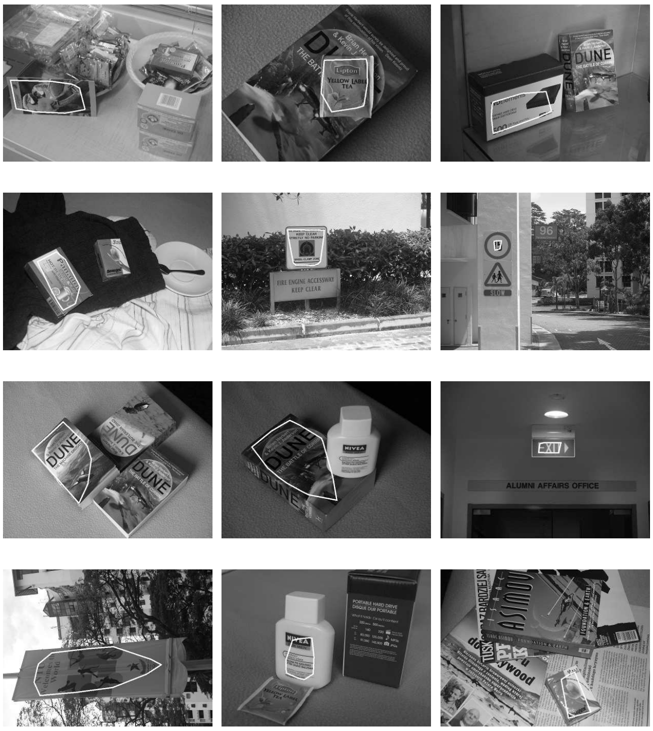


Fig. 9. Exemplary image fragments (instances of prototypes) representing visually formed objects.

prototypes. The resulting grouping algorithm is, therefore, a *k-edge-connected* subgraph search routine.

Exemplary image fragments belonging to different prototypes (but within the same visual object), found in various images are presented in Fig. 8.

In Fig. 9 we show a subset of visual objects automatically formed in the analysed collection of images (due to size limitations, each object is represented just by a single image fragment from one of the prototypes belonging to the object). Although no semantics is used during the object formation process, one my find that the created visual objects represent the actual physical objects appearing in multiple images within the database.

III. Discussion

Before presenting the experimental results we would like to discuss the applicability of the method and note its limitations. As stated above, the key requirement for successful object formation is very high precision of matching. Nowadays, precision near 100% may only be reached for image (fragment) matching problem, i.e. localization of image fragments containing identical objects. State-of-the-art approaches applicable for similarity based retrieval and grouping could not be used for the stated object formation problem, because their precision is still too low. Thus, the proposed method applies only for image collections depicting the same objects. Popular image databases such as *Caltech 101/256* may not be processed by the method, because they mostly contain similar (but *not identical*) underlying physical objects. For such databases *nothing would be found*.

Due to the mentioned, specific requirements, we have tested the method on a dataset containing images depicting a set of physically identical objects. For the reference and test purposes, the dataset may be downloaded from a web site[1]. First, we will discuss this dataset, later on we will show how to set up method's parameters. In the last part of this section we will summarize the achieved results.

A. The dataset and the objective of experiments

The dataset consists of 100 diversified images, captured both indoors and outdoors, containing a variety of objects. Most images in the database contains more than one object of interest, appearing in different configurations with diversified backgrounds. Camera settings and lighting conditions also differ between images.

Some of the physical objects repeat in several database images and thus, they are the candidates for visual objects. We have identified a set of physical objects present on at least three different images. These objects are: four different books, a notepad, a leaflet, two different medicine boxes, tissue pack, three different bottles, tea bag, two road signs, an exit sign and a street advertising poster. There are also other physical objects repeating only twice in the database (we consider them irrelevant because the assumed minimum number of prototypes in a visual object is $k = 3$, see Section III-B).

Our objective is to find all those repeating objects in the database and form visual object out of them. We expect that there will be no errors in the created objects, i.e. each visual object may represent *only one* underlying physical object. If a single physical object is represented by more than one visual object, we consider it a problem of lesser grade, because it is easily solvable in the proposed framework (see Section II-B3). Of course, the more correctly formed objects (without errors and object duplications) the better.

B. Method parameters

The proposed method has five parameters, related to (1) image pre–retrieval, (2) prototype formation and (3) visual object formation. All parameters, their short description and suggested values are listed in Table II.

The parameter m defines the size of the subset of most similar pre–retrieved images. The larger value of the parameter, the higher chance to capture important visual connections

[1] Image fragment matching dataset: http://www.ii.pwr.wroc.pl/∼visible

TABLE II
DEFAULT VALUES OF THE METHOD'S MAIN PARAMETERS.

Param.	*Meaning*	*Value*
m	Number of pre–retrieved images (% of n)	0.25n
r	Size of topological neighbourhood	60
t_A	Minimum ratio of image fragments area size	0.75
t_I	Minimum ratio of image fragments intersection size	0.75
k	Visual objects graph edge connectivity	3

between images from the database. However, larger values of the parameter significantly slows down the method. The maximum value of this parameter is n; this represents a disabled pre–retrieval, i.e. all image pairs are matched. The parameter r defines the size of the topological neighbourhood (see Eq. 2). To choose the proper value of the parameter, we need to consider two issues. The topological neighbourhood has to be large enough to be informative and it has to be small enough due to memory requirements (to speed up the method all neighbourhoods are pre–computed off–line).

Another two parameters are related to prototypes. They are used in the image fragment merging routine. As mentioned in Section II-C, merging should happen only if two image fragments (they are on the same image) represent the same physical object. Both parameter values t_A and t_I have been experimentally set to 0.75.

The last parameter is related to the visual object formation. To eliminate potential error matches, we want each prototype to be linked to at least k other prototypes (i.e. each prototype should contain at least k image fragments). In the presented approach, we have assumed $k = 3$, which is the minimum possible value. Due to high precision of image fragment matching, it is fully sufficient. However, if matching errors are more frequent (lower precision of image fragment matching) especially on larger databases, one should consider a larger value of the parameter. As a result, it would be more difficult to form visual objects but they would be more credible.

C. Discussion on created visual objects

The minimum requirement to form a visual object is having a physical object *correctly matched* on at least three different images ($k = 3$, see Section III-B). The quality of matching (i.e. precision of matched area) is in fact the most important element of successful object formation. To measure how well the visual objects are formed, we use the ground truth information as discussed in Section III-A. In fact, we want to capture as much of underlying physical objects as possible.

Given the processed database and the proposed parameter set up, there are *no errors* in the generated visual objects, i.e. each formed visual object represents only one physical object. In some cases, a physical object is represented by more than one visual object, but such objects are merged together when more data is provided (the pre-retrieval size is increased). Also, larger amount of matching data leads to larger number of formed objects. Given the pre–retrieval mechanism, we can easily set up how many images will be given to image fragment matching and subsequently how many similar fragments may be found. Comparison of results for different pre–retrieval scenarios are given in Tab. III. We note that not all objects have been found (see Section III-A), non–planar ones were the most problematic. Low number of pre–retrieved images causes only a few objects to be created. The more pre–retrieved images, the more formed objects and the quality increases, but the computational cost also increases.

TABLE III
AUTOMATICALLY FORMED OBJECTS VERSUS GROUND TRUTH OBJECTS. "+" REPRESENTS A CORRECTLY FORMED OBJECT, "–" REPRESENTS A MISSING OBJECT, "±" MEANS THAT MORE THAN ONE VISUAL OBJECT HAVE BEEN FORMED FOR THE UNDERLYING GROUND TRUTH OBJECT.

Ground truth	Pre–retrieval size (m) [% of n]							
defined object	3	4	5	10	15	20	50	100
	Geometrical fragment matching							
Book 1	±	±	+	+	+	+	+	+
Book 2	–	+	+	+	+	+	+	+
Book 3	+	+	+	+	+	+	+	+
Book 4	+	+	+	+	+	+	+	+
Bottle	–	–	–	–	+	+	+	+
Box	–	–	–	±	±	+	+	+
Exit sign	–	+	+	+	+	+	+	+
Leaflet	+	+	+	+	+	+	+	+
Medicine 1	±	+	+	+	+	+	+	+
Medicine 2	+	+	+	+	+	+	+	+
Road poster	+	+	+	+	+	+	+	+
Road sign 1	+	+	+	+	+	+	+	+
Road sign 2	–	–	–	–	+	+	+	+
Tea bag	+	+	+	+	+	+	+	+
Tissues	–	–	–	+	+	+	+	+
	Topological fragment matching							
Book 1	–	±	±	+	+	+	+	+
Book 2	+	+	+	+	+	+	+	+
Book 3	+	+	+	+	+	+	+	+
Book 4	–	+	+	+	+	+	+	+
Bottle	+	+	+	+	+	+	+	+
Box	–	–	–	+	+	+	+	+
Exit sign	–	–	–	–	–	–	–	–
Leaflet	–	+	+	+	+	+	+	+
Medicine 1	–	–	–	+	+	+	+	+
Medicine 2	–	–	–	–	–	–	–	–
Road poster	+	+	+	+	+	+	+	+
Road sign 1	–	–	–	–	–	–	–	–
Road sign 2	–	–	–	–	–	–	–	–
Tea bag	–	–	–	–	–	–	–	–
Tissues	–	–	–	–	–	–	–	–

Having the geometrical method employed for the matching task, we may expect very high area precision (see Tab. I). The number of pixel–level false positive errors in generated image fragments is minimal. But there is also a cost, non–planar objects will be only partially captured and they will most probably not constitute correct prototypes. The topological fragment matching overcomes the problem of non–planar objects, but has much lower precision in terms of matched area. Statistics demonstrating the ability to recreate meaningful objects are shown in Tab. III. Due to high precision the geometrical method is a more suitable candidate for object formation than the topological one. The main problem with the topological approach is much smaller number of prototypes successfully used in object formation. Large differences in image fragments (the main criterion for successful prototype formation) prevent capturing identical parts on the same image. Even though the number of matched fragments (mostly correct) is higher for the topological method, the overall number of formed objects is lower. Statistics for the numbers of created prototypes and visual objects are given in Tab. IV.

m [%]	Matched fragments	Formed prototypes	Prototypes in objects	Formed objects	True objects
	Geometrical fragment matching				
3	203	106	39	11	9
4	291	131	64	12	11
5	357	138	72	11	11
10	589	162	95	14	13
15	**709**	**174**	**105**	**16**	**15**
20	**778**	**176**	**102**	**15**	**15**
50	951	188	108	15	15
100	1063	196	108	15	15
	Topological fragment matching				
3	218	114	13	4	4
4	307	155	33	8	7
5	378	186	45	8	7
10	617	244	63	9	9
15	752	290	67	9	9
20	840	307	72	9	9
50	1059	348	80	9	9
100	1211	382	80	9	9

TABLE IV
CREATED PROTOTYPES AND VISUAL OBJECTS FOR VARIOUS SETTINGS OF PRE-RETRIEVAL (m).

An interesting case is present for geometrical matching, $m = 15$ and $m = 20$. The number of prototypes successfully used in object formation is in fact *decreasing*. Surprisingly, this is a correct behaviour. Some image fragments, which should create a single prototype, were not merged together for $m = 15$. Instead, multiple different prototypes are created. If the number of pre–retrieved images is increased to $m = 20$, missing links between these prototypes are found. Image fragments are properly merged and a single prototype is formed. Thus, we can also see a decrease (by 1) of the number of visual objects. In fact, similar behaviour happens quite often (see "±" in Tab. III), but in this case it is well captured.

We may also observe a much lower number of formed visual objects for the topological fragment matching. It is related to the already mentioned problem of lower area precision. Decreasing the values area related merging thresholds (t_A and t_I) results in creating of *false objects* and thus is not an acceptable approach. This confirms one of the initial statements, that successful visual object formation requires image fragment matching working with very high precision.

Apart of the listed features of the proposed method, there are also weak points. In some cases one detected prototype is a part of another prototype. Prototype formation criterion based on fixed merging thresholds (t_A and t_I) can not capture it correctly. Such prototypes (and later on visual objects) will not be joined, even though they represent the same underlying physical object (or a part of it). Therefore, we should consider building a *hierarchy* of prototypes and visual objects, instead of a plain structure.

IV. SUMMARY

A method for automatic formation of visual objects has been presented. The method is able to find meaningful image fragments from an unlabelled set of images. Visual objects are formed out of repeating, similar image fragments within the dataset. The proposed method employs *image fragment matching* techniques to extract such similar fragments of images. Two matching techniques are used, namely: the geometric and the topological ones. Apart from the visual object formation solution, we have also presented a novel image pre–retrieval method that effectively identifies images prospectively containing similar fragments. The method uses a topology–based similarity function. It is an important component of the system, because it significantly shortens the matching process.

The proposed solution creates a set of visual objects, without any kind of structure or hierarchy. This might be its weak point, because some similar fragments may indeed be structured (e.g. one object represents a visual fragment of another object). Our further research will focus on building such a hierarchy of visual objects.

ACKNOWLEDGEMENT

The research presented in this paper is a part of A*STAR Science & Engineering Research Council grant 072 134 0052. The financial support of SERC is gratefully acknowledged.

This work is partially financed from the Ministry of Science and Higher Education Republic of Poland resources in 2008–2010 years as a Poland–Singapore joint research project 65/N-SINGAPORE/2007/0.

REFERENCES

[1] A. W. Smeulders and A. Gupta, "Content-based image retrieval at the end of the early years," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, 2000.

[2] S. Dickinson, A. Leonardis, B. Schiele, M. Tarr, S. Edelman, and P. Valery, *Object Categorization: Computer and Human Vision Perspectives*, 2009.

[3] A. Śluzek and M. Paradowski, "A vision-based technique for assisting visually impaired people and autonomous agents," in *Proc. 3th Int. Conf. on Human System Inteaction HSI2010*, 2010, pp. 653–660.

[4] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, pp. 63–86, 2004.

[5] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th IEEE Int. Conf. Computer Vision*, vol. 2, 1999, pp. 1150–1157.

[6] ——, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[7] M. Paradowski and A. Śluzek, "Keypoint-based detection of near-duplicate image fragments using image geometry and topology," in *Proc. International Conference on Computer Vision and Graphics, LNCS*, 2010, in press.

[8] A. Andoni, M. Datar, N. Immorlica, P. Indyk, and V. Mirrokni, "Locality-sensitive hashing using stable distributions," *Nearest Neighbor Methods in Learning and Vision: Theory and Practice*, 2006.

[9] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Applications*, 2009.

[10] M. Paradowski and A. Śluzek, "Detection of image fragments related by affine transforms: Matching triangles and ellipses," in *Proc. International Conference on Information Science and Applications*, vol. 1, 2010, pp. 189–196.

[11] ——, "Local keypoints and global affine geometry: triangles and ellipses for image fragment matching," *Innovations in Intelligent Image Analysis*, 2010, in press.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Learning taxonomic relations from a set of text documents

Mari-Sanna Paukkeri*, Alberto Pérez García-Plaza†, Sini Pessala*, and Timo Honkela*
*Aalto University School of Science and Technology, Adaptive Informatics Research Centre
P.O. Box 15400, FI-00076 AALTO, Email: first.last@tkk.fi
†NLP & IR Group, E.T.S.I. Informática, UNED
28040, Madrid, Spain, Email: alpgarcia@lsi.uned.es

***Abstract*—This paper presents a methodology for learning taxonomic relations from a set of documents that each explain one of the concepts. Three different feature extraction approaches with varying degree of language independence are compared in this study. The first feature extraction scheme is a language-independent approach based on statistical keyphrase extraction, and the second one is based on a combination of rule-based stemming and fuzzy logic-based feature weighting and selection. The third approach is the traditional *tf-idf* weighting scheme with commonly used rule-based stemming. The concept hierarchy is obtained by combining *Self-Organizing Map* clustering with agglomerative hierarchical clustering. Experiments are conducted for both English and Finnish. The results show that concept hierarchies can be constructed automatically also by using statistical methods without heavy language-specific preprocessing.**

I. Introduction

An ontology is a directed graph consisting of concepts as nodes and relations as the edges between the nodes. A well-defined ontology has names for the concepts and specify what kind of relation there is between the concepts. The ontology is represented using a formal language, such as DARPA Agent Markup Language (DAML) or Web Ontology Language (OWL). With the use of the formal language, axioms can be specified to determine validity and to define constraints in ontologies. Ontologies are used widely in different Natural Language Processing (NLP) applications: examples are word sense disambiguation [1], annotating images [2], assessing text difficulty [3], and monitoring disease epidemics by analysing textual reports from the Web [4].

Creation of ontologies has obvious benefits, since it appears ideal that all systems within some domain would use similar terminologies and shared ontologies. The continuous change of conceptual systems through innovations and other activities requires that also ontologies need to be updated. The costs related to ontologies stem from two main sources: the development of a shared conceptual system and the use of it [5]. The development of an ontology typically consists of defining the concepts and the relationships between the concepts. The typical stages of an ontology building process are the following [6]: (1) domain analysis resulting into the requirements specification, (2) conceptualization resulting into the conceptual model, (3) implementation that leads into the specification of the conceptual model in the selected representation language, and (4) the ontology population i.e. the generation of instances and their alignment to the model that results into the instantiated ontology. Ontology maintenance includes getting familiar with and modifying the ontology. Reuse involves costs for the discovery and reuse of existing ontologies in order to generate a new ontology [6]. Simperl et al. [7] present a cost estimation model for ontology engineering. They estimate the person months associated to building, maintaining and reusing ontologies calculated as the product of the size of the ontology.

Ontologies have been mainly created manually but since the cost of manual creation is huge, also automated ways have been studied, often using methods developed originally for other fields of science. Ontology learning is the process of identifying terms, concepts, taxonomic and non-taxonomic relations and optionally axioms from natural language text and using them to construct and maintain an ontology.

In this paper, we assume that we have a set of text documents, each describing a concept in natural language. We extract taxonomic relations between the described concepts by using a selection of feature extraction methods and clustering the resulting feature vectors. This approach is automatic and easily portable to other domains and languages since it needs just a set of dictionary-like pages or documents as resources. One of our approaches is language-independent while the others use some language-dependent preprocessing steps.

A. Related work

Learning or extracting taxonomic relations means finding hypernymies between concepts with the goal of constructing a concept hierarchy. On the other hand, non-taxonomic relations are other than hyponymic relations between concepts in an ontology including e.g., synonymy and antonymy.

Taxonomic relations have been extracted automatically for a long time: Amsler created automatically a taxonomy for English nouns and verbs using dictionary definitions [8]. Hearst introduced lexico-syntactic patterns that indicate hyponymy relations [9]. Those have been further used for taxonomy learning e.g. in [10], [11]. Taxonomies have been learned also from Wikipedia by using the Wikipedia categories as concepts in a semantic network, connectivity of the network and on applying lexico-syntactic patterns [12]. Statistical methods for extraction of taxonomic relations have been covered in [13] including hierarchical and non-hierarchical clustering,

978-83-60810-27-9/09/$25.00 © 2010 IEEE

similarity measures and different linking schemes. [14] introduced their guided agglomerative hierarchical clustering algorithm that create concept hierarchies from text collections exploiting a hypernym oracle. The oracle exploits hypernyms from WordNet and using the Hearst lexico-syntacic patterns matched in a corpus or Internet. [15] proposed a clustering approach for taxonomy learning that incorporates evidence from multiple classifiers to optimize the entire structure of the taxonomy.

Also a combination of natural language processing tools have been used in extracting relations from text. A massive approach uses lemmatiser, syntactic parser, part-of-speech tagger, pattern-based classification and word sense disambiguation models together with resources such as domain ontology, knowledge base, and lexical databases [16]. Another approach is an unsupervised model for learning arbitrary relations between concepts of an ontology [17]. The approach uses corpus of manually tagged named entities, corresponding to ontology concepts and syntactic patterns.

In a recent work [18] the variety of ontology and concept hierarchy learning have been explained comprehensively. The work also introduces the Tree-traversing Ants (TTA) clustering technique for learning taxonomic relations. TTA is based on dynamic tree structures and it adopts a two-pass approach for term clustering. During the first pass, nodes are recursively broken into sub-nodes using Normalized Google Distance (NGD). The second pass is a refinement phase where terms are relocated according to n-degree of Wikipedia (noW) measure that uses Wikipedia Categories information.

The problem of the current methods is that they use of a wide range of language-specific tools, dictionaries or ontologies and thus exporting to new languages or domains is difficult or in some cases even impossible due to the lack of resources. The work by Wong seems to be more independent of the used language but instead needs access to Google and Wikipedia.

B. *Our contribution*

Our methodology can use any encyclopedia entries or other topic-related documents as definitions of concepts and creates taxonomic relations based on this data alone. No access to online sources is needed after the collection of the document set. Our methodology uses a very small amount of language-specific information and is thus easily portable to other languages. Other works that use Wikipedia use mostly the category information as concepts for taxonomy learning. Instead, we use the Wikipedia articles as concepts. Wikipedia articles have been used as concepts also in [19] but they do not create taxonomies but only measure relatedness between words or documents. In the existing work, term extraction methods are usually used for extraction of labels for concepts in the ontology, but we use keyphrase extraction for selection of a relatively small amount of terms for document representation.

II. Methods

We propose a methodology to generate taxonomic relations or concept hierarchies automatically from a set of encyclopedia documents. Each document is a description of a concept (the same assumption is made e.g. in [19]). These concepts are on the lowest level of the ontology and we aim to cluster the documents hierarchically to obtain an ontological structure of the concepts. Our methodology for generation of taxonomic relations consists of two basic steps: 1) feature extraction and 2) hierarchical clustering of the feature vectors. The result is a hierarchical structure generated according to the contents of the text documents.

In this study, generation of taxonomic relations is carried out for both English and Finnish languages. Finnish is a highly agglutinative language and as such relatively different from English. By using this language pair in our experiments we want to show that our methodology is further exploitable for several other languages.

A. *Feature extraction*

Feature extraction aims to reduce input data dimensionality, extracting the relevant information and removing redundancies. Traditional document representations are built over the *Vector Space Model (VSM)* [20], using term weighting functions based on term and document frequencies. The weighting is supposed to reflect the importance of each word to represent a particular document in the context of a document collection or corpus.

We propose three different approaches for document representation. One of them is a combination of heuristic criteria exploiting document structure by means of fuzzy logic. The second approach utilizes a statistical keyphrase extraction method to automatically extract features for document representation. The third method is the traditional *tf-idf* term weighting function that is also used as a baseline for our study. These representations are selected to compare how purely statistical method performs compared to a more heuristic method that gathers extra knowledge from the documents' meta information.

1) Fuzzy combination of criteria: When a human reader tries to understand the contents of a document, his or her attention is focused on some particular elements. Title or emphasized words are usually considered more important than the rest of the document. Moreover, the first and the last parts of a document usually contain overviews, summaries or conclusions with important keywords.

A fuzzy logic-based representation called *Extended Fuzzy Combination of Criteria* (*efcc*) [21] aims to exploit the semantics reflected by the use of a specific subset of HTML tags. The main idea is to define the importance of each word by combining several heuristic criteria. These criteria are related to word frequencies in the whole document, in titles, in emphasized text segments, or in first or last parts of a document. If similar information is available, the *efcc* approach can be used with any kind of document, not only with HTML-encoded documents.

The fuzzy system is built over the concept of linguistic variable, which value can be defined using natural language words and fuzzy sets. Each variable describes the membership degree of an object to a particular class. In the *efcc* approach, they are defined from human expert knowledge based on word frequencies mentioned above. Then, the knowledge base is defined by a set of IF-THEN rules combining these variables, in order to describe system behaviour as much precisely as possible. The aim of these rules is to combine one or more input fuzzy sets (antecedents), associating them with another output fuzzy set (consequent). Once the consequents of each rule have been calculated, and after an aggregation stage, the final set is obtained.

In this way, the knowledge used to build the rules is based on the following simple ideas: (1) A word appearing in the title or emphasized should appear in any other criterion to be considered important; (2) Words appearing in the first or last part of a document could be more important than others, because documents usually contain summaries or relevant ideas to attract reader's interest; (3) A non-emphasized word could mean that no words are emphasized in the web page; (4) A word not appearing in the title may indicate that the page has no title or the title has no meaning, i.e. it does not enclose relevant words; (5) High frequency of a word in a page could be important when the previous criteria are not enough to choose the most relevant words.

Some samples of these rules are (the rest can be found in [21]):

TABLE I
SAMPLE RULES

IF Title == High AND emph == High
THEN relevance = Very relevant
IF Title == High AND emph == Medium AND Position == Preferential
THEN relevance = High relevance
IF Title == High AND emph == Medium AND Position == Standard
THEN relevance = Medium relevance
...
IF Frequency == Medium
THEN relevance = Medium relevance

The inference engine is based on a center of mass algorithm (COM) that weights the output of each fired rule taking into account the truth degree of its antecedent. The output is a linguistic label with an associated number related to the relevance of a specific word in the page. A more detailed explanation of the fuzzy system can be found in ([21], [22] and [23]).

In addition to the *efcc* term weighting function, also inverse document frequency *idf* is used (equation 1).

$$\mathrm{idf_i} = \log \frac{|D|}{|\{d : t_i \in d\}|} \tag{1}$$

Where D is the total number of documents, and the denominator represents the number of documents containing the term i in the collection.

This modification of *efcc* aims to penalize those words appearing in a high number of documents due to the fact that Wikipedia pages are template-based and, therefore, there are terms appearing in each and every page (equation 2).

$$(\mathrm{efcc\text{-}idf})_{\mathrm{i,j}} = \mathrm{efcc_{i,j}} \times \mathrm{idf_i} \tag{2}$$

Where $\mathrm{efcc_{i,j}}$ is the relevance value obtained by means of the fuzzy system for a term i in a document j.

Moreover, discarding terms appearing in less documents than a particular percentage of the whole collection size can alleviate the effect of *idf* over too discriminative terms. If a concrete term appears in less than 4 documents, it is removed. This approach is called *df-efcc-idf*. Other intermediate variants were tested, like using just *efcc* values and the one shown in equation 2, but the results were uniformly worse.

To reduce the dimensionality of document vectors by selecting the most important features, we use the *Most Frequent Terms until n level* (*mft*) reduction method. This method consists of ranking the terms document by document based on the term weighting function values. A separate ranking is created for each document. In the first step, we will take terms appearing in the first position of each document ranking, ordering them first based on how many times a term has been found in different document rankings, and then, if two or more terms appear the same number of times in different rankings, based on the maximum weight found for each term. If we do not have terms enough, then we will take the terms appearing in second position in each ranking, and so forth.The process stops when the desired number of terms has been reached.

2) Statistical keyphrase extraction: The second feature extraction approach *Likey* [24] comes from the tradition of statistical machine learning. It extracts keyphrases of a document based on phrase frequencies. *Likey* does not use any language-dependent preprocessing tools or vocabularies and the only language-specific component needed is a reference corpus in each language. The basic idea in the method is to see whether the relative frequency of term candidates in the document collection is larger than their frequency in the reference corpus.

The *Likey ratio* [24] for each phrase is defined as

$$L(p, d) = \frac{rank_d(p)}{rank_r(p)}, \tag{3}$$

where $rank_d(p)$ is the rank value of phrase p in document d and $rank_r(p)$ is the rank value of phrase p in the reference corpus. Phrases are all the n-grams of the document up to a phrase length n. The rank values are the ordered frequencies of the phrases of the same length; the phrase having the largest frequency gets the rank of 1. In case of the same frequency value the rank value also stays the same. If the phrase p does not exist in the reference corpus the value of the maximum rank for phrases of length n is used: $rank_r(p) = max\_rank_r(n) + 1$. The *Likey ratio* is used to order the phrases existing in each document with those phrases having the smallest ratio being the best candidates for being a keyphrase.

As a post processing step, the phrases of length $n > 1$ face an extra removal process. If the reference rank value $rank_r$ of any of the single words constituting the phrase is smaller than the rank of the whole phrase, that means, the word is more common than the phrase, the phrase is removed. This uses the assumption that the maximum rank value is usually smaller for longer phrases than for unigrams since the frequencies of longer phrases are lower. In addition, lower rated subphrases of the existing keyphrase are also pruned.

The *Likey ratio* cannot be used directly as keyphrase weights since the best keyphrases get the smallest *Likey ratio* values. We thus scale the ratio to values between [0, 1], where values closer to 1 are the best keyphrases. The scaled weights are calculated with

$$w_2(p) = \begin{cases} (\frac{1}{t} - L(p,d)) * t & \text{if } L(p,d) < \frac{1}{t} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where $1/t$ is the maximum *Likey ratio* taken into account.

3) Tf-idf *term weighting:* As our third feature extraction method and also the baseline, we use the traditional *tf-idf* term weighting scheme (equation 5).

$$(\text{tf-idf})_{\text{i,j}} = \text{tf}_{\text{i,j}} \times \text{idf}_{\text{i}} \quad (5)$$

Where $\text{tf}_{\text{i,j}}$ is the number of times a given term i appears in document j.

We weight the unigrams of each document with the other documents as a reference. To reduce vector size, terms with high and low document frequencies are discarded.

This is selected as the baseline method because it is a very well known representation used nowadays in many fields like clustering, classification, information retrieval, etc. *Tf-idf* has also been used as a baseline in several unsupervised learning and clustering works that combine VSM with Self-Organizing Maps, e.g. [25], [26], and [27].

B. Hierarchical clustering

To learn the taxonomic relations between concepts, we assume that each concept has a hypernym, i.e. a parent node. Moreover, we assume that there is a maximum number of distinct hypernyms among the concepts. The concepts are represented by term feature vectors, and *Self-Organizing Map (SOM)* [28] is used to create an ordered space of the concept vectors. The *SOM* is an artificial neural network that orders data using unsupervised learning. Even though the *SOM* is a good method for reducing dimensionality and finding a topological order of data, it does not perform the explicit classification task. Because of this, hierarchical clustering of the prototype vectors of the *SOM* is applied on the top of the *SOM* map to define the borders of clusters. We use agglomerative hierarchical binary clustering to produce up to m clusters for the *SOM* map. The next levels are obtained by training a new *SOM* separately for the feature vectors in each cluster and using agglomerative hierarchical clustering for the new *SOM* maps.

A K-level concept hierarchy is built by first assuming that the zero-level data (concepts, i.e. term vectors) constitute the zero-level cluster, i.e., the root node. The zero-level data are then clustered into first-level clusters. The feature vectors in each of the level k clusters are further clustered to get the $k + 1$-level clusters and thus a more fine-grained clustering until $k = K$. The level K concepts are then the original concepts represented by the feature vectors. Somewhat similar approach is the *growing Self-Organizing Map* [29] and further the *growing hierarchical Self-Organizing Map* [30]. The growing of the map in that case is based on variations of Average Quantization Error, expanding single neurons instead of clusters.

Another possibility in this study would have been to use hierarchical clustering algorithm directly to the feature vectors but that would have needed another method that reduces the resulting binary tree into a small number of hierarchy levels.

C. Evaluation methodology

The evaluation of automatically generated ontologies may concentrate on different levels: evaluation of the lexical layer, hierarchy, or context level [31]. Four approaches for evaluating ontologies have been considered in the literature: comparison to a gold standard (which may itself be an ontology), evaluation as a part of an application, comparison to a source data about the domain, and evaluation by humans [31]. We follow the first approach and compare the generated conceptual hierarchies to a manually constructed reference ontology.

To be able to compare the hierarchical structure to an ontology, we label the concepts of the generated hierarchical structure with the terms (concept names) in the reference ontology. First, the topic of each text document is extracted from the incorporated meta data of the documents. The topics are used to label concepts on the lowest-level in the generated concept hierarchy. Also the parents for each topic are extracted from the reference ontology. The clusters on the generated hierarchy act as the hypernyms (parent concepts) and they are labeled according to the concepts forming the cluster. The majority of the parents is selected as the label for the cluster, i.e., the hypernym. In a case of two parent candidates of equal sizes the hypernym is selected randomly.

Each label can exist only once in the hierarchy in evaluation. Therefore, for labeling purposes, clusters having both the same label and the same parent are merged. In a case where the clusters with the same label have different parents, the cluster having less concepts assigned with that label is considered as unclassified. In a case of equal size, the unclassified cluster is selected randomly. In this way, we penalize similar clusters when they are far away, avoiding penalizing the errors due to too fine-grained clustering.

We use the TP_{csc} evaluation metrics [32] for evaluation of the automatically generated ontologies. Also Wilks and Brewster [33] and Brewster et al. [34] have performed their evaluation by a gold standard using these metrics, but not using reference labels. The experiments are focused on finding single concepts and their relations, and not the whole hierarchy as is our case.

Dellshcaft and Staab [32] introduce TP_{csc}, TR_{csc} and TF_{csc} values for the evaluation of a concept hierarchy of an ontology. The TP_{csc} metric is a global taxonomic precision TP, which uses the common semantic cotopy csc. The common semantic cotopy csc is defined as a set of concept identifiers, which are sub concepts $c <_{\mathcal{C}_1} c_1$ and super concepts $c >_{\mathcal{C}_1} c_1$ of the concept identifier c_1 in concept a hierarchy $\mathcal{C}_1$ and which are also concept identifiers in the ontology $\mathcal{O}_2$

$$csc(c, \mathcal{O}_1, \mathcal{O}_2) = \{c_i | c_i \in \mathcal{C}_1 \cap \mathcal{C}_2 \wedge (c_i <_{\mathcal{C}_1} c \vee c <_{\mathcal{C}_1} c_i)\}. \tag{6}$$

Local taxonomic precision tp_{csc} compares common semantic cotopies of concepts of $c_1 \in \mathcal{C}_1$ and $c_2 \in \mathcal{C}_2$

$$tp_{csc}(c_1, c_2, \mathcal{O}_1, \mathcal{O}_2) = \frac{|csc(c_1, \mathcal{O}_1, \mathcal{O}_2) \cap csc(c_2, \mathcal{O}_1, \mathcal{O}_2)|}{|csc(c_1, \mathcal{O}_1, \mathcal{O}_2)|}. \tag{7}$$

The global taxonomic precision TP_{csc} metric is based on local taxonomic precision of the common concepts of a learned ontology $\mathcal{O}_C$ and the reference ontology. The value of TP_{csc} tells how many of the semantic relations of the learned ontology can be found in the reference ontology

$$TP_{csc}(\mathcal{O}_C, \mathcal{O}_R) = \frac{1}{|\mathcal{C}_C \cap \mathcal{C}_R|} \sum_{c \in \mathcal{C}_C \cap \mathcal{C}_R} tp_{csc}(c, c, \mathcal{O}_C, \mathcal{O}_R). \tag{8}$$

The global taxonomic recall TR_{csc} metric is calcuated using global taxonomic precision TP_{csc}

$$TR_{csc}(\mathcal{O}_C, \mathcal{O}_R) = TP_{csc}(\mathcal{O}_R, \mathcal{O}_C). \tag{9}$$

The taxonomic F-measure TF_{csc} is the harmonic mean of the global taxonomic precision and recall.

$$TF_{csc}(\mathcal{O}_C, \mathcal{O}_R) = \frac{2 \cdot TP_{csc}(\mathcal{O}_C, \mathcal{O}_R) \cdot TR_{csc}(\mathcal{O}_C, \mathcal{O}_R)}{TP_{csc}(\mathcal{O}_C, \mathcal{O}_R) + TR_{csc}(\mathcal{O}_C, \mathcal{O}_R)} \tag{10}$$

III. Data

The data used in this study consists of a document collection from Wikipedia that is used in learning of taxonomic relations, evaluation data that is a manually constructed ontological structure of the text documents, and a reference corpus for the *Likey* keyphrase extraction method.

A. Wikipedia articles

The data for concept representation are collected as HTML pages from the English[1] and Finnish Wikipedia[2]. The Wikipedia data are articles about animals (mammals and birds) in both English and Finnish. The data are collected manually using the meta information from both categories and information boxes. The data were collected originally for Finnish such that the length of each article exceeds 200 words to be able to extract keyphrases of sufficient quality, resulting 119 articles. The English articles are the Wikipedia-provided translations of the Finnish articles, resulting 113 articles due to five too short articles and one article that is linked from two Finnish articles. These data sets are available on our web pages.

B. Reference ontology

The evaluation data are collected from the Wikipedia articles. Most of the Wikipedia articles about animals contain a separate meta information box explaining the scientific classification of the animal. This meta information is collected to construct the reference ontology manually. In our evaluation, we use a simplified version of the scientific classification: just three levels of hierarchy are taken into account besides the actual articles. The three levels were inherent in the document collection and it is also the first non-trivial number of levels.

The Wikipedia articles are situated as the leaf nodes, located on the third level of the reference ontology. Their topics are about different Families, Subfamilies or Species, e.g. Black Grouse, Galapagos Hawk and Jaguar. In the English reference ontology there are 113 third-level concepts and in the Finnish ontology 119 concepts. Each third-level concept has a super concept on the second level, which consists of nine different animal Orders (according to the scientific classification) in both languages, e.g. the parent of Black Grouse is Order *Galliformes*, Galapagos Hawk is of Order *Accipitriformes* and Jaguar is of Order *Carnivora* (see Figure 1). There are two concepts on the first level in both languages: the Classes *Mammalia* (parent of e.g. *Carnivora*) and *Aves* (parent of e.g. *Accipitriformes* and *Galliformes*). The root concept of both of the reference ontologies is Kingdom *Animalia*. Both ontologies have 5 subclasses (*Orders*) for *Mammalia* and 4 subclasses for *Aves*. In the Finnish ontology, 84 concepts on the third level (79 in English) belong to Class *Mammalia* and 35 (34) to Class *Aves*.

C. Europarl corpus

The statistical keyphrase extraction method *Likey* requires a reference corpus that is a sample of the general language. We use English and Finnish parts of Europarl, European Parliament plenary speeches [35] as the reference corpus. Our preprocessing excludes all XML tags containing some meta data and results in the sizes of 35 758 149 word tokens in English and 22 676 344 word tokens in Finnish.

IV. Experiments

Our concept hierarchy generation process is based on unsupervised learning. As ontologies are structured by default, in this first approach we decided to select manually the maximum number of clusters on each hierarchy level to simplify the process of comparison to the reference ontology. In our experiments, the maximum was two clusters on the first level and ten clusters (5 + 5) on the second level. If the hierarchical

[1] http://en.wikipedia.org, accessed on 12th January 2010

[2] http://fi.wikipedia.org, accessed on 13th August 2009

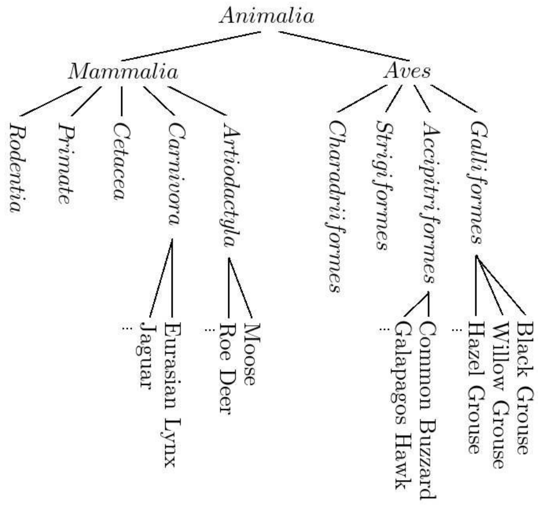


Fig. 1. Part of the reference ontology.

clustering was not able to find the total number of classes, a smaller amount was accepted.

On the first level, we trained a 3x3-cell (5x5 cells in a larger map) *SOM* with normalization of the variance and batch training using SOM Toolbox[3]. On the second level, the map size was slightly larger, 5x5 cells (7x7 cells). These sizes were selected according to our preliminary tests where we found that larger maps (up to 20x20 cells) do not perform very well. Hierarchical clustering was carried out with correlation distance using complete linking scheme. In the preliminary tests, we also used other distance measures (e.g. cosine and spearman) and linking schemes (single and average). Correlation and cosine distance performed very similarly but spearman distance usually slightly worse. Overall, the differences were very small. Single and average linking had the problem that with our implementation in Matlab there were cases where the clustering could not be found.

A. Fuzzy-based representation

In the *efcc* feature extraction approach, the data was preprocessed in the following way. A set of stop words for each language was used to remove common words. The HTML-specific entities were converted to their corresponding characters or discarded, depending on the case. The punctuation marks were also removed. Finally, suffixes were removed using a language dependent stemmer: a standard implementation of Porter's algorithm for English and Snowball Finnish stemmer[4].

The number of features per document vector was chosen to be 100 since that had been enough in our preliminary tests. A second representation for both languages is selected to be approximately the sizes used by the keyphrase extraction method. The second vector size is 913 for English and 1157 for Finnish.

B. Keyphrase extraction

The keyphrase extraction method uses only the plain text portion of the Wikipedia HTML pages without any meta data and is thus a language-independent approach. The Wikipedia preprocessing produces plain text by removing HTML and Wikipedia markup-specific tags, figures, tables, lists, links, and references. Preprocessing for both Wikipedia articles and Europarl reference corpora in Finnish and English removes punctuation and changes numbers to a <NUM> tag. Note that no stemming nor other linguistic filtering takes place in this approach.

We selected the first 15 keyphrases from each article and weighted them with a scaled Likey weight (Eq. 4) that might decrease to zero. We used a threshold value $1/t = 0.01$ since in most of the documents the *Likey ratio* for the first 15 keyphrases is less than that. The resulting feature vectors have 934 features in English, and 1211 features in Finnish.

C. Tf-idf *baseline*

For *tf-idf* the same preprocessing as for *efcc* representation was carried out. The number of features per document vector were selected in the same manner than *efcc* representation, described in section IV-A.

D. Results

The results of the experiments for English Wikipedia documents are given in Table II. The global taxonomic precision (TP), recall (TR) and F-measure (TF) results for the three feature extraction approaches are presented. For *df-efcc-idf* and similarly for *tf-idf*, results for vectors with 100 features (df-efcc-idf 100 and tfidf 100, respectively) and 913 features (df-efcc-idf 913 and tfidf 913, respectively) are shown. Also the results for *Likey* are presented. The results are calculated as averages of the two different map sizes.

TABLE II
RESULTS FOR ENGLISH FOR DIFFERENT REPRESENTATIONS. TP, TR, AND TF STAND FOR GLOBAL TAXONOMIC PRECISION, RECALL AND F-MEASURE, RESPECTIVELY.

Representation	TP	TR	TF
df-efcc-idf 100	0.782	0.900	0.836
df-efcc-idf 913	0.725	0.956	0.825
Likey	0.764	0.886	0.820
tf-idf 100	0.707	0.853	0.772
tf-idf 913	0.698	0.869	0.774

The second experiment is for the Finnish language with the same parameters and representatons than for English, except for the vector size of the second representation of *df-efcc-idf* and *tf-idf* is 1157. The results are presented in Table III.

These results show that for both English and Finnish languages a level of about 80% in Taxonomic F-measure can be achieved within the task of generating hierarchical structures. For the Finnish language, the language-independent feature extraction approach *Likey* does not perform as well as the other approaches. Anyway, the stemming preprocessing step is missing from the *Likey* results and that might explain the poorer performance at least partly.

[3] http://www.cis.hut.fi/somtoolbox/

[4] http://snowball.tartarus.org/algorithms/finnish/stemmer.html

TABLE III
RESULTS FOR FINNISH FOR DIFFERENT REPRESENTATIONS. TP, TR, AND TF STAND FOR GLOBAL TAXONOMIC PRECISION, RECALL AND F-MEASURE, RESPECTIVELY.

Representation	TP	TR	TF
df-efcc-idf 100	0.734	0.872	0.796
df-efcc-idf 1157	0.722	0.852	0.781
Likey	0.685	0.841	0.755
tf-idf 100	0.779	0.847	0.812
tf-idf 1157	0.837	0.865	0.851

For the English language, the heuristic fuzzy logic-based *df-efcc-idf* performs better than the statistical approaches. This is of course a natural result since *df-efcc-idf* exploits more knowledge by using the semantic information of the HTML structure.

The results seem to be not very consistent. This may be due to the fact that a small difference in the *SOM* clustering may have large effect in the resulting ontology. If *SOM* were initialized randomly instead of using the default setting of linear initialization along the two greatest eigenvectors, different results could be obtained on different runs and more precise results reached.

V. Conclusions and discussion

In this paper, we have presented an automated methodology for concept hierarchy generation from a set of text documents. We used three different representations of the documents: 1) a combination of rule-based stemming and fuzzy logic-based feature weighting and selection, 2) automatic keyphrase extraction and 3) the traditional *tf-idf* measure with rule-based stemming. The hierarchy generation has been run by a hierarchical approach of the Self-Organizing Map (*SOM*) together with agglomerative hierarchical clustering. The experiments have been conducted for English and Finnish to show the applicability to different kinds of languages.

We used more than 100 Wikipedia articles about *Animalia* as our data for both English and Finnish. We also created reference ontologies out of the Wikipedia articles for both languages. In the future work, a much larger collection of Wikipedia articles could be used to obtain larger number of levels in the ontology. We also want to exclude the information about the amount of clusters needed for building the hierarchy. Another future improvement consist of extracting concept identifiers from the corpus instead of generating just the taxonomy. Any of our representations could be utilized also here.

References

[1] D. Yuret and M. A. Yatbaz, “The noisy channel model for unsupervised word sense disambiguation,” *Computational Linguistics*, vol. 36, no. 1, pp. 111–127, 2010.

[2] T. Ruotsalo, L. Aroyo, and G. Schreiber, “Knowledge-based linguistic annotation of digital cultural heritage collections,” *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 64–75, 2009.

[3] N. Duran, C. Bellissens, R. Taylor, and D. McNamara, “Quantifying text difficulty with automated indices of cohesion and semantics,” in *Proceedings of the 29th Annual Meeting of the Cognitive Science Society*, 2007, pp. 233–238.

[4] R. Steinberger, F. Fuart, E. van der Goot, C. Best, P. von Etter, and R. Yangarber, “Text mining from the Web for medical intelligence,” in *Mining Massive Data Sets for Security*, F. Fogelman-Souli, D. Perrotta, J. Piskorski, and R. Steinberger, Eds. The Netherlands: IOS Press, 2008.

[5] T. Honkela, V. Könönen, T. Lindh-Knuutila, and M.-S. Paukkeri, “Simulating processes of concept formation and communication,” *Journal of Economic Methodology*, vol. 15, no. 3, pp. 245–259, 2008.

[6] M. Fernández-López and A. Gómez-Pérez, “Overview and analysis of methodologies for building ontologies,” *Knowledge Engineering Review*, vol. 17, pp. 129–156, 2002.

[7] E. P. B. Simperl, C. Tempich, and Y. Sure, “A cost estimation model for ontology engineering,” in *International Semantic Web Conference 2006*, 2006, pp. 625–639.

[8] R. A. Amsler, “A taxonomy for English nouns and verbs,” in *Proceedings of the 19th annual meeting on Association for Computational Linguistics*, Standford, CA, 1981, pp. 133–138.

[9] M. A. Hearst, “Automatic acquisition of hyponyms from large text corpora,” in *Proceedings of the 14th International Conference on Computational Linguistics*, 1992, pp. 539–545.

[10] S. A. Caraballo, “Automatic construction of a hypernym-labeled noun hierarchy from text,” in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*. Association for Computational Linguistics, 1999, pp. 120–126.

[11] M. Pennacchiotti and P. Pantel, “A bootstrapping algorithm for automatically harvesting semantic relations,” *Proceedings of Inference in Computational Semantics (ICoS-06)*, 2006.

[12] S. Ponzetto and M. Strübe, “Deriving a large scale taxonomy from Wikipedia,” in *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, Vancouver, Canada, 2007, pp. 1440–1445.

[13] A. Maedche, V. Pekar, and S. Staab, “Ontology learning part one – on discovering taxonomic relations from the Web,” in *Proceedings of the Web Intelligence conference*, 2002, pp. 301–322.

[14] P. Cimiano and S. Staab, “Learning concept hierarchies from text with a guided hierarchical clustering algorithm,” in *Proceedings of Workshop on Learning and Extending Lexical Ontologies by using Machine Learning Methods at ICML 2005*, 2005.

[15] R. Snow, D. Jurafsky, and A. Y. Ng, “Semantic taxonomy induction from heterogenous evidence,” in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL*. Association for Computational Linguistics, 2006, pp. 801–808.

[16] L. Specia and E. Motta, “A hybrid approach for extracting semantic relations from texts,” in *Proceedings of 2nd Workshop on Ontology Learning and Population*, Sydney, 2006, pp. 57–64.

[17] M. Ciaramita, A. Gangemi, E. Ratsch, J. Saric, and I. Rojas, “Unsupervised learning of semantic relations between concepts of a molecular biology ontology,” in *Proceedings of the 19th international joint conference on Artificial intelligence*, 2005, pp. 659–664.

[18] W. Y. Wong, “Learning lightweight ontologies from text across different domains using the web as background knowledge,” Ph.D. dissertation, The University of Western Australia, September 2009.

[19] E. Gabrilovich and S. Markovitch, “Computing semantic relatedness using Wikipedia-based explicit semantic analysis,” in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, Hyderabad, India, 2007, pp. 1606–1611.

[20] G. Salton, A. Wong, and C. S. Yang, “A vector space model for automatic indexing,” *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[21] A. García-Plaza, V. Fresno, and R. Martínez, “Web page clustering using a fuzzy logic based representation and self-organizing maps,” in *Web Intelligence*, 2008, pp. 851–854.

[22] V. Fresno, “Representacion autocontenida de documentos html: una propuesta basada en combinaciones heuristicas de criterios,” Ph.D. dissertation, DITTE, URJC, 2006.

[23] A. Ribeiro, V. Fresno, M. C. Garcia-Alegre, and D. Guinea, “A fuzzy system for the web page representation,” *Studies in Fuzziness and Soft Computing*, vol. 111, pp. 19–37, 2003.

[24] M.-S. Paukkeri, I. T. Nieminen, M. Pöllä, and T. Honkela, “A language-independent approach to keyphrase extraction and evaluation,” in *Coling 2008: Companion volume: Posters*. Manchester, UK: Coling 2008 Organizing Committee, August 2008, pp. 83–86.

[25] J. Bakus, M. Hussin, and M. Kamel, “A som-based document clustering using phrases,” in *ICONIP*, 2002.

[26] C. Hung and S. Wermter, “Neural network based document clustering using wordnet ontologies,” *Int. J. Hybrid Intell. Syst.*, 2004.

[27] Y. Liu, X. Wang, and C. Wu, "Consom: A conceptional self-organizing map model for text clustering," *Neurocomput.*, 2008.

[28] T. Kohonen, *Self-Organizing Maps*. Springer, 2001.

[29] P. Koikkalainen and E. Oja, "Self-organizing hierarchical feature maps," in *Proceedings of International Joint Conference on Neural Networks*, vol. 2, 1990, pp. 279–285.

[30] M. Dittenbach, D. Merkl, and A. Rauber, "The growing hierarchical self-organizing map," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2000)*, vol. 6, 2000, pp. 15–19.

[31] J. Brank, M. Grobelnik, and D. Mladenic, "A survey of ontology evaluation techniques," in *Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD 2005)*, 2005.

[32] K. Dellschaft and S. Staab, "On how to perform a gold standard based evaluation of ontology learning," *Lecture Notes in Computer Science*, vol. 4273, pp. 228–241, 2006.

[33] Y. Wilks and C. Brewster, "Natural language processing as a foundation of the semantic web," *Foundations and Trends in Web Science*, vol. 1, no. 3–4, pp. 199–327, 2006.

[34] C. Brewster, J. Iria, Z. Zhang, F. Ciravegna, L. Guthrie, and Y. Wilks, "Dynamic iterative ontology learning," *Recent Advances in Natural Language Processing (RANLP 07)*, 2007.

[35] P. Koehn, "Europarl: A parallel corpus for statistical machine translation," in *Proceedings of MT Summit 2005*, 2005.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Metric properties of populations in artificial immune systems using Hadamard representation

Zbigniew Pliszka and Olgierd Unold,

***Abstract*—A Hadamard representation, which is an alternative towards the binary representation, is considered in this study. It operates on numbers $+1$ and -1. Several properties of such defined representation were pointed out and properties of the immune system were expressed based on this representation.**

***Index Terms*—Genetic algorithm, binary coding, Hadamard representation, artificial immune system.**

I. Introduction

DESPITE the continuous development since the 1960s, the discipline of genetic algorithms [5], [11], [1] is still focused more on the empirical aspects of algorithms than theoretical studies. Methods, which are currently in use in theoretical studies of these algorithms, could be classified into one of the following groups: schema theory [7], markov chains theory [10], dimensional analysis [12], order statistics [4], quantitative genetics [8], orthogonal functions analysis [3], quadratical dynamical systems QDS [14], statistical physics [2].

Despite the fact that sophisticated and complex mathematical models were implemented, neither results were obtained which could have broader field of application (e.g. the schema theory or the Markov chains theory) nor were these methods subject for a vivid discussion. Simplistic assumptions adopted frequently in the theoretical analyses deform the analyzed algorithms in such a way that they question the real connection between the obtained results and the investigated algorithms.

This study undertakes rather unfertile topic of representation of binary chromosomes. In place of the classic, zero-one binary representation, other representation {-1, 1} has been proposed for which the metrics in the binary chromosomes space was determined. This representation, called later the Hadamard representation, allows for an effective proof of the dependence between the whole groups of chromosomes. In the Hadamard representation all chromosomes are equal to the length in a defined metric, and there is no chromosome "'zero"', which often had to be separately discussed in the binary representation due to its distinct properties. Note that in the zero-one binary representation each chromosome can be normalized, and each chromosome has a clearly defined direction except chromosome "'zero"'.

The Hadamard representation was applied for defining properties of the artificial immune systems.

II. Hadamard representation

In 1893, French mathematician Jacques Hadamard in his study *Résolution d'une question relative aux déterminants* [6] presented the properties of a matrix, whose only elements were +1 or -1. In this study, we used this representation as a substitute of a binary representation, omitting the requirement of orthogonal columns pairs. Subject of our deliberation was the following series:

TABLE I
Indexing and representation of points in A_n space.

Element's symbol	Decimal representation	Binary representation	Hadamard representation
h_1	0	(0,0,. . . ,0,0,0)	(1, 1,. . . , 1, 1, 1)
h_2	1	(0,0,. . . ,0,0,1)	(1, 1,. . . , 1, 1,-1)
h_3	2	(0,0,. . . ,0,1,0)	(1, 1,. . . , 1,-1, 1)
$\vdots$	$\vdots$	$\vdots$	$\vdots$
h_{2^n-2}	$2^n - 3$	(1,1,. . . ,1, 0, 1)	(-1,-1,. . . ,-1, 1,-1)
h_{2^n-1}	$2^n - 2$	(1,1,. . . ,1, 1, 0)	(-1,-1,. . . ,-1,-1, 1)
h_{2^n}	$2^n - 1$	(1,1,. . . ,1, 1, 1)	(-1,-1,. . . ,-1,-1,-1)

$$A_n = \{(h_n, h_{n-1}, \ldots, h_2, h_1) : \forall i \in \{1, 2, \ldots, n\} \quad h_i \in \{-1, 1\}\} \tag{1}$$

Its elements represent all possible binary chromosomes of equal length n. We will be considering in our work that 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 coordinates of each element of the A_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 A_n, as well as determination of the distance (level of differentiation) between the elements of this space.

At the beginning, we determined the order of the indexing of points in A_n and their four representations, which we will use alternating (see Table I).

The number of points included in A_n

$$|A_n \,| = 2^n \tag{2}$$

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. The same can be said about the Hadamard representation in relation to the binary representation. Three pairs of such functions are presented in Table II.

The distance of points in A_n

The distance of two points $a = (a_n, \ldots, a_2, a_1)$ and $b =$

978-83-60810-27-9/09/$25.00 © 2010 IEEE

TABLE II

EXEMPLARY FUNCTIONS TRANSFORMING THE HADAMARD REPRESENTATION INTO THE BINARY REPRESENTATION AND INVERSELY.

For binary representation	For Hadamard representation
$B_1(h_{j,i}) = \begin{cases} 0 & for \quad h_{j,i} = 1 \\ 1 & for \quad h_{j,i} = -1 \end{cases}$	$H_1(b_{j,i}) = \begin{cases} 1 & for \quad b_{j,i} = 0 \\ -1 & for \quad b_{j,i} = 1 \end{cases}$
$B_2(h_{j,i}) = \frac{1-h_{j,i}}{2}$ where $h_{j,i} \in \{-1,1\}$	$H_2(b_{j,i}) = 1 - 2b_{j,i}$ where $b_{j,i} \in \{0,1\}$
$B_3(h_{j,i}) = \log_{-1} h_{j,i}$ where $h_{j,i} \in \{-1,1\}$	$H_3(b_{j,i}) = (-1)^{b_{j,i}}$ where $b_{j,i} \in \{0,1\}$

$(b_n, \ldots, b_2, b_1)$ in A_n space is measured according to the following equation:

$$\forall a, b \in A_n \qquad w(a,b) = n - \frac{1}{4}\sum_{i=1}^{n}(a_i + b_i)^2 \tag{3}$$

The distance defined in that way will always be a positive integer, which will tell us on what coordinates in Hadamard representation the values differ (exactly as in the binary representation) (see Table I). In addition, A_n space with the w distance determined in this way is metric.

Proof:

1. (identity of indiscernibles)

$$w(a,a) = n - \frac{1}{4}\sum_{i=1}^{n}(a_i + a_i)^2 = n - \frac{1}{4}\sum_{i=1}^{n}4(a_i)^2 = n - n = 0$$

and at the same time

$w(a,b) = 0 \Rightarrow n = \frac{1}{4}\sum_{i=1}^{n}(a_i + b_i)^2$. Therefore, for $a_i, b_i \in \{-1,1\}$, $\forall i \in \{1,2,...,n\} \quad a_i = b_i$, we finally arrive at the equation $a = b$.

2. (symmetry)

$$w(a,b) = n - \frac{1}{4}\sum_{i=1}^{n}(a_i + b_i)^2 = n - \frac{1}{4}\sum_{i=1}^{n}(b_i + a_i)^2 = w(b,a).$$

3. (triangle inequality)

$$w(a,b) + w(b,c) =$$

$$= n - \frac{1}{4}\sum_{i=1}^{n}(a_i + b_i)^2 + n - \frac{1}{4}\sum_{i=1}^{n}(b_i + c_i)^2 =$$

$$= 2n - \frac{1}{4}(\sum_{i=1}^{n}(a_i + b_i)^2 + \sum_{i=1}^{n}(b_i + c_i)^2) =$$

$$= 2n - \frac{1}{4}\sum_{i=1}^{n}\left(a_i^2 + 2b_i\left(a_i + b_i + c_i\right) + c_i^2\right)$$

Due to $a_i, b_i, c_i \in \{-1,1\}$, we obtain the weak inequality $b_i\left(a_i + b_i + c_i\right) \leq a_i c_i + 2$. Therefore,

$$\geq 2n - \frac{1}{4}\sum_{i=1}^{n}\left(a_i^2 + 2\left(a_i c_i + 2\right) + c_i^2\right) =$$

$$= 2n - \frac{1}{4}\sum_{i=1}^{n}\left(a_i^2 + 2a_i c_i + c_i^2\right) - n =$$

$$= n - \frac{1}{4}\sum_{i=1}^{n}\left(a_i^2 + 2a_i c_i + c_i^2\right) = w(a,c).$$

Hence, we have $w(a,b) + w(b,c) \geq w(a,c)$.

■

Remark R1

For any point established in A_n space, there are $\begin{pmatrix} n \\ n-z \end{pmatrix} = \begin{pmatrix} n \\ z \end{pmatrix}$ different points at exactly z distance.

A set of points in A_n with w metrics constitute the limited metric space of diameter n, it can be easily observed that for any two points $h_t, h_k \in A_n$ there is an inequality

$$w(h_t, h_k) \leq n, \tag{4}$$

and the equality occurs only for $k = 2^n - t + 1$, which is presented in Theorem **T1**.

Lemma L1

$$\forall h_t, h_k \in A_n \qquad w(h_t, h_k) = n \Leftrightarrow$$
$$\forall i \in \{1,2,...,n\} \qquad h_{t,i} = -h_{k,i}$$

Proof:

$$w(h_t, h_k) = n$$

$$\Updownarrow$$

$$n - \frac{1}{4}\sum_{i=1}^{n}(h_{t,i} + h_{k,i})^2 = n$$

$$\Updownarrow$$

$$\sum_{i=1}^{n}(h_{t,i} + h_{k,i})^2 = 0$$

The last equation is true if and only if all the elements of the sum are equal to 0. A single element of the sum $(h_{t,i} + h_{k,i})^2$ is equal to 0 if and only if $h_{t,i} = -h_{k,i}$. ■

Theorem T1

$$\forall t \in \{1,...,n\} \qquad w(h_t, h_k) = n \Leftrightarrow$$
$$ID(h_k) = k = 2^n - t + 1 = 2^n - ID(h_t) + 1$$

where $ID(h_i)$ returns index of h_i.

Proof:

$$\Rightarrow$$

From the Lemma L1, we have $h_{t,i} = -h_{k,i}$. As $h_{j,i} \in \{-1, 1\}$, we conclude that at i-th position one of the element has 1, and the second one -1. Transforming h_t and h_k to a binary representation, we obtain two numbers having the feature, that at i-th position one of the number has 0, and the second one 1. The value of the sum of such elements is equal to $2^n - 1$. At the same time, in a decimal representation h_t and h_k represent $t-1$ and $k-1$, respectively. Therefore,

$$(t-1) + (k-1) = 2^n - 1,$$

which is equivalent to

$$k = 2^n - t + 1.$$

$$\Leftarrow$$

Note that the sum of h_t and h_k in a decimal representation is given by

$$(t-1) + (k-1) = (t-1) + (2^n - t) = 2^n - 1$$

Hence, the sum in a binary representation has n-digits 1. Therefore, the components of this sum are the two binary numbers with the property, that at each i-th position one of the components has 1, and the second one 0. Transforming the components into Hadamard representation, we obtain two numbers with the property, that at each i-th position one of the components has -1, and the second one -1. Therefore,

$$h_{t,i} + h_{k,i} = 0$$

$$\frac{1}{4}\sum_{i=1}^{n}(h_{t,i} + h_{k,i})^2 = 0$$

$$w(h_t, h_k) = n$$

■

The points which comply with the above theorem will be called *polar points* and designated as a and $\overline{a}$ pairs. Thus, following equations for the polar points are received:

$$\forall a \in A_n \quad w(a, \overline{a}) = n \tag{5}$$

$$\forall a \in A_n \quad \overline{\overline{a}} = a \tag{6}$$

We can assume:

$$\overline{a} = (\overline{a_n}, \ldots, \overline{a_1}) \tag{7}$$

Lemma **L1** can be presented as follows:

Lemma L2

$$\forall h_k \in A_n \quad \forall i \in \{1, 2, ..., n\} \quad h_{k,i} = -\overline{h_{k,i}}$$

Based on the **R1** remark and **T1** Theorem, it could be concluded that for each point h_t in A_n space, exactly one point h_k occurs in this space, different from h_t, which gives a pair of polar points. Polar points have the following extra two properties

$$\forall c \in A_n \quad \forall a \in A_n \quad w(a, c) + w(c, \overline{a}) = n$$

$$\forall a, b \in A_n \quad w(a, b) = w(\overline{a}, \overline{b})$$

Proof:

$$\forall c \in A_n \quad \forall a \in A_n \quad w(a, c) + w(c, \overline{a}) = n$$

Note that

$$w(a, c) + w(c, \overline{a}) = n$$

$$\Updownarrow$$

$$n - w(a, c) + n - w(c, \overline{a}) = n$$

$$n - w(a, c) + n - w(c, \overline{a}) = \frac{1}{4}\sum_{i=1}^{n}(a_i + c_i)^2 + \frac{1}{4}\sum_{i=1}^{n}(c_i + \overline{a_i})^2 =$$

$$= \frac{1}{4}\sum_{i=1}^{n}(a_i^2 + 2(a_i + c_i + \overline{a}_i)c_i + (\overline{a}_i)^2) =$$

from the Lemma L2, for each $i \in \{1, \ldots, n\}$ we have $a_i + \overline{a_i} = 0$, and $a_i^2 = (\overline{a_i})^2 = c_i^2 = 1$. Hence,

$$= \frac{1}{4}\sum_{i=1}^{n} 4 = n.$$

■

Proof:

$$\forall a, b \in A_n \quad w(a, b) = w(\overline{a}, \overline{b})$$

From Theorem T2 we have

$$w(a, b) + (b, \overline{a}) = n$$

$$w(b, \overline{a}) + w(\overline{a}, \overline{b}) = n$$

From the equality of the right sides, follows the equality of the left sides:

$$w(a, b) + w(b, \overline{a}) = w(b, \overline{a}) + w(\overline{a}, \overline{b})$$

Hence,

$$w(a, b) = w(\overline{a}, \overline{b}).$$

■

After translation to the indices of points of the A_n space, the equation demonstrates another symmetry, besides the one, which results from the second condition of the metric space:

$$w(h_k, h_t) = w(h_{2^n - t + 1}, h_{2^n - k + 1}) \tag{8}$$

The distance $w(h_k, h_t)$ between two points, h_k i h_t, can be estimated according to the **Pod_Od** algorithm. We reduce indices k and t by 1 and divide n times by 2, saving the remainders. The number of differences between the remainders will constitute the sought distance.

Pod_Od Algorithm

```
Input data:
n - number of positions in representation of point from $A_n$
k, t - indices of points $h_k$ and $h_t$
  begin
    od:=0;
    k:=k-1; t:=t-1;
    i:=0;
    while i < n do
    begin
      od:=od+((k mod 2)-(t mod 2))$^2$;
      i:=i+1;
      k:=k div 2;
      t:=t div 2;
    end;
    return od;
  end.
```

During estimation of the distance between two points belonging to the A_n space, the following theorem can be useful.

Theorem T2

$\forall s \in \{0, 1, \ldots, n-1\}$ and $\forall k, t \in \{1, 2, \ldots, 2^s\}$ with $h_k, h_t \in A_n$

$$w(h_k, h_{t+2^s}) = w(h_k, h_t) + 1 \tag{9}$$

$$w(h_{k+2^s}, h_{t+2^s}) = w(h_k, h_t) \tag{10}$$

$$w(h_{k+2^s}, h_{t+2^s}) = w(h_{t+2^s}, h_k) - 1 \tag{11}$$

Proof:

For each $s \in \{0, 1, ..., n-1\}$, if $j = ID(h_j)$ satisfies the inequality $2^{s-1} < j \leq 2^s$ (or equivalent $2^s < j + 2^s \leq 2^{s+1}$), the elements h_j and h_z, where $z = j + 2^s$, differs only at the position $s + 1$. The element h_j at this position has the value 1 ($h_{j,s+1} = 1$), and the element h_z -1 ($h_{z,s+1} = -1$). Therefore, two elements h_t and h_z, where $z = t + 2^s$ while meeting the assumptions of Theorem T2 about s and t, differ at exactly one position only. Hence, if h_k is remote from the element h_t with $w(h_k, h_t)$, then the distance from h_k to the element h_z will be greater by one (according to the Equation (3)), what proves Equation (9). In case of Equation (10), a difference in the value of the coordinate s occurs at the same time in both pairs, i.e., with the k indexes, as well as with t indexes. The simultaneous change in a value on the same coordinate, according to the Equation (3), does not change the value $w(h_k, h_t)$, what proves Equation (10). Equation (11) is to be obtained by using the Equations (10) and (9) and the symmetry rule

$$w(h_{k+2^s}, h_{t+2^s}) = w(h_k, h_t) =$$

$$= w(h_k, h_{t+2^s}) - 1 = w(h_{t+2^s}, h_k) - 1$$

■

Theorem **T2** allows for the construction of an algorithm, which produces the table of distances between any elements from the A_n space. In the **Tab_Od** algorithm, a table was built mechanically (without the estimation of the distances between particular elements). The number situated on the cross of the k-row with the t-column corresponds to the $w(h_k, h_t)$ distance. Therefore, a table of dimension $2n$ could be obtained from the table of dimension n according to the following symbolic equation:

$$\langle W_n \rangle \overset{\mapsto}{\longrightarrow} \left\langle \begin{array}{cc} W_n & \langle W_n + 1 \rangle \\ \langle W_n + 1 \rangle^T & W_n \end{array} \right\rangle$$

where +1 means addition to each element of the table a value of 1.

Tab_Od Algorithm

```
Input data: Enter_size_of_gene  $m$;
  begin
    n:=1;
    T[n,n]:=0;
    while n<2$^m$ do
    begin
      n:=2*n; {enlarging the table twice}
      {extending all rows, copying the already existing one
      and adding 1 to each expression}
      for i:=1 to (n div 2) do
        for j:=(n div 2)+1 to n do
          T[i,j]:=T[i,j-(n div 2)]+1;

      {copying the new part symmetrically, below the first part}
      for i:= n/2+1 to n do
        for j:=1 to n/2 do T[i,j]:=T[j,i];

      {Pasting a copy of the original square under the calculated
      in the beginning of the loop, as the last part of a new,
      twice as large square}
      for i:= n/2+1 to n do
        for j:=(n div 2)+1 to n do T[i,j]:=T[i-(n div 2),j-(n div 2)];
    end;
  end.
```

The conditions for the distance alignment for $n = 4$ are presented in Table III. The elements were placed according to the order of indices increase. It is worth to mention that the sum of k and $2^n - (k-1)$ indices is constant and equals $2^n + 1$ (with accordance to the **T1** Theorem). Thus, elements calculated in this way constitute a pair of polar points (the distance between polar points in our example is $n = 4$, which lies along the diagonal starting in the right upper corner).

III. Definitions Describing States of Artificial Immune Systems

Artificial immune systems constitute currently a significant trend in the studies on biologically inspired calculations [13]. Idealized states of the artificial immune system are determined in the study and subsequently defined using Hadamard representation. Properties determined in such a way are illustrated by the examples based on the content of Table III.

Radius of tolerance R

A radius of tolerance is understood as the border value enabling a mutual recognition of elements in A_n space.

Two elements $x, y \in A_n$ recognize or do not tolerate each other if the distance between them is higher than the radius of tolerance.

$$w(x, y) > R \tag{12}$$

where R complies with the inequality: $0 \leq R \leq n$.

Elements $x, y \in A_n$ complying the weak inequality

$$w(x, y) \leq R \tag{13}$$

will be described as not recognizing or tolerating each other.

Example 0

In the examples considered here we use the A_n space, whose

TABLE III
TABLE OF DISTANCES BETWEEN ANY ELEMENTS IN A_n SPACE $(n = 4)$.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	0	1	1	2	1	2	2	3	1	2	2	3	2	3	3	4
2	1	0	2	1	2	1	3	2	2	1	3	2	3	2	4	3
3	1	2	0	1	2	3	1	2	2	3	1	2	3	4	2	3
4	2	1	1	0	3	2	2	1	3	2	2	1	4	3	3	2
5	1	2	2	3	0	1	1	2	2	3	3	4	1	2	2	3
6	2	1	3	2	1	0	2	1	3	2	4	3	2	1	3	2
7	2	3	1	2	1	2	0	1	3	4	2	3	2	3	1	2
8	3	2	2	1	2	1	1	0	4	3	3	2	3	2	2	1
9	1	2	2	3	2	3	3	4	0	1	1	2	1	2	2	3
10	2	1	3	2	3	2	4	3	1	0	2	1	2	1	3	2
11	2	3	1	2	3	4	2	3	1	2	0	1	2	3	1	2
12	3	2	2	1	4	3	3	2	2	1	1	0	3	2	2	1
13	2	3	3	4	1	2	2	3	1	2	2	3	0	1	1	2
14	3	2	4	3	2	1	3	2	2	1	3	2	1	0	2	1
15	3	4	2	3	2	3	1	2	2	3	1	2	1	2	0	1
16	4	3	3	2	3	2	2	1	3	2	2	1	2	1	1	0

distance tables are presented in Table III. Moreover, for all the demonstrated examples we assume the value of the radius of tolerance $R = 2$.

Self-aggression

System $B_k \subseteq A_n$ undergoes self-aggression if elements x, y occur, which recognize each other and belong to this system.

$$\exists x, y \in B_k \ : \quad w(x, y) > R$$

Example 1

In A_4 , the systems undergoing self-aggression are for example:
$B_8 = \{h_1, h_2, h_3, h_5, h_9, h_4, h_6, h_7\}$ where $w(h_2, h_7) = 3$
$B_4 = \{h_4, h_6, h_7, h_{10}\}$ where $w(h_7, h_{10}) = 4$

System $B_k \subseteq A_n$ is free of self-aggression if any two elements belonging to this system do not recognize themselves.

$$\forall x, y \in B_k \qquad w(x, y) \leq R$$

Example 2

Free systems of self-aggression, when $R = 2$:
$B_5 = \{h_1, h_2, h_3, h_5, h_9\}$
$B_3 = \{h_4, h_6, h_7\}$
$B_2 = \{h_1, h_2\}$

Let us notice that system B_2 is free of self-aggression also when $R = 1$.

Striking distance

A striking distance of a system $B_k \subseteq A_n$ is a series of points of A_n recognized by any point of B_k .

$$P(B_k) = \{z \in A_n : \quad \exists x \in B_k \ \wedge \ w(x, z) > R\}$$

Example 3

For $B_3 = \{h_4, h_6, h_7\}$ from the **Example 2** the striking distance is:

$$P(B_3) = \{h_2, h_3, h_5, h_9, h_{11}, h_{12}, h_{13}, h_{14}, h_{15}\}$$

If B_k undergoes self-aggression then some points belonging to B_k simultaneously belong to $P(B_k)$, which means that

$$B_k \cap P(B_k) \neq \emptyset$$

Example 4

In such state occurs system $B_4 = \{h_4, h_6, h_7, h_{10}\}$ from the **Example 1**:

$$P(B_4) = \{h_2, h_3, h_5, h_7, h_8, h_9, h_{10}, h_{11}, h_{12}, h_{13}, h_{14}, h_{15}\}$$

thus:

$$B_4 \cap P(B_4) = \{h_7, h_{10}\} \neq \emptyset$$

Otherwise, if B_k is free of self-aggression, then B_k and $P(B_k)$ are disjunctive series, which can be presented as follows:

$$B_k \cap P(B_k) = \emptyset$$

Example 5

Free system of self-aggression is B_3, described in **Example 2** and **3**, for which identity occurs:

$$B_3 \cap P(B_3) = \emptyset$$

Complete system

System B_k is complete if its striking distance contains its whole completion $\overline{B_k} = A_n \backslash B_k$.

$$\overline{B_k} \subseteq P(B_k)$$

Example 6

The conditions of the complete system are fulfilled by $B_5 = \{h_1, h_2, h_3, h_4, h_5\}$, for which following identities occur:

$$P(B_5) = \{h_4, h_5, h_6, h_7, h_8, h_9, h_{10}, h_{11}, h_{12}, h_{13}, h_{14}, h_{15}\}$$

$$\overline{B_5} = \{h_6, h_7, h_8, h_9, h_{10}, h_{11}, h_{12}, h_{13}, h_{14}, h_{15}\},$$

Thus, a relation occurs:

$$\overline{B_5} \subseteq P(B_5)$$

Balanced system

System B_k is balanced if at the same time it is a system free of self-aggression, and complete.

$$\overline{B_k} = P(B_k)$$

Example 7
This time let us assume that $B_5 = \{h_1, h_2, h_3, h_5, h_9\}$. For such a system the following identities are fulfilled:

$$P(B_5) = \{h_4, h_6, h_7, h_8, h_{10}, h_{11}, h_{12}, h_{13}, h_{14}, h_{15}\}$$

$$\overline{B_5} = \{h_4, h_6, h_7, h_8, h_{10}, h_{11}, h_{12}, h_{13}, h_{14}, h_{15}\}$$

and

$$\overline{B_5} = P(B_5)$$

Extensive system
We call $B_k \subseteq A_n$ an extensive system if any crossing of its elements results in offspring, which also belongs to this system.

$$\forall x, y \in B_k \subseteq A_n \quad \forall c \in \{0, 1, \ldots, n\} \quad K(\{x, y\}, c) \subseteq B_k$$

where $K(\{x, y\}, c)$ denotes the crossing operation of the two elements x, y from the A_n space in the point c, and is defined as follows:

Let $r_t, r_k \in A_n$, where $r_t = (r_{t,n}, r_{t,n-1}, \ldots, r_{t,2}, r_{t,1})$, $r_k = (r_{k,n}, r_{k,n-1}, \ldots, r_{k,2}, r_{k,1})$.

For $0 \leq c \leq n$,

$$K(\{r_t, r_k\}, c) \mapsto \{(r_{t,n}, r_{t,n-1}, \ldots, r_{t,c+1}, r_{k,c}, r_{k,c-1}, \ldots, r_{k,1}), (r_{k,n}, r_{k,n-1}, \ldots, r_{k,c+1}, r_{t,c}, r_{t,c-1}, \ldots, r_{t,1})\} \quad (14)$$

Example 8
The examples of the extensive systems are presented below:

$$B_2 = \{h_1, h_2\}$$

$$B_4 = \{h_1, h_2, h_3, h_4\}$$

To check the extensibility of B_2 and B_4 systems, equations given by [9] can be used. It can be noticed that both the singleton system and any A_n space, as a whole, are extensive systems.

Expansive system
A system is expansive if it possesses elements (not necessary different), which after a peculiar crossing produce elements out of the system.

$$\exists x, y \in B_k \subseteq A_n \quad \exists c \in \{0, 1, \ldots, n\} : \quad K(\{x, y\}, c) \not\subseteq B_k$$

Example 9
Let us assume $B_3 = \{h_1, h_2, h_3\} \subseteq A_4$. After crossing on chromosomes h_2 and h_3, performing cuts between the first and the second allele, we obtain chromosomes h_1 and h_4. Chromosome $h_4 \notin B_3$ represents an expansive character. In the study [9], equations allowing calculation of crossings' results directly on chromosomes indices, without the need to recreate their internal structure, could be found. The conclusion of those equations is:

$$K(\{h_2, h_3\}, 1) \mapsto \{h_1, h_4\} \not\subseteq B_3$$

IV. Summary

In this study a so called Hadamard representation was implemented. It allows to prove the dependence between subsequent generations of binary chromosomes. Some properties of this representation were pointed out, which allows for quick and simple operations on chromosomes indices, instead of processing the binary sequences.

Hadamard representation was applied in a brief definition of some properties of the artificial immune systems. This representation will also allow employing natural numbers in calculating the results of such operations like crossing, mutation, or genetic inversion, as well as determining the influence of these operations on the whole concerned population.

Introduced concepts allow a distinction and classification of different populations, which allows us to ponder the future potential directions of their evolution (states reachable, unreachable, etc.) regardless of the crossing algorithm. Further, it should allow us to compare the genetic algorithms in terms of effectiveness and optimization! Comparing the two algorithms, we must ensure comparability of the populations in which we conduct experiments. It is obvious that the same algorithm for example in the population of the class of expansive systems, has a chance of finding new solutions in subsequent generations, but populations with extensive class, after reviewing the current population, better solutions will no longer find. Self-aggression systems have dispersed chromosomes in the space, as opposed to the free systems of self-aggression, which are concentrated in the vicinity of a chromosome. For such systems we can have a suspicion that for the purpose of continuous functions we have to deal with a local extremum. In the case of complete systems, we can be sure that we control the entire space under consideration, although we use only a separate part of the chromosomes of that space. It is important in many cases to set minimum-complete systems for the space in question.

The presentation of the majority of the basic genetic operators known in the literature as Hadamard representation is planned, as well as studies on the analytical demonstration of temporal properties of the genetic operators and the artificial immune system expressed an {+1, -1} notation will be carried out.

References

[1] Arabas J. (2004), *Wykłady z algorytmów ewolucyjnych,* PWN, Warszawa (in Polish).

[2] Asselmeyer T., Ebeling W. (1997), *Unified description of evolutionary strategies over continuous parameter spaces*, BioSystems 41(3), 167–178.

[3] Bethke A.D. (1980), *Genetic algorithms as function optimizers*, PhD Thesis, University of Michigan.

[4] Beyer H.G. (1995), *Toward a theory of evolution strategies: On the benefits of sex – the* (μ/μ, λ) *theory*, Evolutionary Computation 3(1), 81–111.

[5] Goldberg D.E. (1998), *Algorytmy genetyczne i ich zastosowania*, WNT, Warszawa (in Polish).

[6] Hadamard J. (1893), *Résolution d'une question relative aux déterminants*, Bull. Sci. Math. 2, 240-246.

[7] Holland J. (1992), *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge (MA), 2nd edition.

[8] Mühlenbein H., Schlierkamp-Voosen D. (1995), *Analysis of Selection, Mutation and Recombination in Genetic Algorithms*, In: Evolution as a Computational Process, eds. W. Banzhaf, F.H. Eeckman, LNCS 899, Springer, Berlin, Heidelberg, 188–214.

[9] Pliszka Z., Unold O. (2007), O pewnych własnościach operatorów genetycznych i stanach sztucznego systemu immunologicznego w reprezentacji Hadamarda, Raporty Inst. Inform. Autom. Robot. PWroc. 2007 Ser. PRE nr 71 (in Polish).

[10] Rudolph G. (1998), *Finite Markov chain results in evolutionary computation: A tour d'horizon*, Fundamenta Informaticae, 34, 1–22.

[11] Rutkowska D., Piliński M., Rutkowski L. (1997), *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*, PWN, Warszawa (in Polish).

[12] Thierens D. (1996), *Dimensional analysis of allele-wise mixing revisited*, w: Parallel Problem Solving From Nature, PPSN IV, red. H.M. Voigt et al., Springer, Berlin, Heidelberg, 225–265.

[13] Wierzchoń S.T. (2001), *Sztuczne systemy immunologiczne. Teoria i zastosowania*, Akademicka Oficyna Wydawnicza EXIT, Warszawa (in Polish).

[14] Vose M.D., Liepins G.E. (1991), *Punctuated equilibria in genetic search*, Complex Systems 5(1), 31–44.

ISBN 978-83-60810-27-9
ISSN 1896-7094

The development features of the face recognition system

Rauf Sadykhov
United Institute of Informatics Problems
6, Surganov str., Minsk, Belarus
Email: rsadykhov@bsuir.by

Igor Frolov
Belarusian State University
of Informatics and Radioelectronics
6, P.Brovka str., Minsk, Belarus
Email: frolovigor@yandex.ru

***Abstract*—Nowadays personal identification is a very important issue. There is a wide range of applications in different spheres, such as video surveillance security systems, control of documents, forensics systems and etc. We consider a range of most significant aspects of face identification system based on support vector machines in this paper. At first we propose improved face detector to get the region of interest for next face recognition. In paper the technique of face detection jointly image normalization is introduced. We compare three algorithms of feature extraction in application on face identification (PCA NIPALS, NNPCA, kernel PCA). The presented system is intended for process the image with low quality, the photo with the different facial expressions. Our goal is to develop face recognition techniques and create the system for face identification.**

I. Introduction

THE PERSON identification systems are increasingly becoming popular in modern society. The producers of security systems are interested in the new technologies for the automation of the person identification process. This fact is due to rise the level of these systems reliability because of depreciation of the components of used hardware in the designing and the construction of ones.

The range of the biometric systems identification is wide enough, there're the identification on the fingerprints, the iris identification, the face recognition methods and etc. All these technologies are different by the algorithms, methods and techniques that used for the system development. The considerable quantity of solutions are proposed in the field of face recognition and in the sphere of the person identification by photo. Nowadays the development of the automatic personal identification system is a very important issue because of the wide range of applications in different spheres, such as video surveillance security systems, control of documents, forensics systems and etc.

It should be noted that the process of pattern recognition in the field of image processing consists of several required stages before getting final result. There are the preprocessing of the source patterns (the image data processing such as the readjustment of light conditions, the detection of region of interest, the image resizing), the dimension reduction of source data space by data transformation (to remove the noise and to approximate the data), the selection and the implementation of techniques for the data classification.

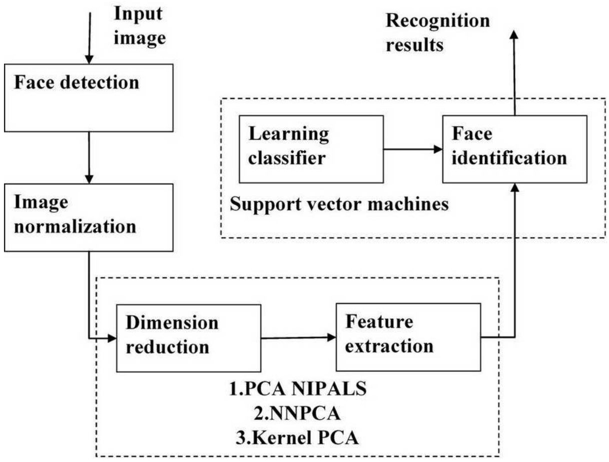


Fig. 1. The structure of face identification system

In this paper we describe the experimental face identification system based on support vector machines (SVM) [1] and we consider some more interesting aspects with designing and constructing of the person system identification by photo. Our system consists of several typical modules (see fig. 1) that are important for the systems of this type such as block of the region of interest (face) detection, block of the image normalization (with the functions of enhancement of brightness and contrast characteristics), the features' extraction block for the dimension reduction of source data space, the module of face recognition (identification) with the functions for the training SVM-classifier and the functions of classification of the processed pattern by the definition of the test image to the certain class.

We researched automated biometric identification systems that were tested on data of the National Institute of Standards and Technology (USA) in particular FERET database. So we propose you to read the results of testing several systems that were developed by different companies (see fig. 2). There are automated portrait identification system "Portrait 2005", developed by specialists LLC "Bars International", LLC "Portland" (Russian Federation) and LLC "ASPI-Soft" (Belarus); automated portrait identification system "Crime Face", developed by RPLLC "Todes" (Belarus); hardware-software complex "Image++"(or"SOVA"), developed by LLP

978-83-60810-27-9/09/$25.00 © 2010 IEEE

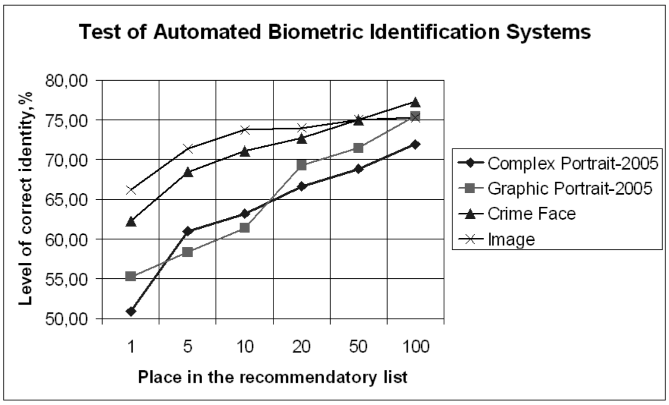


Fig. 2. Test of Automated Biometric Identification Systems

"DANA" (Kazakhstan). Information with test results of automated biometric identification systems that were tested on data of the National Institute of Standards and Technology (USA) is presented at figure 2. In February-July 2006 the State Expert Forensic Center of Ministry of Internal Affairs of Belarus carried out a test of these automated biometric identification systems based on technology of face recognition. The source of this research is [2].

This paper is organized as follows. In the next section the methods of face detection are described. In section III the approaches of the preparation and normalization of the images are introduced. In section IV the neural network approaches as tool for reducing source space of data is considered. The section V contains the description of classifier for pattern recognition based on support vector machines. Finally, the section VI collect some experimental results and brief conclusions.

II. Face detection approaches

A precise detection of face at image strongly simplifies the process of classification. At stage of system development we have realized some experiments and established that at first face detection procedure must be executed due to its importance. The module of image normalization should work at second stage. The results of performance of these procedures in determinate order like this are displayed in a fig. 3 and fig. 4. Presence of background and of facial parts (ears, hair) have effect on obtained results of these experiments. We apply the image enhancement unit in detected region of interest (face in particular) to get more contrasting images that are more suitable for a consequent procedure of reduction of original data space and pattern recognition process. At the beginning the unit of face detection performs detection of facial region using the famous algorithm of Viola-Jones [3]. This method uses an image representation called the "Integral Image" which allows the features used by detector to be computed very quickly [4]. A simple and efficient classifier is built using the AdaBoost learning algorithm [5] to select a small number of critical visual features from a very large set of potential features. Viola-Jones proposed a method for combining classifiers in a "cascade" which allows background regions of the image to be quickly discarded while spending more computation on promising face-like regions. However, the results of initial algorithm are not acceptable to further process of face recognition. We can observe a lot of noised data such as background, hair, clothes (see fig. 5).

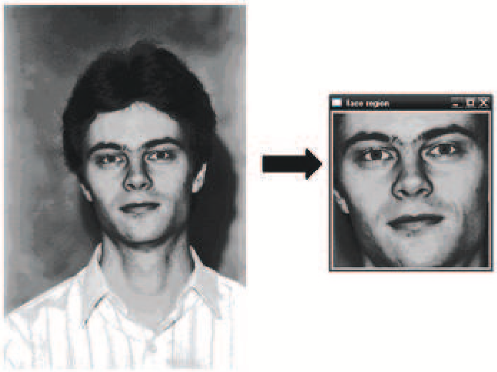


Fig. 3. Image enhancement by "normalization-detect"

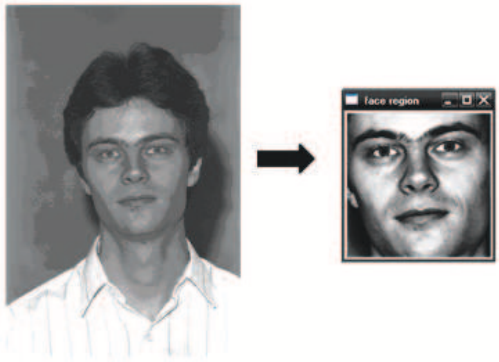


Fig. 4. Image enhancement by "detect-normalization"

These elements of image are not interested for process of pattern recognition. To achieve a higher level of authenticity in face recognition it is necessary to select a more narrow region of interest. We have developed the technique to detect region of face only. Our method intends for extracting facial features only without any noised data. It is based on applying of information about human anthropometric measurements. The results of application of presented algorithm are shown in fig. 7. Our approach is based on discrete adaboost ap-

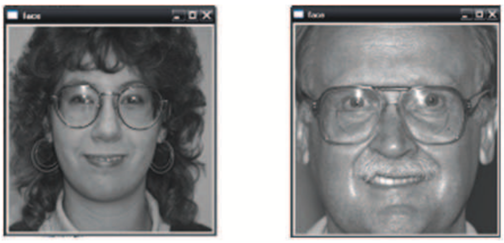


Fig. 5. The face region with the noised data

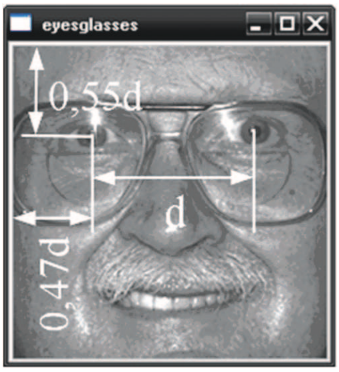


Fig. 6. Facial anthropometric Measurements

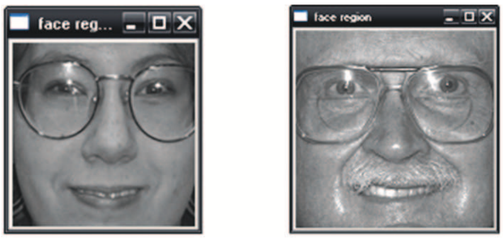


Fig. 7. The region of interest

plication [6] to select simple classifiers based on individual features drawn from a large and over-complete feature set in order to build strong stage classifiers of the cascade. This technique executes the iris area only, not whole face as the previous method. The input image for iris detection procedure is image obtained at the previous stage of image processing that contains parts of clothes, hair etc.

At first we perform the search of left-hand eye area only. The procedure of search of right-hand eye starts if the previous stage was finished successfully. When we get a beneficial effect the distance between pupils in pixels is calculated. At the next step we compute coordinates of left upper point of region of interest. The facial region is presented as squared area with left-upper point calculated at previous stage.

To compute the coordinates of each of four points we use the distance between pupils and some of facial anthropometric data. A length from left iris to upper boundary is calculated as $0,55 \cdot D$ and length from the left iris to the left boundary of the region of interest is equal $0,47 \cdot D$, where D is the distance between pupils in pixels. Thus the length of the side of square of face region is calculated as $2 \cdot 0,47 \cdot D + D = 1,94 \cdot D$. All of these estimated coefficients were obtained experimentally(see fig. 6).

Our approach allows of finding the specified face region on the majority tested images. Failing of iris detection we use additional techniques to solve this problem and to find the region of interest. This technique is enhancement of basic detector and it is designed to find eyeglasses. The current detector was trained to find each part of eyeglasses and after that to detect the iris area. The results of search of eyeglasses are represented in fig. 8(a). The search of region of pair eyes starts after unsuccessful attempt of search of eyeglasses (see fig. 8(b) with results). Our algorithm has found the face regions of interest on all tested images. However, when nothing found at work image we provide the original entry image in the capacity of region of interest.

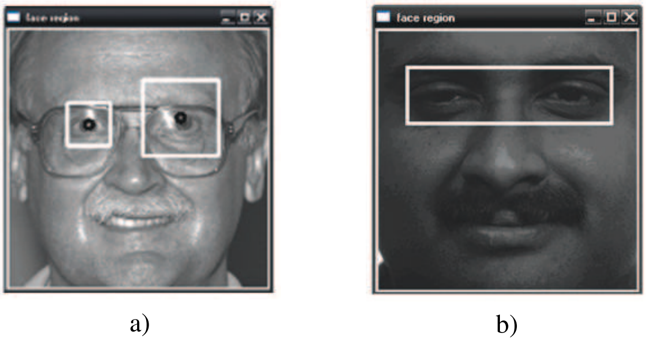


Fig. 8. Face detect with: a)- eyeglasses, b)- pair eyes

The region of interest is presented as squared area which contains necessary data (features of face) with minimal noise level to face recognition process. The initial size of facial region has arbitrary size but then we scale it to size of 169×169 pixels. Square side is chosen subject to several important factors. The most important restriction imposed on the procedure of identification is the quality of processed images, primarily due to the resolution image and due to conditions of exposure. The minimum distance between eyes of frontal images that are used at face identification is determined by international standards as "ISO 19794-5:2005" [7], "ANSI/INCITS 385-2004 (Information technology - Face Recognition Format for Data Interchange)" [8]. This parameter is equal 90 pixels. If we keep constraint about minimal distance between eyes then the square side of region of interest is equal about 170×170 pixels for our system face identification. We have chosen the square side size equal 169 pixels subject to some details of image processing with artificial neural networks PCA.

III. Image enhancement techniques

At next stage our system performs the procedures of image normalization. We perform an expansion of pixels values to the whole intensity range and the equalization of histogram. The first approach maps the values in intensity image to new values such that values between low and high values in current image map to values between 0 and 1. Thus new pixel values allocate to whole intensity range. After that we perform the histogram equalization which enhances the contrast of images by transforming the values in an intensity image so that the histogram of the output image approximately matches a specified histogram. After use these methods image contains some distortions as sharp face lines. That's why we apply the median filter to dither face features. This method performs median filtering of the input image in two dimensions. Each output pixel contains the median value in the 3×3 neighborhood around the corresponding pixel in the input image. This part of image processing removes significantly the illumination

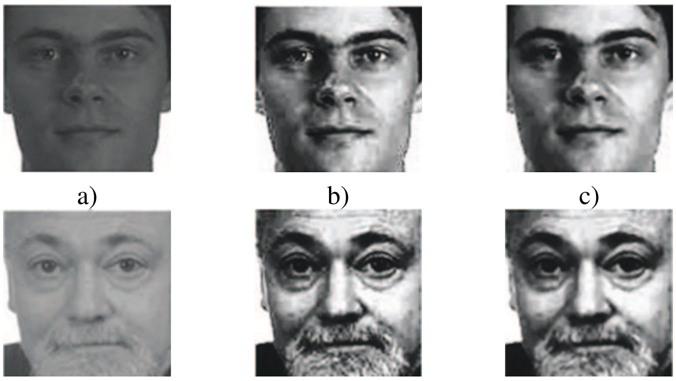


Fig. 9. Examples of normalized images: a - input images, b - after application adjusting image intensity values and histogram equalization, c - after using median filter

changes among the images. The fig. 9 illustrates the results of introduction the image pre-processing methods described above. We use these methods in the following sequence: expansion of pixels values to the whole intensity range - the equalization of histogram - median filtering.

IV. Dimension reduction and feature extraction

Some classification-based methods use the intensity values of window images as input features of classifier. However, direct use of intensity values of image pixels are dramatically increases the computation time. On the other hand the huge capacity of data contains many waste data being overfull. In our system we extract features via method of principal component analysis. We use three techniques for implementation of this approach. There are the algorithm NIPALS (non-linear iterative partial least squares) [9] for compute the principal components, the neural network PCA (NNPCA) [13], [16], and the kernel principal component analysis [17]. These methods have different computational cost and various confidence levels at recognition stage. The user of system can choose the method to work by oneself.

Principal Component Analysis (PCA) - is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension. PCA involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. PCA is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

The NIPALS ("Nonlinear Iterative Partial Least Square") algorithm is one of the many methods that exist for finding the eigenvectors (another example is SVD [10]). It was originally made for PCA, but it has been used in other methods as well. Algorithm is used in principal component analysis to decompose a data matrix into score vectors and eigenvectors (loading vectors) plus a residual matrix. This is an overview of the algorithm:

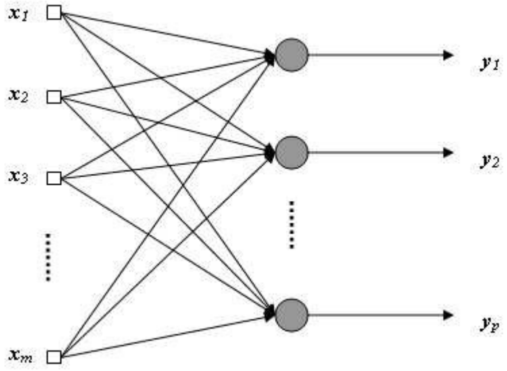


Fig. 10. A single-layered feedforward neural network for extracting p principal components

X is a mean centered data matrix, $E_{(0)} = X$. The E-matrix for the zero-th PC (PC_0) is mean centered X, t vector is set to a column in X, t will be the scores for PC_i, p will be the loadings for PC_i, $threshold = 0,00001$, just a low value, to do the convergence check.

Iterations ($i = 1$ to number-of-PCs):

1.Project X onto t to find the corresponding loading p

$$p = (E_{i-1}^T / t^T t) \tag{1}$$

2.Normalise loading vector p to length 1

$$p = p \cdot p(p^T p)^{-0.5} \tag{2}$$

3.Project X onto p to find corresponding score vector t

$$t = (E_{(i-1)} p / (p^T p)) \tag{3}$$

4.Check for convergence. If difference between eigenvalues $\tau_{new} = (t^T t)$ and τ_{old} (from last iteration) is larger than $threshold \cdot \tau_{new}$ return to step 1.

5.Remove the estimated PC component from $E_{(i-1)}$

$$E_{(i)} = E_{(i-1)} - (tp^T) \tag{4}$$

Principal components can be extracted using single-layer feed-forward neural networks [11]. These networks learn unsupervised by using variants of the Hebbian rule. The Generalized Hebbian Algorithm (GHA) [12], also known in the literature as Sanger's rule, is a linear feedforward neural network model for unsupervised learning with applications primarily in principal components analysis. It is similar to Oja's rule in its formulation and stability, except it can be applied to networks with multiple outputs.

$$y_j(n) = \sum_{i=1}^{p} w_{ij} x_i(n), \qquad j = 1, 2, ..., m. \tag{5}$$

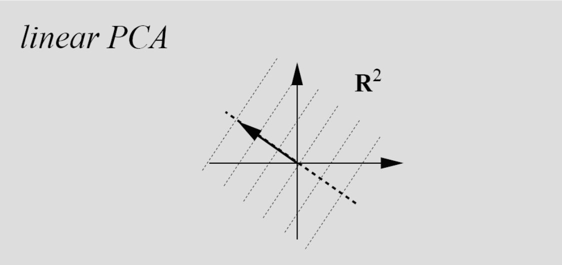


Fig. 11. Linear PCA

$$\Delta w_{ij}(k) = \eta[y_j(n)x_i(n) - y_j(n)\sum_{k=1}^{j} w_{ki}(n)y_k(n)] \qquad (6)$$

The neural network PCA (NNPCA) is used in our work. The Generalized Hebbian Algorithm by Sanger [6] is one among the best known learning algorithms that allow a neural network (see fig. 10) to extract a selected number of principal components from a multivariate random process. It applies to a single-layered feedforward neural network that may be described by equation(5) with the rule for updating weights (see equation 6).

The weight of first eigenvector has been estimated and its value lies within the range from 75 to 84%. Therefore, to decrease the computation expenses we used only one eigenvector for calculation one principal component.

Kernel principal component analysis (kernel PCA) [14] is an extension of principal component analysis using techniques of kernel methods. Instead of directly doing a PCA, the original data points are mapped into a higher-dimensional (possibly infinite-dimensional) feature space defined by a (usually non-linear) function Φ through a mathematical process called the "kernel trick":

$$\Phi : \mathbf{R}^N \to F,\ x \to \mathbf{X} \qquad (7)$$

The kernel trick [15] transforms any algorithm that solely depends on the dot product between two vectors. Wherever a dot product is used, it is replaced with the kernel function. Thus, a linear algorithm can easily be transformed into a non-linear algorithm. This non-linear algorithm is equivalent to the linear algorithm operating in the range space of Φ. However, because kernels are used, the Φ function is never explicitly computed. This is desirable, because the high-dimensional space may be infinite-dimensional (as is the case when the kernel is a Gaussian).

Like in PCA, the overall idea is to perform a transformation that will maximize the variance of the captured variables while minimizing the overall covariance between those variables. Using the kernel trick, the covariance matrix is substituted by the Kernel matrix and the analysis is carried analogously in feature space. An Eigen value decomposition is performed and the eigenvectors are sorted in ascending order of Eigen values, so those vectors may form a basis in feature space that explain most of the variance in the data on its first dimensions.

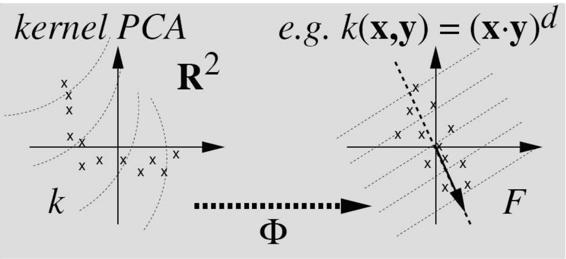


Fig. 12. The basic idea of kernel PCA. In some high dimensional feature space F (see fig. 12 right), we are performing linear PCA, just as a PCA in input space(figure 11). Since F is nonlinearly related to input space (via Φ), the contour lines of constant projections onto the principal Eigenvector (drawn as an arrow) become nonlinear in input space. Note that we cannot draw a pre-image of the Eigenvector in input space, as it may not even exist. Crucial to kernel PCA is the fact that we do not actually perform the map into F, but instead perform all necessary computations by the use of a kernel function k in input space (here: R^2).

However, because the principal components are in feature space, we will not be directly performing dimensionality reduction. Suppose that the number of observations m exceeds the input dimensionality n. In linear PCA, we can find at most n nonzero Eigen values. On the other hand, using Kernel PCA we can find up to m nonzero Eigen values because we will be operating on a $m \times m$ kernel matrix.

Each time the features extracted vector is presented as the sequence of more significant coefficients of the principal components. In our work the size of face region extracted in face detection block is 169×169 pixels. Thus the original dimension of data space counts 28.561 points. We form sequence with 169 features only by use of most important coefficients from process of feature extraction. The second part of data (less significant coefficients) is rejected. Thus we use three different approaches to extract the vector of features from original data set (region of interest that contains a facial image).

V. Face identification with SVMs

The Support Vector Machines (SVMs) [1] present one of kernel-based techniques. SVMs based classifiers [18] can be successfully apply for text categorization, face identification. A special property of SVMs is that they simultaneously minimize the empirical classification error and maximize the geometric margin; hence they are also known as maximum margin classifiers. SVMs are used for classification of both linearly separable (see fig. 13) and unseparable data. SVMs based classifiers can be successfully apply for text categorization, face identification.

Linear classifiers are not complex enough sometimes. SVM solution: map data into a richer feature space including non-linear features, then construct a hyperplane in that space so all other equations are the same. Basic idea of SVMs is creating the optimal hyperplane and calculating the decision function for linearly separable patterns. This approach can be extended to patterns that are not linearly separable by transformations of

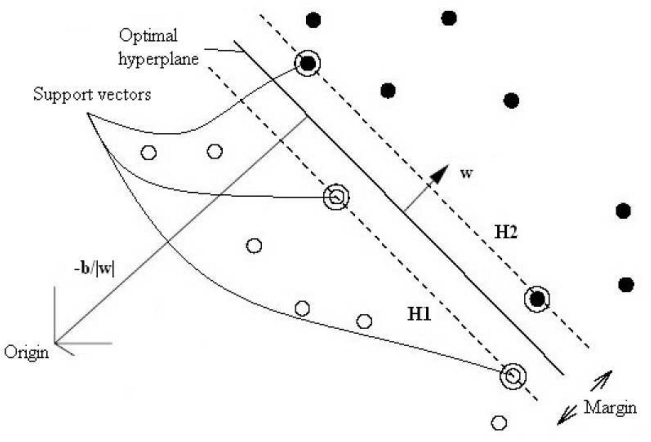


Fig. 13. Linear separating hyperplanes for the separable case.

original data to map into new space due to using "kernel trick". In the context of the Fig. 13, illustrated for 2-class linearly separable data, the design of the conventional classifier would be just to identify the decision boundary w between the two classes. However, SVMs identify support vectors (SVs) $H1$ and $H2$ that will create a margin between the two classes, thus ensuring that the data is "more separable" than in the case of the conventional classifier.

Suppose we have N training data points $(x1, y1), (x2, y2), \ldots, (x_N, y_N)$ where $x_i \in \Re^d$ and $y_i \in \pm 1$. We would like to learn a linear separating classifier:

$$f(x) = sgn(w \cdot x - b) \tag{8}$$

Furthemore, we want this hyperplane to have the maximum separating margin with respect to two classes. Specifically, we wish to find this hyperplane $H : y = w \cdot x - b$ and two hyperplanes parallel to it and with equal distances to it:

$$H_1 : y = w \cdot x - b = +1 \tag{9}$$

$$H_2 : y = w \cdot x - b = -1 \tag{10}$$

with the condition that there are no data points between H_1 and H_2, and the distance between H_1 and H_2 is maximized.

For any separating plane H the corresponding H_1 and H_2 we can always "normalize" the coefficients vector w so that H_1 will be $y = w \cdot x - b = +1$, and H_2 will be $y = w \cdot x - b = -1$ as shown [1].

We want to maximize the distance between H_1 and H_2. So there will be some positive examples on H_1 and some negative examples on H_2. These examples are called support vectors because only they participate in the definition of the separating hyperplane, and other examples can be removed and moved around as long as they do not cross the planes H_1 and H_2.

In the space the distance from a point on H_1 to $H : w \cdot x - b = 0$ is $|w \cdot x - b|/||w|| = 1/||w||$, and the distance between H_1 and H_2 is $2/||w||$. Thus, to maximize the distance we should minimize $||w|| = w^T w$ with the condition that there are no data points between H_1 and H_2:

$$w \cdot x - b \geq +1, \text{ for positive example } y_i = +1 \tag{11}$$

$$w \cdot x - b \leq -1, \text{ for negative example } y_i = -1 \tag{12}$$

These two conditions can be combined into

$$y_i \cdot (w \cdot x_i - b) \geq 1 \tag{13}$$

So, this problem can be formulated as

$$\min_{w,b} \frac{1}{2} w^T w \text{ subject to } y_i \cdot (w \cdot x_i - b) \geq 1 \tag{14}$$

This is a convex quadratic programming problem (in w, b) in convex set.

Introducing Lagrange multipliers $\alpha_1, \alpha_2, \ldots, \alpha_N \geq 0$, we have the following Lagrangian:

$$L(w, b, \alpha) \equiv \frac{1}{2} w^T w - \sum_{i=1}^{N} \alpha_i y_i (w \cdot x_i - b) + \sum_{i=1}^{N} \alpha_i \tag{15}$$

We can solve the wolfe dual insread: maximize $L(w, b, \alpha)$ with respect to α subject to constrains that the gradient of $L(w, b, \alpha)$ with respect to the primal variables w and b vanish:

$$\frac{\partial l}{\partial w} = 0 \tag{16}$$

$$\frac{\partial l}{\partial b} = 0 \tag{17}$$

and that $\alpha \leq 0$

From equations (16) and (17) we have

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \tag{18}$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \tag{19}$$

Substitute them (18), (19) into $L(w, b, \alpha)$ we have

$$L_D \equiv \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j x_i x_j \tag{20}$$

in which the primal variables are eliminated.

When we solve α_i, we can get $w = \sum_{i=1}^{N} \alpha_i y_i x_i$ and we can classify a new object x with:

$$\begin{aligned} f(x) &= sgn(w \cdot x + b) \\ &= sgn((\sum_{i=1}^{N} \alpha_i y_i x_i) \cdot x + b) \\ &= sgn(\sum_{i=1}^{N} \alpha_i y_i (x_i \cdot x) + b) \end{aligned} \tag{21}$$

Note that in the objective function and solution, the training vector x_i is occurred only in the form of dot product.

If the surface separating the two classes are not linear we can transform the data points to another high dimensional space such that the data points will be linearly separable [19]. Let the transformation be $\Phi(\cdot)$. In the high dimensional space, we solve

$$L_D \equiv \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \Phi(x_i) \cdot \Phi(x_j) \tag{22}$$

Suppose, in addition, $\Phi(x_i) \cdot \Phi(x_j) = k(x_i \cdot x_j)$. That is, the dot product in that high dimensional space is equivalent to a kernel function of the input space. So we need not be explicit about the transformation $\Phi(\cdot)$ as long as we know that the kernel function $k(x_i \cdot x_j)$ is equivalent to the dot product of some other high dimensional space. There are many kernel functions that can be used this way, for example, the radial basis function (Gaussian kernel):

$$K(x_i, x_j) = e^{-||x_i - x_j||^2/2\sigma^2} \tag{23}$$

Formally, preprocess the data with $\Phi : \Re^N \rightarrow F$, then a data set that is not linearly separable in the input data space (as in the left hand side of fig. 14) is separable in the nonlinear feature space (right hand side of fig. 14) defined implicitly by the non-linear kernel function.

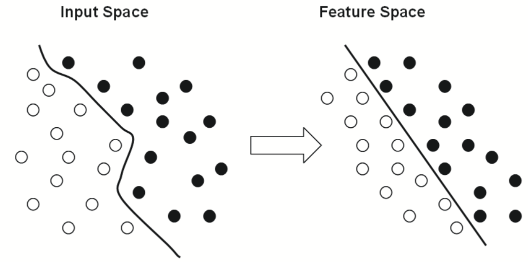


Fig. 14. Kernel trick for the unseparable case

For multi-class classification we use the "one-against-one" approach in which $k \cdot (k-1)/2$ classifiers are constructed and each one trains data from two different classes. In classiffication we use a voting strategy: each binary classiffcation is considered to be a voting where votes can be cast for all data points x - in the end point is designated to be in a class with maximum number of votes. We implemented probabilistic approach to identify the processed pattern by calculating the confidence level for this face. And we construct a list of 6 samples by descending to make a final decision.

Basic idea of SVMs relative to the Nearest Neighbor [20] approach is creating the optimal hyperplane and calculating the decision function for linearly separable patterns. This approach can be extended to patterns that are not linearly separable by transformations of original data to map into new space due to using kernel trick.

To train the SVM, we search trough the feasible region of the dual problem and maximize the objective function. The optimal solution can be checked using the Karush-Kuhn-Tucker (KKT) conditions [1].

The KKT optimality conditions of the primal problem are

$$\alpha_i[y_i(w^T x_i - b) + \xi_i - 1] = 0 \tag{24}$$

$$\sum_{i=1}^{N} \mu_i \xi_i = 0 \tag{25}$$

To solve this quadratic programming problem we used the sequential minimal optimization (SMO)-type decomposition method[21] for support vector machines [22].

The SMO algorithm searches through the feasible region of the dual problem and maximizes the objective function

$$L_D \equiv \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j \tag{26}$$

$$0 \leq \alpha_i \leq C, \quad \forall i$$

It works by optimizing two α_i's at time (with the other α_i's fixed) and uses heuristics to choose the two α_i's for optimization [22].

The decision function is

$$sgn(\sum_{i=1}^{l} y_i \alpha_i K(x_i, x) + b). \tag{27}$$

VI. Experiment results

Our system contains two basic blocks. There are training SVM-classifier module and face identification unit developed on SVM-classifier. At first we have to create the model for following pattern recognition. At this stage we train our SVM-classifier by the algorithm proposed Jones C.Platt [22]. In our system we used the libsvm implementation [23] of this algorithm. The one-type input feature vector containing the significant coefficients from PCA is used both for train and classification.

Scaling data before applying SVM is very important. [24] explains why we scale data while using Neural Networks, and most of considerations also apply to SVM. The main advantage is to avoid attributes in greater numeric ranges dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors, e.g. the linear kernel and the polynomial kernel, large attribute values might cause numerical problems. We linearly scale each attribute to the range $[0; 1]$. Of course we use the same method to scale testing data before testing.

To increase the level of correct identity we applied choose the parameters of C-support vector classification with cross validation via parallel grid search. There are two parameters while using RBF kernels: C and γ. It is not known beforehand which C and γ are the best for one problem; consequently some kind of model selection (parameter search) must be done. Therefore, a common way is to separate training data into two parts of which one is considered unknown in training the classifier. Then the prediction accuracy on this set can more precisely reflect the performance on classifying unknown data. An improved version of this procedure is cross-validation.

In v-fold cross-validation, we first divide the training set into v subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining $v - 1$ subsets. Thus, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly classified.

We used the sample collection of images with size $512 - by - 768$ pixels from database FERET [25] containing 100 classes (unique persons) to test our face recognition system

TABLE I
RESULTS OF TESTING PERSON IDENTIFICATION SYSTEM

	Recognition rate, percent	Feature extraction time for each vector, s	Training time, s
PCA NIPALS	80	0,6	28,4
NNPCA	84	12	28,8
Kernel PCA	81	0,8	28,3

based on support vector machines. This collection counts 300 photos. Each class was presented by 3 images. So, to train SVM-classifier we used 200 images where 2 photos introduced each class. 100 images were used to test our system. Note, that any image for testing doesn't use in training process. The results of realized experiments are shown in the table I. In this paper we proposed an efficient face identification system based on support vector machines. This system performs several algorithms to ensure the full process of pattern recognition. Thus, our system is intended for face identification by processing the image even low quality. The face detection region procedure without any noise is a very important stage of the person identification process. The angle of inclination and the rotation angle of head influence on the level of validity of recognition. These factors are the most significant in person identification system.

REFERENCES

[1] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, vol. 2, 1998, pp.121-167.

[2] http://www.portret.tomsk.ru/index.php?page=informations&subject=gabitoskopia

[3] P. Viola, M. J. Jones, "Robust Real-Time Face Detection", *International Journal of Computer Vision*, vol. 57 (2), 2004, pp.137-154.

[4] Bae, H. and S. Kim, "Real-time face detection and recognition using hybrid-information extracted from face space and facial features", *Image and Vision Computing*, vol. 23, 2005, pp.1181-1191.

[5] K. Tieu, P. Viola, "Boosting image retrieval", *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[6] R. E. Schapire, Y. Freund, "A short introduction to boosting", *Journal of Japan Society for Artificial Intelligence*, vol. 5 (14), 1999, pp.771-780.

[7] http://www.iso.org/iso/catalogue_detail.htm?csnumber=38749

[8] http://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+INCITS+385-2004

[9] H. Risvik, "Principal Component Analysis (PCA) & NIPALS algorithm", http://folk.uio.no/henninri/pca_module/pca_nipals.pdf, 2007.

[10] Wall, E. Michael, A. Rechtsteiner, L. M. Rocha, "Singular value decomposition and principal component analysis", *in A Practical Approach to Microarray Data Analysis*, 2003, pp. 91–109.

[11] Oja, Erkki, "Simplified neuron model as a principal component analyze". *Journal of Mathematical Biology*, vol.15 (3), 1982, pp. 267–273.

[12] S. Haykin, "Neural Networks: A Comprehensive Foundation (2 ed.)", Prentice Hall, 1998.

[13] T. D. Sanger, "Optimal Unsupervised Learning in A Single-Layer Linear Feedforward Neural Network", *Neural Networks*, vol. 2, 1989, pp. 459-473.

[14] B. Scholkopf1, A. Smola, K.R. Muller, "Kernel Principal Component Analysis", http://cseweb.ucsd.edu/classes/fa01/cse291/kernelPCA_article.pdf

[15] M. Aizerman, E. Braverman, and L. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning", *Automation and Remote Control*, 1964, pp. 821–837.

[16] S. Knerr, L. Personnaz, G. Dreyfus, "Single-layer learning revisited: a stepwise procedure for building and training a neural network", *In J. Fogelman, editor, Neurocomputing: Algorithms, Architectures and Applications*, 1990, Springer-Verlag.

[17] K. Varmuza, P. Filzmoser, "Introduction to Multivariate Statistical Analysis in Chemometrics", 2009, p. 321.

[18] V. Vapnik, "Universal Learning Technology: Support Vector Machines", *NEC Journal of Advanced Technology*, vol. 2, 2005, pp. 137–144.

[19] E. Osuna, R. Freund, and F. Girosi, "An Improved Training Algorithm for Support Vector Machines", *Proceedings IEEE Neural Networks for Signal Processing VII Workshop*, 1997, pp. 276–285.

[20] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, A. Y. Wu, "An Optimal Algorithm for Approximate Nearest Neighbor Searching in Fixed Dimensions", *Journal of the ACM*, vol. 45(6), 1998, pp. 891–923.

[21] R.-E. Fan, P.-H. Chen, and C.-J. Lin, "Working set selection using second order information for training SVM", *Journal of Machine Learning Research*, 2005, http://www.csie.ntu.edu.tw/ cjlin/papers/quadworkset.pdf.

[22] J. C. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines", *Technical Report MSR-TR-98-14 Microsoft Research*, 1998, p.21.

[23] C. W. Hsu, C. C. Chang, C. J. Lin, "A practical guide to support vector classification", http://www.csie.ntu.edu.tw/c̃jlin.

[24] W. S. Sarle, "Neural Network FAQ. Periodic posting to the Usenet newsgroup comp.ai.neural-nets", 1997.

[25] http://www.face.nist.gov

ISBN 978-83-60810-27-9
ISSN 1896-7094

Multiscale Segmentation Based On Mode-Shift Clustering

Wojciech Tarnawski
Chair of Systems and
Computer Networks
Wroclaw University
of Technology, Poland
Email: wojciech.tarnawski@pwr.wroc.pl

Lukasz Miroslaw
Institute of Informatics
Wroclaw University
of Technology, Poland
Email: lukasz.miroslaw@pwr.wroc.pl

Roman Pawlikowski
and Krzysztof Ociepa
Institute of Informatics
Wroclaw University of Technology,
Poland

***Abstract*—We present a novel segmentation technique that effectively segments natural images. The method is designed for the purpose of image retrieval and follows the principle of clustering the regions visible in the image. The concept is based on the multiscale approach where the image undergoes a number of diffusions. The algorithm has been visually compared with a reference segmentation.**

I. Introduction

KNOWLEDGE-BASED society is taking advantage of communication services and IT-tools that manage, store and retrieve the information more and more often. There is a continuing demand to enhance the services that are based on visual information such as movies or images.

The problem of content-based image retrieval in large databases has been a research topic for many years. There exist semi-automatic tools to accomplish this task but still there is no standard method of general applicability. The difficulty lies in understanding what actually the image presents (image understanding) and how the contents can be described (image annotation problem). Additional limitations arise when the goal is to search for similar images in large databases. Since the analysis will require enormous demand for computer resources, the framework allowing for automated analysis will be of great use. In this paper we present a novel segmentation method based on the multiscale approach that was designed especially for image retrieval.

The concept treats the image as a set of disjoint regions that can be described by a set of features such as color, texture or shape:

$$\textbf{Image} \rightarrow \textbf{Image regions} \rightarrow \textbf{Region features} \rightarrow \\ \rightarrow \textbf{Distance between feature vectors} \quad (1)$$

With this assumption an image can be represented in multidimensional feature space as a set of points which number is consistent with the number of significant regions identified during segmentation. Regions characterized by a set of features are located in different locations in the feature space and the position depends on their visual properties. Following the Leibniz's principle called *Identity of indiscernibles* ”two things are identical if and only if they share the same and only the same properties” we assumed that similar images share objects/regions of similar properties. Which means that for a subset of images containing objects from the same category, we will observe a set of closely located points, as the regions in that subset will be similar.

Such an approach has the following consequences. Images of little complexity, for example with a few objects and an uniform background will contain only a small number of clusters separated in the feature space [1]. For more complex scenes the number of clusters will be higher and their separation will be probably difficult.

In image understanding it is difficult to find a compromise between interpretation of all image details and the interpretation where certain details could be omitted. Without *a priori* knowledge there is no automated way to determine which details can be disregarded and which objects are large enough to be treated as significant. Therefore, a scale should be considered as a parameter that changes dynamically and generates images with different level of details. With changing scale the degree of precision also changes. Generated images form a so-called multiscale representation.

The concept of multiscale representation was first introduced by Rosenfeld and Thurston [2]. They observed the influence of linear operators of different scale on edge detection. Also, Klinger [3], [4] and Tanimoto [5] used the multiscale approach to describe an image, similarly to Burt and Adelson [6] who proposed a popular, pyramidal representation of an image.

An important aspect in all these attempts is that the images at a larger scale are simplified version of images at smaller scales. Therefore, increasing the scale is equivalent with eliminating the details from images at lower scales. Following such a definition, the scale-space filtering was firstly introduced by Witkin [7] and then further developed by Koenderink [8].

Our concept to image retrieval differs from methods named *Query By Image Content* that tries to extract the information from the whole image [9]. In contrary, the method considers the image as a set of regions represented by a cluster of points in the feature space where the similarity between images is equivalent to a degree of similarity between feature vectors. Therefore, correct segmentation of the regions is a prerequisite

978-83-60810-27-9/09/$25.00 © 2010 IEEE

in image retrieval and machine vision tasks where objects play the necessary role. Determination of the criterion for object homogeneity is equally important. Mostly features based on color are used but also other features are often employed, i.e. low-level features such as SIFT, visual descriptors in MPEG-7 standard [10]–[12].

Multiscale approach to segmentation has been already proposed. Wang used a multiscale approach based on high frequency wavelet coefficients and their statistics to perform context-dependent classification of individual blocks of the image. Unlike other edge-based approaches, his algorithm does not rely on the process of connecting object boundaries [13], [14].

The next section describes the segmentation method. The method takes into account the color as the most discriminating feature. Since it is based on concepts such as mean-shift clustering and multiscale approach based on anisotropic diffusion, also these concepts are presented. The last section presents the results and conclusions.

II. Segmentation method

The aim of segmentation is to partition the image into non-overlapping regions that share common features. In case of images of human nature the significant information are derived by color, therefore the features of interest are taken from color model and position. We have decided to take the multiscale approach because such a concept is natural for human perception. When we see the picture, first, we focus on the core objects and analyze regions of strong contrast and different color, next we analyze their details, such as texture. By running a number of diffusions on the image, such concept can be imitated, as with the number of diffusions the details in the image get blurred and only the objects with highest contrast remain. The information on the image is, therefore, simplified.

The segmentation method is depicted in Fig. 1 and can be described as follows. Original image (1) undergoes the multiscale operation and a set of images with different degree of diffusion is generated (2). Next, mean-shift segmentation is produced for each of the image. The results are accumulated in special storage system called accumulator (4). The mode-shift clustering together with a certain metric (6) and a threshold value assigned to it (7) is used in order to label disjoint region of interest and partition them into two layers. The principle layer stores labeled objects that are clearly visible at all the scales and the vague layer contain the regions less distinguishable (8).

A. Anistropic diffusion in multiscale approach

This is the initial step of the algorithm. Here, a number of images are generated in the process of convolving the original image I_0 with the Gaussian kernel with the t variance:

$$I(x, y, t) = I_0(x, y) * G(x, y; t) \tag{2}$$

The variance controls the degree of details visible in the image. Higher values correspond to the image with fewer details that can be clearly distinguishable. The set of derivative images $I(x, y, t)$ is equivalent to concurrent solutions of the heat transport problem or diffusion on the plane [8], namely:

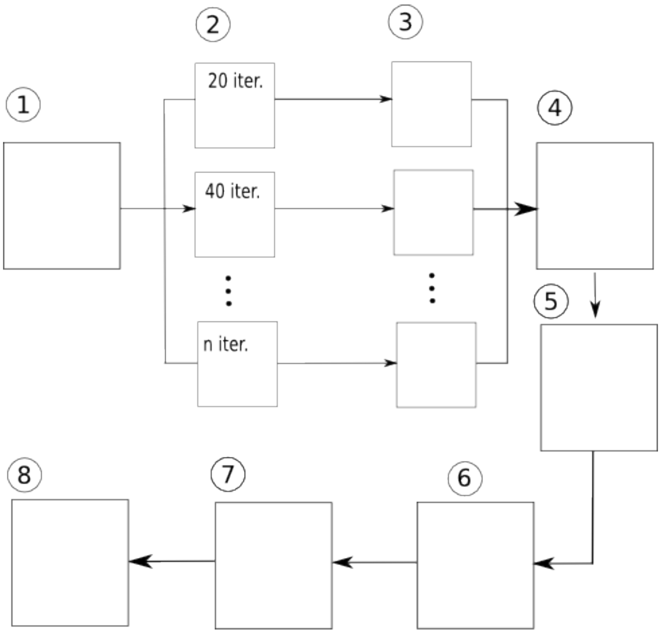


Fig. 1. Segmentation Algorithm. 1. Input image. 2. Multiscale approach, anisotropic diffusion. 3. Meanshift Segmentation. 4.Accumulator. 5. ”Modeshift” clustering. 6. Calculation of the metric. 7. Adaptive thresholding. 8. Output image.

$$I_t = \nabla^2 I = I_{xx} + I_{yy} \tag{3}$$

with initial conditions defined as $I(x, y, 0) = I_0(x, y)$.

Since, the convolution operation smoothes the whole image together with boundaries between objects, we decided to use edge-preserving anisotropic diffusion. The importance of this approach lies in the fact that the diffusion coefficient is not the same for all the pixels. The method is define as follows [15]:

$$I_t = div(c(x, y, t) \cdot \nabla I) = c(x, y, t)\nabla^2 I + \nabla c \cdot I \tag{4}$$

where $c(x, y, t) = g(||\nabla I(x, y, t)||)$ is monotonically decreasing so that within homogeneous regions the diffusion is stronger than in the vicinity of region edges.

In the case of approximation of ∇I with 4-directional neighbourhood we can describe this process as

$$\begin{aligned} I^{(t+1)}_{x,y} = I^{(t)}_{x,y} + \lambda[D_N \cdot \Delta_N I + D_S \cdot \Delta_S I + \\ D_E \cdot \Delta_E I + D_W \cdot \Delta_W I]^{(t)}_{x,y} \end{aligned} \tag{5}$$

where $\lambda = 1/4$, symbols N, S, E, W correspond to directions North, South, West, East, respectively, and Δ is the difference between pixel values in the directions for each iteration t:

$$\begin{aligned} \Delta_N I \equiv I_{x,y-1} - I_{x,y},\ \Delta_S I \equiv I_{x,y+1} - I_{x,y}, \\ \Delta_W I \equiv I_{x-1,y} - I_{x,y},\ \Delta_E I \equiv I_{x+1,y} - I_{x,y}, \end{aligned} \tag{6}$$

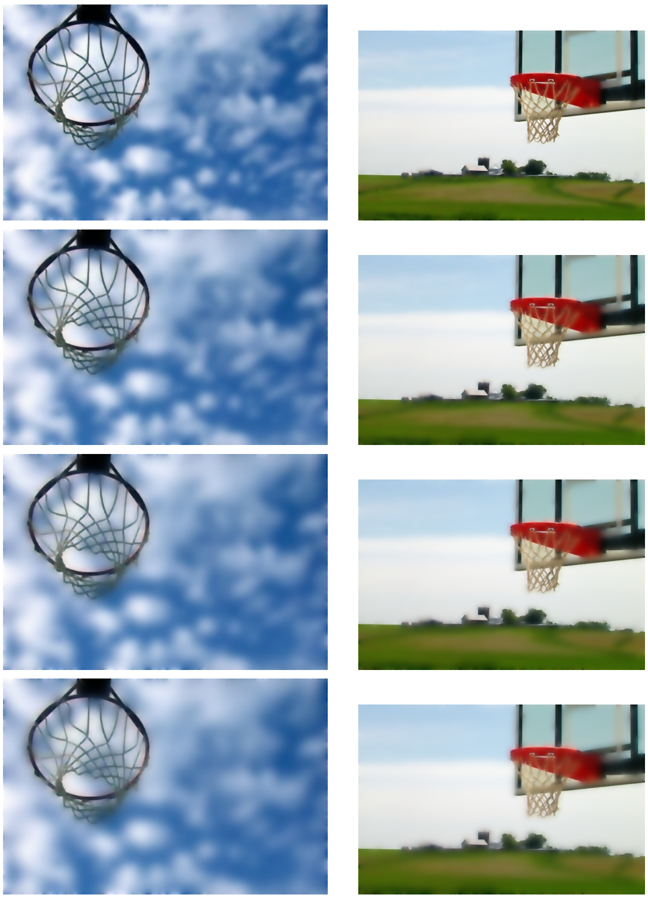


Fig. 2. Filtration results with anisotropic diffusion for $t = 50, 100, 150, 200$ iterations and . $K = 25$.

Similarly, diffusion coefficients are modified for each iteration t as follows:

$$\begin{aligned} D_N &= g\left(||(\nabla I)_{x,y-\frac{1}{2}}||\right)\ D_S \\ &= g\left(||(\nabla I)_{x,y+\frac{1}{2}}||\right) \\ D_W &= g\left(||(\nabla I)_{x-\frac{1}{2},y}||\right)\ D_N \\ &= g\left(||(\nabla I)_{x+\frac{1}{2},y}||\right) \end{aligned} \quad (7)$$

Gradient values $||(\nabla I)||$ were calculated by the Canny filtering [16] that guarantees that the edges are one pixel thick. Similarly to [17] the $g(\cdot)$ was defined as :

$$g\left(||(\nabla I)||\right) = 1 - \exp(-\frac{\tau}{[\Psi(||\nabla I||)/K]^m}). \quad (8)$$

where $m = 4$, $\tau = 3.31488$ a K is a diffusion parameter.

Fig. II-A describes the results of anisotropic diffusion for $K = 25$ and $t = 50, 100, 150, 200$ iterations.

B. Mean Shift Segmentation

During this step, each of the image at different scale goes through so called meanshift segmentation. In contrary to low-level segmentation methods that depend on parameters or apriori knowledge about image contents, clustering methods have ability to be independent on these limitations. The ”mean-shift” algorithm is a non-parametric clustering method widely used for segmentation of images of human nature. The method does not require to know the number and the shape of clusters. For each pixel of an image, the set of adjacent pixels is established according to a certain kernel function. For these adjacent pixels the new spatial and color mean values are calculated and used the new center for the next iteration. These steps are repeated until the means do not change. At the end of the iteration, the final mean color will be stored.

The method can be described as follows. Let us define an image as a set of n points $x_i, i = 1, \cdot, n$ in d-dimensional space R^d, and estimator for the density kernel $K(x)$ with radius h defined as

$$f(x) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \quad (9)$$

For radially symmetric kernels it is sufficient that $K(x)$ follows

$$K(x) = c_{k,d}k(||x||^2) \quad (10)$$

where $c_{k,d}$ is normalization constant for which $\int_x K(x) = 1$. Modal values of density function are located in intercepts of the $\nabla f(x) = 0$. Density gradient is defined as follows:

$$\nabla f(x) = \frac{2c_{k,d}}{nh^{d+2}}\sum_{i=1}^{n}(x_i - x)g\left(||\frac{x - x_i}{h}||^2\right) =$$
$$\frac{2c_{k,d}}{nh^{d+2}}\left[\sum_{i=1}^{n} g\left(||\frac{x - x_i}{h}||^2\right)\right]\left[\frac{\sum_{i=1}^{n} x_i g\left(||\frac{x-x_i}{h}||^2\right)}{\sum_{i=1}^{n} g\left(||\frac{x-x_i}{h}||^2\right)} - x\right]. \quad (11)$$

where $g(s) = -k'(s)$. The first term is propotional to the estimator of the density kernel of x vector calculated with the the kernel function $G(x) = c_{g,d}g(||x||^2)$ and the second term is defined as

$$\frac{\sum_{i=1}^{n} x_i g\left(||\frac{x-x_i}{h}||^2\right)}{\sum_{i=1}^{n} g\left(||\frac{x-x_i}{h}||^2\right)} - x \quad (12)$$

as is called a ”mean shift” vector. This vector is responsible for traversals between the given point to the prototype of a given cluster (attractor). Direction of the vector is oriented to the highest ascend of density. The range of the kernel h is described by two components: a planar one h_s and the feature-driven one h_r that describes the range of features. This procedure is an iterative process made of two steps:

- calculation of the mean-shift vector

$$m_h(x^{(t)}) \quad (13)$$

- translation within the window with the vector

$$x^{(t+1)} = x^{(t)} + m_h(x^{(t)}) \quad (14)$$

and converges to the point where the density gradient is equal to zero. The more detailed description of the method can be found in [18].

C. Definition of Accumulator

Here all the segmentation results at various scale t_q are accumulated and the final segmentation is derived. This procedure has the following steps:

Step 1 Define L-dimensional matrix A which we will call *accumulator*, where L corresponds to the dimension of the feature space that describe the regions. A will store regions defined by a prototype $v_i(t_q)$ composed of features $v_i^{(\cdot)}(t_q)$. In the current stage the features will be defined in the following LAB color model. Obviously, each feature space will determine different prototype distributions in the feature set $v_i(t_q)$. Let us assume, that a single prototype will be described by a $L+2$–dimensional vector:

$$v_i(t_q) = [v_i^{(x)}(t_q), v_i^{(y)}(t_q), v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)]^T \tag{15}$$

where T is a symbol of transposition. In the accumulator space the features $v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)$, will be added without planar features of the prototypes. Matrix A is initialized with zeros.

Step 2 Scan the pixels of $I(x,y,t_q)$ image for each scales t_q and define the feature vector for corresponding prototypes $[v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)]$. Store the mapping:

$$\begin{aligned} I(x,y,t_q) &\to [v_i^{(x)}(t_q), v_i^{(y)}(t_q), v_i^{(1)}(t_q), \ldots, \\ &\ldots v_i^{(L)}(t_q)] \to [v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)] \end{aligned} \tag{16}$$

Step 3 For each pixel in each scale increment the accumulator:

$$\begin{aligned} &A[v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)] = \\ &\qquad A[v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)] + 1 \end{aligned} \tag{17}$$

D. Mode Shift clustering

This procedure aims at finding significant *modal values* in the accumulator space (4-dimensions: three color values and one corresponding to the accumulator range) where each region is represented by a single point. Mode shift clustering groups points in the vicinity of modal values. The resulting clusters are restrained by spatial and range limit values: σ_S and σ_R. The first parameter defines the searching window of the cubic shape, the latter one compares the accumulator range with local extrema. During this searching procedure, each point in the accumulator moves along the direction of the closest modal value with some finite number of steps.

The points that are grouped form a cluster and define the next mapping:

$$\begin{aligned} &[v_i^{(1)}(t_q), \ldots, v_i^{(L)}(t_q)] \to \\ &\qquad [V_j^{(1)}, \ldots, V_j^{(L)}];\ j = 1, 2, \ldots, C \end{aligned} \tag{18}$$

where C is the number of detected modal values. Note, that the mapping (18) does not have to be defined for all prototypes from the set $v_i(t_q)$.

E. Calculation of the metric

The clusters formed in the previous step are restrained according to a certain metric that defines the *type of distance*. Different metrics (for the sake of simplicity we named them Epsilon) were considered to calculate the distance and number of steps between a given point in the accumulator and the modal value (the cluster prototype) established during ”mode-shift” clustering. They are defined as follows:

- Epsilon type 0—Sum of consecutive steps between a given point and the modal value
- Epsilon type 1—Manhattan distance
- Epsilon type 2—Euclidean distance
- Epsilon type 3—Sum of consecutive steps in a given time
- Epsilon type 4—Euclidean distance weighted by the number of steps needed to achieve a cluster prototype
- Epsilon type 5—Sum of consecutive steps in a given time weighted by the number of steps needed to achieve a cluster prototype

F. Adaptive threshold

With each metric a certain threshold value is associated. This value controls whether the point belongs to the *principle layer* or to the *vague layer*. The regions of higher contrast that survived diffusions and have higher values in the accumulator constitute the principle layer. Objects with lower contrast have smaller values in accumulator and, therefore, belong to the vague layer.

During this step the labeling of the pixels in the image with the mappings: (18) and (16) is performed according to the mentioned threshold. Pixels with undefined mapping (16) are labeled as -1.

The output image contains disjoint regions and the algorithm is terminated. The image will contain two subsets with pixels:

- with a label equal to -1, which form the vague layer.
- with a label different than -1, which will create the principle layer.

III. Results

The algorithm has been tested for different parameter settings on Segmented and Annotated Benchmark set [19]. Especially metric Epsilon has been thoroughly tested and the segmentation results were compared visually with a reference segmentation done by the expert. The analysis of results indicated that the best results were achieved for Epsilon type 4 and 5. Example images 3 were analyzed with two different threshold values and presented in consecutive rows in Figs. 4–5. As one can see, the results are comparable to each other, alas, more work is needed to perform a detailed analysis of parameter influence.

Nevertheless, it may be easily noticed that the segmented regions correspond to the reference segmentation with a satisfactory accuracy, i.e. the significant objects on principle layer are detected (the right column) and their periphery are close to the boundaries defined by the expert. Such region

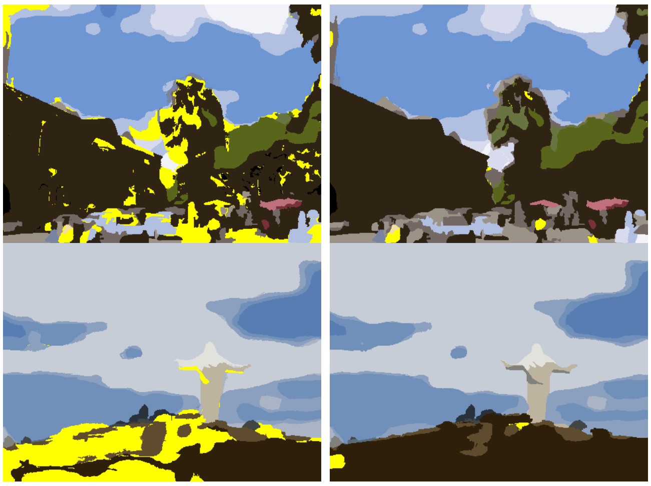


Fig. 5. Example results for epsilon type 5 and different threshold values. The vague layer is indicated by yellow pixels.

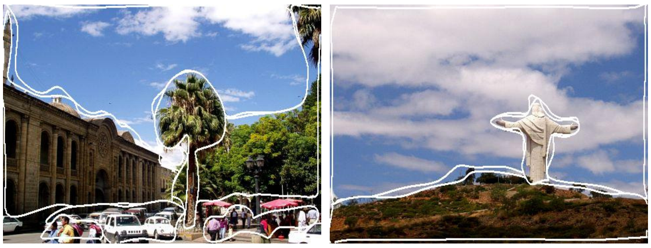


Fig. 3. Original images.

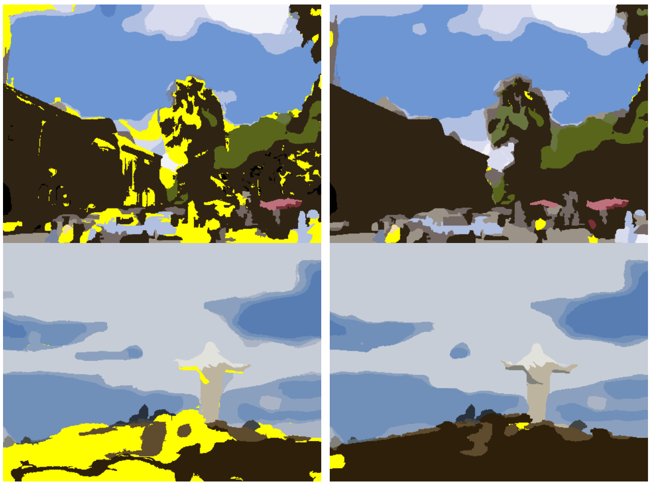


Fig. 4. Example results for epsilon type 4 and different threshold values. The vague layer is defined by yellow pixels.

representation as a feature set are a requirement for image retrieval based on clustering principle.

IV. Conclusions

The paper presents the segmentation method based on multiscale approach and mean-shift clustering. A novel algorithm partitions the image into disjoint sections that form the layers with significant (principal layer) and insubstantial regions (vague layer). Detected regions The results were assessed visually for a number of image classes method. Although, the results were satisfactory for a number of image classes and the method can serve as a potential tool in image retrieval task, more work is needed to fully customise the method.

Acknowledgment

The financial support of the Polish-Singapur Research Fund (project No. 65/N- SINGAPORE/2007/0) has made this research possible and is kindly acknowledged.

References

[1] W. Pedrycz, A. Amato, V. Di Lecce, and V. Piuri, “Fuzzy clustering with partial supervision in organization and classification of digital images,” *Fuzzy Systems, IEEE Transactions on*, vol. 16, no. 4, pp. 1008–1026, Aug. 2008.

[2] A. Rosenfeld and M. Thurston, “Edge and curve detection for visual scene analysis,” *IEEE Trans. Comput.*, vol. 20, no. 5, pp. 562–569, 1971.

[3] M. Klinger, “Pattern and search statistic,” *Optimizing Methods in Statistics*, 1971.

[4] L. Uhr, “Layered ”recognition cone” networks that preprocess, classify, and describe,” *Computers, IEEE Transactions on*, vol. C-21, no. 7, pp. 758–768, July 1972.

[5] S. Tanimoto and T. Pavlidis, “A hierarchical data structure for picture processing,” *Computer Graphics and Image Processing*, vol. 4, no. 2, pp. 104 – 119, 1975.

[6] P. J. Burt and E. H. Adelson, “The laplacian pyramid as a compact image code,” *IEEE Transactions on Communications*, vol. 31,4, pp. 532–540, 1983.

[7] A. Witkin, “Scale-space filtering: A new approach to multi-scale description,” in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP ’84.*, vol. 9, Mar 1984, pp. 150–153.

[8] J. J. Koenderink, “The structure of images,” *Biological Cybernetics*, vol. 50, pp. 363–370, 1984.

[9] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, “Query by image and video content: the qbic system,” *Computer*, vol. 28, no. 9, pp. 23–32, Sep 1995.

[10] M. Bober, “Mpeg-7 visual shape descriptors,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 11, no. 6, pp. 716–719, Jun 2001.

[11] M. Bober and P. Brasnett, “Mpeg-7 visual signature tools,” in *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, 28 2009-July 3 2009, pp. 1540–1543.

[12] M. Bober, W. Price, and J. Atkinson, “The contour shape descriptor for mpeg-7 and its applications,” in *International Conference on Consumer Electronics, Digest of Technical Papers.*, 2000, pp. 286–287.

[13] Z. Chi and H. Yan, “Image segmentation using fuzzy rules derived from k-means clusters,” *Journal of Electronic Imaging*, vol. 4, no. 2, pp. 199–206, 1995.

[14] J. Z. Wang, J. Li, R. M. Gray, and G. Wiederhold, “Unsupervised multiresolution segmentation for images with low depth of field,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, pp. 85–90, 1999.

[15] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, 1990.

[16] F. J. Canny, “A Computational Approach to Edge Detection,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[17] J. Weickert, *Anisotropic Diffusion in Image Processing*, ser. ECMI Series. Stuttgart, Germany: Teubner-Verlag, 1996.

[18] D. Comaniciu and P. Meer, “Mean shift: A robust approach toward feature space analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, 2002.

[19] H. J. Escalante, C. Hernández, J. Gonzalez, A. López, M. Montes, E. Morales, E. L. Sucar, and M. Grubinger, “The segmented and annotated IAPR TC-12 benchmark. computer vision and image understanding,” *Computer Vision and Image Understanding*, vol. 114, no. 4, pp. 419–428, 2009.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 135–139

ISBN 978-83-60810-27-9
ISSN 1896-7094

Relational database as a source of ontology creation

Zdenka Telnarova
University of Ostrava 30. dubna
22, 701 03 Ostrava, Czech
Republic
Email:zdenka.telnarova@osu.cz

Abstract—**The article headed “Relational database as a source of an ontology creation” deals with mapping relational data into ontology, or filling ontology with data from relational databases. It describes the issue of mapping database schemas (particularly relational models) for common data models expressed in the form of ontology. Generous room is given to methods of acquiring ontology from relational databases, where rules are specified and simple example are used to demonstrate their use, mapping of individual concepts of a relational data model into ontology concepts.**

I. Methods of Creating Ontology from Database

AN ontology provides shared and repeatedly usable knowledge about a specific domain. Existing relational databases, which contain enormous amounts of data, can be used to fill ontology with knowledge. There are numerous methods how to use existing databases to fill ontology using rules or using the so-called middle model. Many works deal with the issue of adding semantic to database schemas, e.g. [2]. Also, object-oriented data models are closely linked to ontological theory, nevertheless, even these do not offer sufficient semantic properties, such as hierarchies, for example. Methods of using relational databases for acquiring ontology can be divided minimally into the following four areas.

Area 1: Creation of a new database schema based on an analysis of the current relational schemas and mapping of this schema onto an ontology using reverse engineering. The new data model is designated as a so-called middle model, which is a common model above partial relational models [3].

Area 2: A method based on filling ontology instances using external relational sources. It uses a declarative interface between the ontology and data sources in XML.

Area 3: A method which creates ontology from a conceptual database schema, in the process the aim of this method is to create a RDF(S) ontology. However, this method has an unclear semantic and does not offer an inference model which is necessary for automatic derivation.

Area 4: A method which creates ontology from a conceptual database schema, whereas the aim of this method is to create an OWL ontology.

Further, we shall deal with only the last of the above said methods.

II. Ontology over the Relational Schema

The following definitions are necessary for describing the method of mapping relational data to an OWL ontology [9].

Definition I. (ontology with relation)

The ontological structure consists of five concepts
$O = \{C, R, H^C, rel, A^0\}$, where
C is a finite set of concepts
R is a finite set of relations

H^C is a concept hierarchy or taxonomy, $H^C \subseteq C \times C$, e.g. $H^C (C_1, C_2)$ expresses that C_1 is a sub-concept of C_2

rel is a set of non-taxonomic relations, e.g. $rel(R)=(C_1, C_2)$ expresses that concepts C_1, C_2 enter into relation R.

A^0 is a set of axioms expressed in the respective logical language, e.g. first-order logic.

The ontological structure includes as its extension a set of instances which suit this structure. Most OWL ontology elements involve classes, properties and instances. Most concepts of a specified domain correspond with classes which are the roots of various taxonomic trees. Properties are used to express general facts about class members and specific facts about instances. A property is a binary relation with a specified domain and scope. OWL distinguishes two types of properties. There are property between an instance and data type and property between instances of two classes. Both properties and classes may be classified in a hierarchy.

Definition II. (relational model)

In a relational model the R_i relation corresponds to a table where columns are called attributes and are designated A. The $attr(R_i)$ function returns attributes connected with a specific R_i relation. The dom (A_i) function determines the value range of attribute A_i, $A_i \in A$. The $pkey(R_i)$ function returns the primary key of the given R_i relation, whereas it must apply that $pkey(R_i) \subseteq attr(R_i)$. The $fkey(R_i)$ function returns a foreign key of the given relation R_i, again, it must apply that $fkey(R_i) \subseteq attr(R_i)$. Relations in a relational schema may be interdependent inclusively or equivalently.

Definition III. (inclusive dependence of relations)

Let R_i and R_j be relations and it is presumed that $A_i \subseteq attr(R_i)$ and $A_j \subseteq attr(R_j)$. Let $t_i(A_i)$ be value of t_i with attributes A_i and $t_j(A_j)$ value of t_j with attributes A_j. If for each $t_i(A_i)$ in R_i exists a $t_j(A_j)$ in R_j, for which it applies that $t_i(A_i) = t_j(A_j)$, then A_i and A_j are inclusively dependent, $R_i(A_i) \subseteq R_j(A_j)$ is recorded.

Definition IV. (equivalent dependence of relations)

Let R_i and R_j be relations and it is presumed that $A_i \subseteq attr(R_i)$ and $A_j \subseteq attr(R_j)$. If exists $R_i(A_i) \subseteq R_j(A_j)$ and $R_j(A_j)$

$\subseteq R_i(A_i)$, then A_i and A_j are equivalently dependent, $R_i(A_i) = R_j(A_j)$ is recorded.

III. Acquiring Ontology from a Relational Database

The task of acquiring ontology from a relational database can be divided according to which constructs, characteristics respectively, are acquired. Further, only the following constructs and characteristics of the created ontology shall be considered: classes, properties, hierarchy, cardinality, instances.

The principle of building ontology from a relational database, as is described by the set of rules in the following chapters, presents the reverse procedure to the procedure of transforming the conceptual model to a relational model. This transformation is based on the definition of a relational data model and transforms constructs designed in the conceptual model to constructs which permit a relational data model. This involves primarily decomposition of relations in the N:M cardinality and a conceptual model using multi-value and group attributes. Further, the issue of compulsory or optional participation of entities in relation to a conceptual model entity is discussed. The building of ontology is then based on the revealing of the original conceptualization, which the transformation process rendered unnoticed and thus the original semantic of modelled reality was lost (hidden). We propose following set of rules that allows inverse process to the process of transforming conceptual model into relational model. These rules transform relational model into conceptual model or ontology.

A. Rules for creating classes

The creation of ontology concepts as "classes" is based on two rules.

Rule I:

If we have database relations R_1, R_2, …, R_i, where P_1= pkey(R_1), P_2=pkey(R_2), …P_i=pkey(R_i), for which $R_1(P_1) = R_2(P_2) = … = R_i(P_i)$ apply, then information from these database relations can be integrated into one ontological class C_i. These relations form one entity on a conceptual abstract level.

Rule II:

Ontological class C_i can be created from relation R_i, if no other relation exists which could be integrated with relation R_i according to Rule I. and, at the same time, one of the following conditions applies:

|pkey(R_i)| = 1

|pkey(R_i)| > 1 and, at the same time, it applies that A_i exists, where $A_i \in$ pkey(R_i) and also $A_i \notin$ fkey(R_i).

This rule expresses that if relation R_i is used to describe an identification independent entity, then it may be mapped into one ontological class. The following example shows a practical case of mapping relations into an ontological class.

Example I.

A relational database schema consists of relations, their attributes, primary and foreign keys, as is shown in the following table.

Table I.
Relational Database Schema

Relation	Primary key	Foreign key
Student(Pers_number, Name,...., Town_residence)	Pers_number	Town_residence refers to relation Town Id_town
PhDStudent(Pers_numb er, …, Trainer)	Pers_number	Pers_number refers to relation Student Pers_number
Town(Id_town, Name_town, …..)	Id_town	-
Department(Id_departme n, Name_department, …)	Id_department	-
Studies(Pers_number, Id_department, ….)	Pers_number, Id_department	Pers_number refers to relation Student Pers_number
Employee(Id_employee , ….)	Id_employee	Id_department refers to relation Department Id_department

Example II.

According to Rule I. it is possible to create ontological class `Student`, which shall integrate relations `Student` and `PhDStudent` because `Student(Pers_number) = PhD Student(Pers_number)`. In OWL this class can be created as

```
<owl:Class rdf:ID = "Student" />.
```

According to Rule II. it is possible to create ontological classes `Town`, `Department` and `Employee` (condition 1 of this rule is fulfilled, |pkey(R_i)| = 1, i.e. number of attributes of primary key is one. In OWL classes can be created:

```
<owl:Class rdf:ID = "Town" />
<owl:Class rdf:ID = "Department" />
<owl:Class rdf:ID = "Employee" />.
```

B. Rules for creating properties

In OWL there are two types of properties – object properties and data type properties. The following rule must be defined for capturing properties arising from the relation schema into ontology.

Rule III:

If R_i and R_j are relations, where $R_i(A_i) \subseteq R_j(A_j)$ and, at the same time, $A_i \not\subset$ pkey(R_i), object property of ontology P can be created, based on attributes A_i of relation R_i. Assuming that relation R_i was mapped into class C_i and relation R_j was mapped into class C_j the domain of P is C_i and range of P is C_j.

Rule IV:

If R_i and R_j are two relations, then it is possible to create two object properties “has_part” and “is_part_of”, if the following two conditions are fulfilled:

$|pkey(R_i)| > 1$

$fkey(R_i) \subset pkey(R_i)$, where $fkey(R_i)$ refers to R_j

Provided that R_i is mapped on C_i and R_j on C_j, the domain, or range respectively, of the “is_part_of” C_i, C_j respectively, and the domain, or range respectively, of the “has_part” is C_j, C_i respectively.

Rule V:

Let R_i, R_j and R_k be relations and let it apply: A_i=pkey(R_i), A_j= pkey(R_j), $A_i \cup A_j$ = fkey(R_k) and $A_i \cap A_j = \emptyset$, the two ontology object properties can be created P_j‘ and P_j‘‘ based on R_k. Provided that R_i is mapped on C_i and R_j on C_j, domain and range of P_j‘ is C_i and C_j and domain and range of P_j‘‘ is C_j and C_i. Pj‘ and P_j‘‘ are two inverse object ontology properties.

Example III.

According to Rule III. `Town_residence` can be created as an ontology object property. The domain of this concept is `Student`, range is `Town`. In OWL the creating of a concept can be written as:

```
 <owl:Object Property
rdf:ID=‶Town_residence‶ />
 <rdfs:domain rdfs:source=‶#Student ‶ />
 <rdfs:range rdfs:source=‶#Town ‶ />
 <owl:ObjectProperty />.
```

Example IV.

According to Rule V., two object properties can be created based on the `Studies` relational relation. We’ll call these properties `pertains_to` and `has_student`, which shall be associated with classes of ontology `Student` and `Department`. The corresponding record in OWL is as follows:

```
 <owl:Object Property rdf:ID = ‶
pertains_to ‶ />
 <rdfs:domain rdfs:source = ‶#Student
‶ />
 <rdfs:range rdfs:source = ‶#Department
‶ />
 <owl:ObjectProperty />
 <owl:Object Property rdf:ID =
‶has_student ‶ />
 <rdfs:domain rdfs:resource =
‶#Department ‶ />
 <rdfs:range rdfs:resource =
‶#Student‶ />
 <owl:inverseOf rdf:resource=‶
pertains_to ‶ />
 <owl:ObjectProperty />.
```

RuleVI:

Rule VI. defines the method of creating data type properties and specifies that each attribute of all relations of the relational schema, which cannot be transferred to an object property, can be transferred to a data type property.

For ontology class C_i and a set of data type designated as DP(C_i), if C_i correspond to database relations R_1, R_2, …R_i, then for each attribute in R_1, R_2, …R_i, for which an object property cannot be created subject to Rule III., a data type property can be created. The domain of each property P_i is C_i and the range of each property is dom(A_i) for each $P_i \in$ DP(C_i) and $A_i \in$ attr(R_i).

Example V.

According to Rule VI. it is possible to create data type properties corresponding to ontology classes from attributes of relational schemas `Pers_number`, `Name`, `Trainer`, `Id_town`, `Name_town`, `Id_department`, `Name_department`, `Id_employee`. The segment in OWL could be as follows:

```
 <owl:Datatype Property rdf:ID =
‶Pers_number‶ />
 <rdfs:domain >
   <owl:Class >
 <owl:unionOf rdf:parseType =
‶Collection ‶ />
 <owl:Class rdf:about = ‶ #Student ‶ />
 <owl:Class rdf:about = ‶ #PhDStudent
‶ />
 <owl:Class rdf:about = ‶ # Pers_number
‶ />
 </owl:unionOf >
 </owl:Class >
 </owl:domain >
 <rdfs:range rdfs:source = ‶&xsd;int
‶ />
 <owl:DatatypeProperty />
 …
```

C. Rules for creating hierarchies

As is well known, the conceptual model uses the so-called Isa hierarchy, which enables the application of heritability in the object model. This Isa hierarchy does not permit the relational data model and therefore this Isa hierarchy must be decomposed during transformation of the conceptual model to a relational data model. Usually the identification dependence of relations is used, i.e. the identification key of the super-type becomes the identification key of the sub-type and also its foreign key. Conversely it can be deduced that if a relation has a foreign key as a part of its primary key, then it is a relation which was created by an Isa hierarchy transformation.

RuleVII:

If R_i and R_j are relations and it applies that P_i=pkey(R_i) and P_j=pkey(R_j), then if Rule I. cannot be applied and $R_i(P_i) \subseteq Rj(P_j)$ is fulfilled, then class/property of the corresponding R_i is a sub-class/sub-property of the corresponding R_j.

Example VI.

According to Rule VII., ontological class `PhDStudent` is a sub-class of ontological class `Student` and can be recorded in OWL as follows:

```
  <owl:Class rdf:ID = ‟PhDStudent‟ />
  <rdfs:subClassOf     rdf:resource     =
‟#Student ‟ />
  </owl:Class >
```

D. Rule for creating cardinalities

Attributes of relations according to the domain integral limitation designed above individual attributes of relations may acquire no, only one, at least one, maximum of one or more values. Based on a specified domain integral limitation in the relational schema the property cardinalities can be defined in the ontology.

Rule VIII:

If R_i is a relation with attributes $A_i \in$ attr(R_i), where A_i = pkey(R_i), then the minimum and, at the same time, maximum cardinality corresponding to property P_i is equal to 1.

Rule IX:

If R_i is a relation with attributes $A_i \in$ attr(R_i), where A_i is declared as NOT NULL, then the minimum cardinality corresponding to property P_i is equal to 1.

Rule X:

If R_i is a relation with attributes $A_i \in$ attr(R_i), where A_i is declared as UNIQUE, then the maximum cardinality corresponding to property P_i is equal to 1.

Example VII.

According to Rule VIII. the minimum and, at the same time, maximum cardinality of attributes `Pers_number`, `Id_town`, `Id_department` is equal to 1. In OWL it is recorded as:

```
  <owl:Restriction />
  <owl:onProperty rdf:resource =
‟#Pers_number ‟ />
  <owl:minCardinality>1</owl:minCardinali
ty>
  <owl:maxCardinality>1</owl:maxCardinali
ty>
  </owl:Restriction >
  <owl:Restriction />
  <owl:onProperty     rdf:resource     =
‟#Id_town ‟ />
  <owl:minCardinality>1</owl:minCardinali
ty>
  <owl:maxCardinality>1</owl:maxCardinali
ty>
  </owl:Restriction >
  <owl:Restriction />
  <owl:onProperty  rdf:resource  =  ‟#Id_
‟ />
  <owl:minCardinality>1</owl:minCardinali
ty>
  <owl:maxCardinality>1</owl:maxCardinali
ty>
  </owl:Restriction >
```

E. Rule for creating instance

The following rule can be used for automatic creation of ontology instances from data stored in relational databases.

Rule XI:

If C_i is an ontological class of the corresponding to database relations R_1, R_2, …R_i, then each n-tuple of t_i, for which $t_i \in R_1$ x R_2 x … x R_i applies, can become an instance C_i.

Note: x designates the Cartesian sum of relations.

Fig 1. Graph shows the fragment of the ontology created by describing examples using Altova SemanticWorks software.

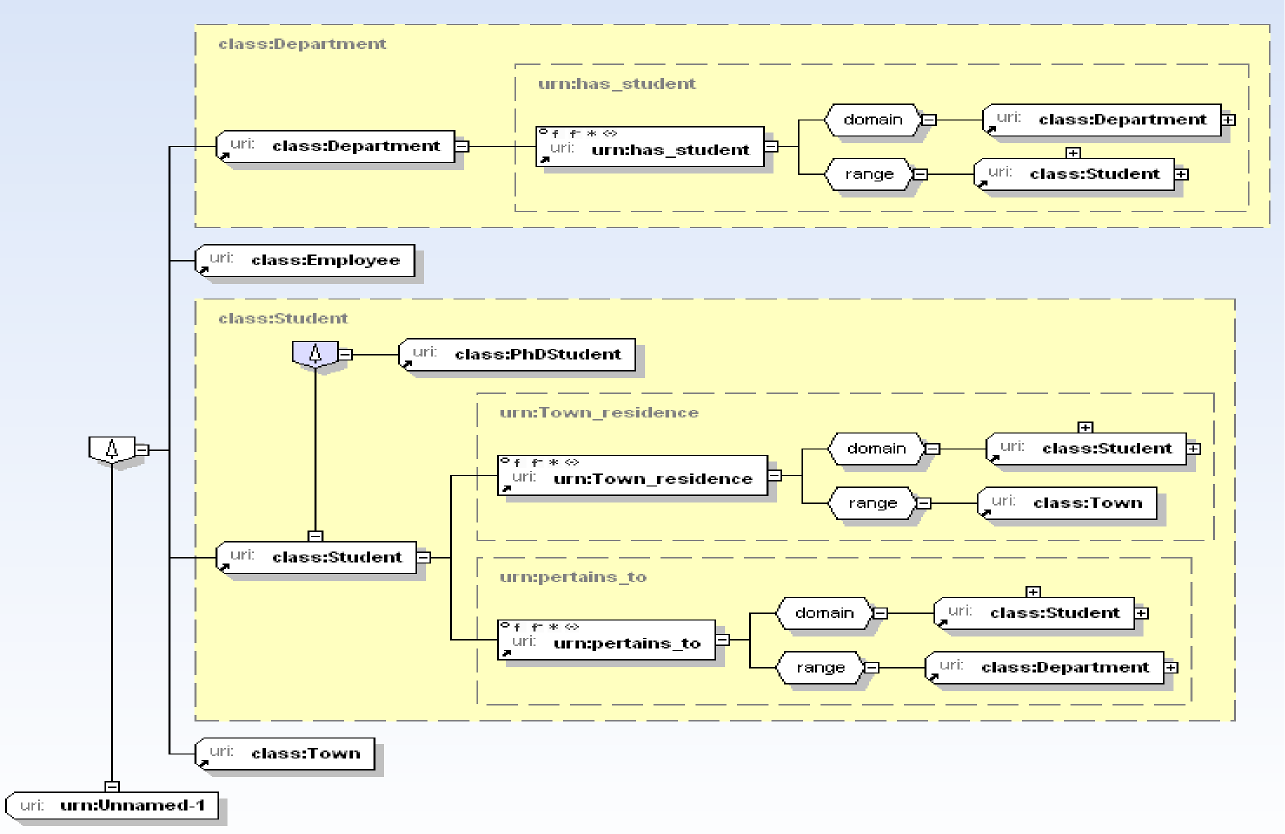


Fig. 1 Graph

IV. Conclusion

Mapping relational data into ontology, or filling ontology with data from relational databases respectively, plays an important role during the creation and updating of ontology. If we consider the quantity of data found in relational databases and the potential of their joint use across various applications, the question of finding a method for their use becomes obvious. The heterogeneity of data complicates use and therefore methods of their integration are being sought. The method based on a common semantic model – ontology is one way of efficient integration. The question is how to correctly map relational database data into ontology instances. The quantity of manual work during creation of ontology poses a serious problem; therefore, automation based on mapping subject according to the rules is a god solution. This article focuses on the principles of automatic conversion of constructs of the relational data model to constructs of OWL ontology and transfer of relational data to ontology instances. Mainly, rules are defined which map the basic concepts of the relational data model onto ontology. Individual rules are complemented by specific examples. Examples utilize a common simple data model.

References

[1] Miller, R., Haas L. M., Hernandez M. A.: Schema Mapping as Query Discovery. In VLDB'00, pages 77–88, 2000.

[2] Biskup, J.: Achivements of relational database schema design Tudory revisited. Semantics in Database. LCNS 1358, Springer Verlag, 1998

[3] Kashyap, V.: Design and creation of ontologie for environmental information retrieval. Proc. Of the 12 th Workshop on Knowledge Acquisition, Modeling and Management. Alberta, Canada. 1999.

[4] Sicilia, M.A., Lytras, M.D.: Metadata and Semantics. Springer, 2009.

[5] Rishe N. Database design: the semantic modeling approach. McGraw-Hill, 1992.

[6] Biskup, J.: Achievements of relational database schema design theory revisited. Semantics in Database, LCNS 1358, Springer Verlag, 1998.

[7] Chiang R, Barron T, Storey V. Reverse engineering of relational databases: extraction of an EER model from a relational database. Journal of data and knowledge engineering, 1994,12(2):107-142.

[8] Vermeer M, Apers P. Object-oriented views of relational databases incorporating behaviour. DASFAA, 1995.

[9] Stojanovic, L., Stojanovic, N., Volz, R.: Migrating data-intensive Web Sites into Semantic Web. ACM press, 2002.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 141–147

ISBN 978-83-60810-27-9
ISSN 1896-7094

Emotional Speech Analysis using Artificial Neural Networks

Jana Tuckova
Czech Technical University in Prague
Faculty of Electrical Engineering
Technicka 2, 166 27 Prague 6, Czech Republic
E-mail: tuckova@fel.cvut.cz

Martin Sramka
Czech Technical University in Prague
Faculty of Electrical Engineering
Technicka 2, 166 27 Prague 6, Czech Republic
E-mail: sramkma2@fel.cvut.cz

***Abstract*—In the present text, we deal with the problem of classification of speech emotion. Problems of speech processing are addressed through the use of artificial neural networks (ANN). The results can be use for two research projects - for prosody modelling and for analysis of disordered speech. The first ANN topology discussed is the multilayer neural network (MLNN) with the BPG learning algorithm, while the supervised SOM (SSOM) are the second ANN topology. Our aim is to verify the various of knowledge from phonetics and ANN but also to try to classify speech signals which are described by musical theory. Finally, one solution is given for this problem which is supplemented with a proof.**

I. Introduction

MANY problems in technology, medicine, and the natural and social sciences still remain unsolved: the complexity of solutions, the importance of time, and the considerable quantity of data required for processing form the real cause of the situation. Seeking help through new information technology is highly appropriate; and one such method is through the development of artificial neural networks (ANN). Success in the application of ANN depends on thorough knowledge of their function, which cuts across a wide range of academic disciplines – mathematics, numerous technical fields, physiology, medicine, phonetics, phonology, linguistics and social sciences. Initially, the ANN paradigm was regarded as a cure-all for many problems, yet simultaneously was often disparaged by its detractors for its inability to solve increasingly high requirements through the use of simple principles. The robustness of the solution for real methods by ANN is a great advantage, for example, in the area of noise signal processing. In this case, ANN should be a highly useful source of help, and the results thus acquired could be of a higher quality than those found with standard methods. The research goal described in this contribution was to verify an ability to classify optional speech through the use of ANN. We use three approaches for comparison of results. First, a frequency dependence and statistical parameters are created from input data, while the second approach is based on music theory (see [8]) and the final approach is a combination of both cases.

The contribution has two parts. A brief notice about some publications from international researchers which concerns emotional speech, basic information about emotions, and the specific ANN applied to the experiments create the first part of the text. The second part is dedicated to the results of the experiments themselves.

A. Classification of emotions in publications of international researchers

Much research around the world is engaged in the processing of emotional speech. Specific projects differ in the number and type of classified emotions, acoustic characteristics, the type of classifiers, and precision. Classification of three emotions (sadness, anger and neutral state) for human-computer communication are described in [13]. Fundamental frequency F_0, voice intensity and cepstral coefficients were the input characteristics. Classification success was 64 % with the classification of the five classes (for anger, pleasure, sadness, surprose, and neutral state) described in [14] and [15]. Data from Danish Emotional Speech were tested by the Bayes classifier and classification success was 54 % (for both gender), 61.1 % for males and 57.1 % for females respectively. Also the five emotions are classified in [12] (fear, pleasure, sadness, anger, and neutrale state). A Gausse SVM (Support Vector Machine) algorithm was applied with a 55 % success rate. A comparison of the SVM, RBF (Radial Basis Function), kNN (k-Nearest Neighbours), Naive Bayes and MLNN (two hidden layers with 15 neurons) is described in [9]. The success for the five classes was 81 %. A description of the five emotional states (pleasure, sadness, fear, anger, and neutral state) is undertaken in [11]. An algorithm is based on relationship of a height note versus the 12 half tones of the melodic scale. The latest publication is closest to our methods described in this contribution.

II. Solving the problems and an applied methods

Automatic speech synthesis is an interdisciplinary part of artificial intelligence, drawing upon knowledge from acoustics, phonetics, phonology, linguistics, physiology, psychology, signal processing and informatics for a successful solution. Many research teams around the world are engaged in the modelling of the prosody of synthetic speech. This problem

This research was supported by grant GACR No. 102/09/0989.

must be solved with relation to the specific attributes of different languages: e.g. [4] for English, [5] for German, [1] for French, [3] for Japanese for example. A majority of prosody control systems are based on the implementation of grammatical rules e.g. realised by decision trees, but some researchers (Sejnowski, Traber), including the authors of this study, use neural networks for prosody modelling. Different input parameters with a significant impact on speech prosody have to be used for neural network training in different languages. As a result, it is very difficult, indeed nearly impossible, to compare the results of prosody controllers for different languages. The most complex evaluation is the listening test, but it is very subjective and cannot be described by an objective metric. A reason for this difficulty is that prosody is deeply affected by the speaker's individual physiologies and mental states, as well as by the uttered speech segments and the universal phonetic properties. The influence of the phonological and phonetic properties of the Czech language, the influence of the quality, size of the speech database, and the influence of the synthesizer type all need to be explored. Furthermore, it is not possible to make complete use of all the information extracted from natural speech signals in automatic input data creation. Our research has taken as its central focus the question of prosody modelling for Text-to-Speech (TTS) Synthesis. A text and its speech signal will be used for the training process of ANN, and only the text and the trained ANN will be used for prosody modelling, allowing it to be as natural as possible. Previously, processing speech had a neutral character, yet in recent months research has concentrated its attention on emotional speech.

A. Basic information about speech emotions

Emotion is a mental state of a living organism accompanied by motive and glandular activities. Emotions are classified according to their psychological aspect. As a result, the term "emotion" represents physiologic disturbance, shock or attack. The second category – attitude – represents a behaviour and a chronic state. Feigned and active emotions have different manners of their division. A physiological reaction (change of cardiac rate and blood pressure, whiteness or redness) is linked to opposite emotions (anger, fear, pleasure, and sadness). Hence it is impossible use this physiological reaction as ANN input features separately. However, it is possible to use prosody characteristics, such as timbre, intensity and rhythm. These is a change of a fundamental frequency f_0, range of fundamental frequency, change of a formant location etc.

The melody, i.e. change of a haight of voice in a sentence, is very important from the point of view of a communication. Expressive changes of melody are important indicators for an emotional and voluntary attitude of a speaker (more in [17]).

B. Classification of emotions by ANN

ANN was used for classification of emotions. A multilayer neural network with one hidden layer was one of the methods applied for the classification of speech emotions. The number of neurons in the input layer is given by the key linguistic parameters which are needed for characterization of the Czech language. The ANN outputs are the various classes of emotions. The target values of prosodic parameters were extracted from the natural speech signal. Many learning algorithms for feed-forward neural networks are based on the gradient descent algorithm. Usually, they have a poor convergence rate and depend on input parameters which characterize specific problems. No theoretical basis for choosing optimal parameters for ANN training exists, but the values of these parameters are often crucial for the success of the training. Therefore we decided to use a Scaled Conjugate Gradient (SCG) algorithm with superlinear convergence rate. SCG belongs to the class of Conjugate Gradient Methods, which shows superlinear convergence for most problems; further description is offered in [19].

Kohonen's Self-Organizing Features Map (KSOM) was the second ANN which was applied for the solution of emotion classification. KSOM is a form of ANN that is trained by unsupervised learning rules, i.e. without target (required) values. It is an iterative process based on the clustering method; cluster analysis methods searching for interdependences and joint properties in a set of submitted patterns. A new SOM variant has been used for emotion classification, namely the supervised Self-Organizing Map (SSOM), which combines aspects of the vector quantization method with the topology-preserving ordering of the quantization vectors. The algorithm of the SSOM represents a very effective method of classification, but only for well-known input data or for well-known classes of input data.

C. Corpus creation

For testing and refining the ANN, it is necessary to create a speech corpus of sentences and, through pre-processing of the corpus, to prepare input data for the network's training and testing. In general, corpuses of natural speech have been created through careful choice from among a wide variety of different neutral sentences. Currently, no emotional speech database is available. As a result, an emotional speech corpus and database for ANN training had to be created for our research. The sentences was read by professional actors, two female and one male. Speech recording was materialized in a recording studio with a professional equipment (format "wav," sampling frequence $44.1kHz$, $24bit$).

The speech corpus is composed of a written text and its corresponding speech signal, both of which will be used for the training of ANN. The compound corpus was divided into two parts, the first set used for training and the second part serving as a testing set, also used for the monitoring of the training process.

Utterances were realised for four types of emotions: anger, boredom, pleasure and sadness – see Table I and Table II.

D. Input data creation

The success of prosody control is clearly dependent on the labelling of the natural speech signal in the database. The labelling (determination of boundaries between speech

TABLE I
DATABASE OF UTTERANCES – IN CZECH

Words (in Czech)	Words – translation
Jé.	*Whoah.*
Má?	*Got it?*
Nevím.	*I don't know.*
Vidíš?	*See you?*
Povídej!	*Tell me!*
Poezie.	*Poetry.*

TABLE II
DATABASE OF UTTERANCES – TRANSLATION INTO ENGLISH

Sentences (in Czech)	Sentences – translation
To mi nevadí.	*I don't mind.*
Neumím to vysvětlit.	*I don't know to explain this.*
To bude světový rekord.	*It will be a world record.*
Jak se ti to líbí?	*How do you like it?*
Podívej se na nebe!	*Look up at the heavens!*
Až přijdeš uvidíš.	*When you come, you'll see.*

units) and phonetic transcription of sentences from the speech corpus is done in the phase of pre-processing. The changes of fundamental frequency F_0, formant frequency F_i, $i = 1, \dots, 4$ and duration Du of phonemes during the voicing of sentences create the melody of the sentence (its intonation). Intonation is also related to the meaning of the sentence and its emotional timbre.

Recorded emotion speech was subjectively evaluated by four persons. The final database contained 720 patterns (360 patterns for one-word sentences and 360 patterns for multi-word sentences).

III. Experiments

All analyses and experiments described in this contribution were performed through use of the computational system MATLAB with NN-toolbox [16] and SOM Toolbox. SOM Toolbox was developed in the Laboratory of Information and Computer Science (CIS) in the Helsinki University of Technology and it is built using the MATLAB script language. The SOM Toolbox contains functions for creation, visualization and analysis of the Self-Organizing Maps. The Toolbox is available free of charge under the General Public License from ([7]). For the projects from the domain of the speech processing by ANN (which are being addressed by our university's department of Circuits Theory), new special M-files, which should be a part of the supporting program package, were created.

MLNN and SOM were applied particularly to the utterances from Table I and Table II. The results from MLNN training are concentrated into the so-called matrix of replacement, where "class 1" is specified as anger, "class 2" is specified as boredom, "class 3" is specified as pleasure and "class 4" is specified as sadness. The database for ANN training obtained 216 patterns, for validation 72 patterns and for testing as many as 72 patterns.

TABLE III
INPUT PARAMETERS – TIME AND FREQUENCY DOMAIN

Time domain
Arithmetic average of absolute value
Standard deviation
Maximum
Minimum
Frequency domain
Fundamental frequency F_0
Formant frequency $F_1, \dots F_4$

The unified distance matrix or U-matrix is a representation of the KSOM that visualizes the distance between the neurons and their neighbors. The KSOM neurons are represented by hexagonal cells (in our experiment). The distance between the adjacent neurons is calculated and presented in different colors. Darker colors between neurons correspond to a larger distance and thus represent a difference between the values in the input space. Light colors between the neurons mean that the vectors are close to each other in the input space. Light areas represent clusters (classes) and dark areas represent cluster boundaries (more in [2]). The size of the map was 15x15, while quantization (QE) and topographic (TE) errors of the map were also computed.

A. Method I: The patterns based on time and frequency characteristics

Nine patterns for MLNN training are created through the characteristics of the time and frequency domains (see Table III). The hidden layer was 20 neurons, while the output layer was 4 neurons. The number of training epoch was 56 resp. 53 for one-word sentences resp. multiword sentences.

B. Method II: The patterns based on musical theory.

The second presented method is based on the idea of the musical interval: the frequency difference between a specific n-tone and reference tone. E.g. quint is ratio of the fifth tone divided by the first tone, with a numerical value of 1.498. The ratios of the musical intervals are shown in Table IV.

TABLE IV
FREQUENCY RATIOS OF THE MUSICAL INTERVALS

Interval	Variant	Frequency ratios
first		1,000
second	minor	1,059
	major	1,122
third	minor	1,189
	major	1,260
fourth		1,335
fifth		1,498
sixth	minor	1,587
	major	1,682
seventh	minor	1,782
	major	1,888
octave		2,000

TABLE V
COMPARISON OF THE RESULTS

Method	one-word sentences		multiword sentences	
	MLNN [%]	SOM [QE/TE]	MLNN [%]	SOM [QE/TE]
I	88.7	0.185 / 0.02	77.8	0.184 / 0.017
II	70.4	0.274 / 0.014	65.3	0.275 / 0.017
III	85.9	0.431 / 0.011	84.7	0.439 / 0.006

This method of the parametrization of the utterances consists in the description of the signal patterns based on the musical intervals or more precisely on their frequency ratios. The reference frequency, i.e. the fundamental frequency in our case, is given by the choices in each utterance feature. We use autocorrelation function. The frequency ratios are compared with the music intervals and input vector for MLNN training with 20 values is computed. The hidden layer was 35 neurons, the output layer was 4 neurons. The number of the training epoch was 75 for one-word sentences resp. 84 for multiword sentences.

C. Method III: Combination of both previous approaches

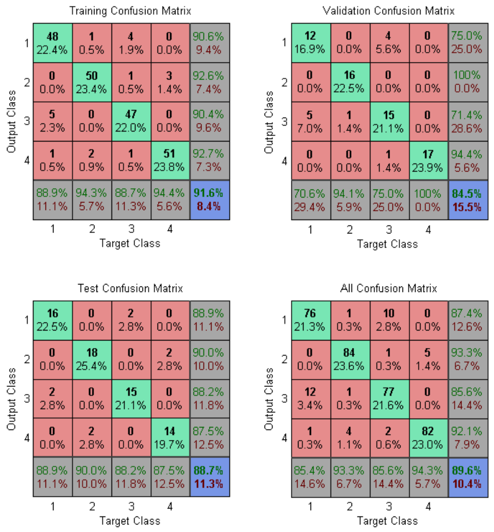
The third method was the combination of both previous methods. 29 patterns for MLNN training are created by 20 values containing the ratios respective to the music intervals and 9 values describing the acoustic qualities of the utterance feature. The hidden layer was 55 neurons, while the output layer was 4 neurons. The number of the training epoch was 57 for one-word sentences resp. 106 for multiword sentences.

IV. RESULTS OF EXPERIMENTS

The results of experiments are shown in the following table and figures.

This table summarizes the success rate of emotional classification for all three described methods. For the MLNN approach the first method (based on acoustic parameters) was best for the one-word utterances, but difference between one-word and multiword utterances are the greatest (9.9 %). Success for the second method (based on music theory) is worse for both type of utterances, but the difference between them is smaller (5.1 %). The third method (combination) is the best of them for the multiword utteraces, while additionally the differences between one-word and multiword utterances are absolutely smallest (1.2 %). With the SOM approach, the determining of SOM quality is complicated. We were monitoring the quality of learning by topographic and quantization errors for the comparison of methods. The topographic error (TE) predicates the conservation of data topology between input and output space. The quantization error (QE) reflects the accuracy of the mapping (it relates to the number of the input matrix elements and the size of the map).

The success of the SOM training decreases when the number of map units is larger than the number of training samples, which may be the main problem in our SOM approach. In our experimets we made use of the uniform size of maps for the comparison of all three methods. Our results show progressive values of the quantization errors in dependence to number of training data features, whereas a decreasing value for topographic error shows a very good ability of the classification.



Training Confusion Matrix

Output Class	1	2	3	4	
1	48 22.4%	1 0.5%	4 1.9%	0 0.0%	90.6% 9.4%
2	0 0.0%	50 23.4%	1 0.5%	3 1.4%	92.6% 7.4%
3	5 2.3%	0 0.0%	47 22.0%	0 0.0%	90.4% 9.6%
4	1 0.5%	2 0.9%	1 0.5%	51 23.8%	92.7% 7.3%
	88.9% 11.1%	94.3% 5.7%	88.7% 11.3%	94.4% 5.6%	91.6% 8.4%

Target Class

Validation Confusion Matrix

Output Class	1	2	3	4	
1	12 16.9%	0 0.0%	4 5.6%	0 0.0%	75.0% 25.0%
2	0 0.0%	16 22.5%	0 0.0%	0 0.0%	100% 0.0%
3	5 7.0%	1 1.4%	15 21.1%	0 0.0%	71.4% 28.6%
4	0 0.0%	0 0.0%	1 1.4%	17 23.9%	94.4% 5.6%
	70.6% 29.4%	94.1% 5.9%	75.0% 25.0%	100% 0.0%	84.5% 15.5%

Target Class

Test Confusion Matrix

Output Class	1	2	3	4	
1	16 22.5%	0 0.0%	2 2.8%	0 0.0%	88.9% 11.1%
2	0 0.0%	18 25.4%	0 0.0%	2 2.8%	90.0% 10.0%
3	2 2.8%	0 0.0%	15 21.1%	0 0.0%	88.2% 11.8%
4	0 0.0%	2 2.8%	0 0.0%	14 19.7%	87.5% 12.5%
	88.9% 11.1%	90.0% 10.0%	88.2% 11.8%	87.5% 12.5%	88.7% 11.3%

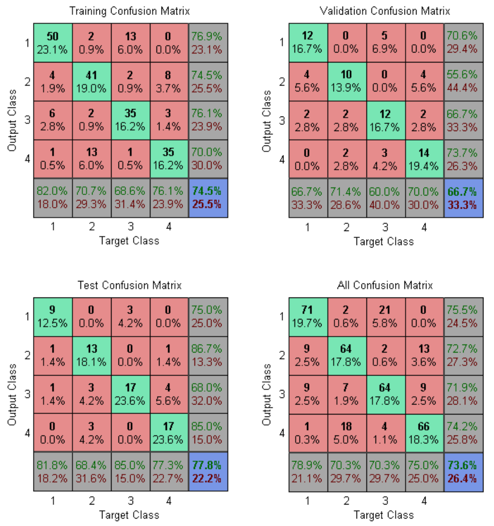
Target Class

All Confusion Matrix

Output Class	1	2	3	4	
1	76 21.3%	1 0.3%	10 2.8%	0 0.0%	87.4% 12.6%
2	0 0.0%	84 23.6%	1 0.3%	5 1.4%	93.3% 6.7%
3	12 3.4%	1 0.3%	77 21.6%	0 0.0%	85.6% 14.4%
4	1 0.3%	4 1.1%	2 0.6%	82 23.0%	92.1% 7.9%
	85.4% 14.6%	93.3% 6.7%	85.6% 14.4%	94.3% 5.7%	89.6% 10.4%

Target Class

Fig. 1. Method I: Matrix of changes for one-word sentences – time and frequency domain parameters



Training Confusion Matrix

Output Class	1	2	3	4	
1	50 23.1%	2 0.9%	13 6.0%	0 0.0%	76.9% 23.1%
2	4 1.9%	41 19.0%	2 0.9%	8 3.7%	74.5% 25.5%
3	6 2.8%	2 0.9%	35 16.2%	3 1.4%	76.1% 23.9%
4	1 0.5%	13 6.0%	1 0.5%	35 16.2%	70.0% 30.0%
	82.0% 18.0%	70.7% 29.3%	68.6% 31.4%	76.1% 23.9%	74.5% 25.5%

Target Class

Validation Confusion Matrix

Output Class	1	2	3	4	
1	12 16.7%	0 0.0%	5 6.9%	0 0.0%	70.6% 29.4%
2	4 5.6%	10 13.9%	0 0.0%	4 5.6%	55.6% 44.4%
3	2 2.8%	2 2.8%	12 16.7%	2 2.8%	66.7% 33.3%
4	0 0.0%	2 2.8%	3 4.2%	14 19.4%	73.7% 26.3%
	66.7% 33.3%	71.4% 28.6%	60.0% 40.0%	70.0% 30.0%	66.7% 33.3%

Target Class

Test Confusion Matrix

Output Class	1	2	3	4	
1	9 12.5%	0 0.0%	3 4.2%	0 0.0%	75.0% 25.0%
2	1 1.4%	13 18.1%	0 0.0%	1 1.4%	86.7% 13.3%
3	1 1.4%	3 4.2%	17 23.6%	4 5.6%	68.0% 32.0%
4	0 0.0%	3 4.2%	0 0.0%	17 23.6%	85.0% 15.0%
	81.8% 18.2%	68.4% 31.6%	85.0% 15.0%	77.3% 22.7%	77.8% 22.2%

Target Class

All Confusion Matrix

Output Class	1	2	3	4	
1	71 19.7%	2 0.6%	21 5.8%	0 0.0%	75.5% 24.5%
2	9 2.5%	64 17.8%	2 0.6%	13 3.6%	72.7% 27.3%
3	9 2.5%	7 1.9%	64 17.8%	9 2.5%	71.9% 28.1%
4	1 0.3%	18 5.0%	4 1.1%	66 18.3%	74.2% 25.8%
	78.9% 21.1%	70.3% 29.7%	70.3% 29.7%	75.0% 25.0%	73.6% 26.4%

Target Class

Fig. 2. Method I: Matrix of changes for multiword sentences – time and frequency domain parameters

We can see the matrix of changes for the MLNN classifier in Figure 1, 2. These figures summarize the first described method. We can observe the replacement of the emotion classification between active emotions, i.e. pleasure – anger, and between passive emotions, i.e. sadness – tedium. The results from the second method are shown in Figure 3, 4, the results from the third method are demonstrated in Figure 5, 6. Just as in the Method I, the worst score is for passive emotions (Method II and Method III).

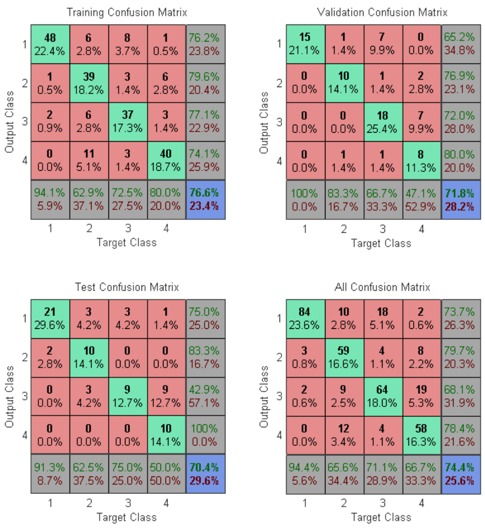


Fig. 3. Method II: Matrix of changes for one-word sentences – time and frequency domain parameters

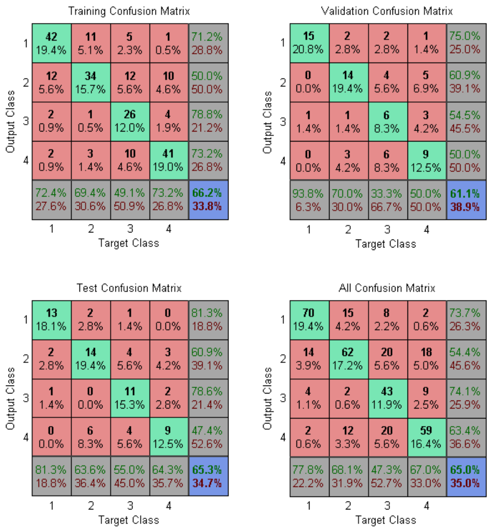


Fig. 4. Method II: Matrix of changes for multiword sentences – time and frequency domain parameters

The unified distance matrix or U-matrix is a representation of the KSOM that visualizes the distance between the neurons and their neighbors. The KSOM neurons are represented by hexagonal cells (in our experiment) marked by 'H' for anger, 'N' for tedium, 'R' for pleasure and 'S' for sadness. Each cell is marked also by a character for class, by real classified font and number registered pattterns.

The distance between the adjacent neurons is calculated and presented with different colors. Dark colors between neurons correspond to a larger distance and thus represent a difference between the values in the input space. Light colors between the neurons means that the vectors are close to each other in the input space. Light areas represent clusters and dark areas represent cluster boundaries.

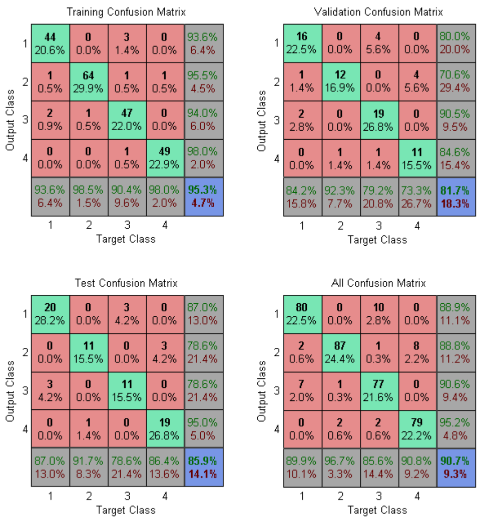


Fig. 5. Method III: Matrix of changes for one-word sentences – time and frequency domain parameters

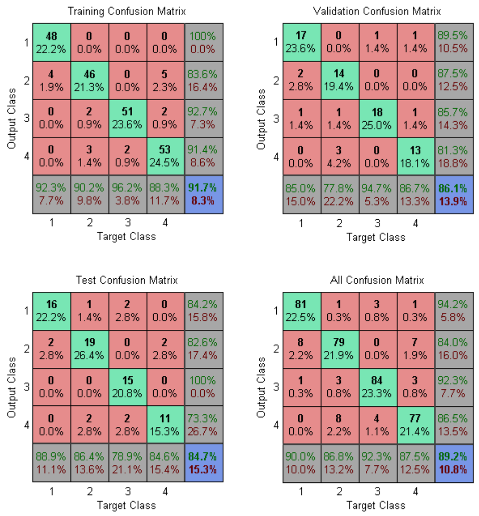


Fig. 6. Method III: Matrix of changes for multiword sentences – time and frequency domain parameters

The U-matrix represents emotion classes based on the parametrization by Method I are visualize in Figure 7, resp. 8, by Method II in Figure 9, resp. 10 and by Method III in Figure 11 and 12. The matrix is divided into four parts respective particular emotions. The topographic error for both type of utterances (one-word and multiwords) is lowest from all three methods (see on the Table V). This result documents the availability to apply the method based on the combination standard and music theory.

V. Conclusion and Future Work

We have established differentiation between the mathematical results and the listening tests. It is necessary to judge recording of emotional speech which is determined for database creation and the resulting synthetic sentences

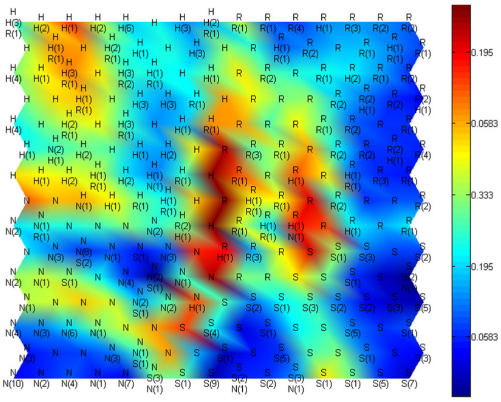


Fig. 7. Method I: U-matrix for one-word sentences

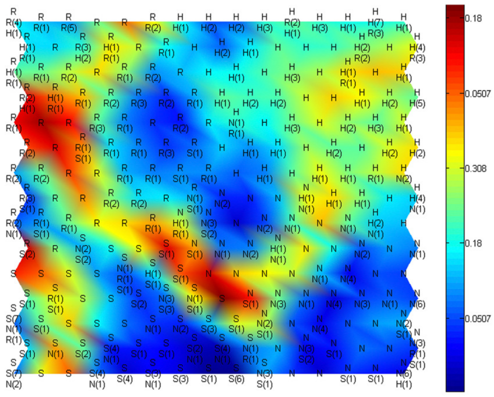


Fig. 8. Method I: U-matrix for multi-word sentences

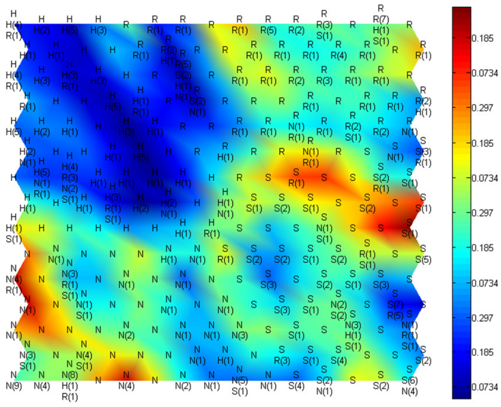


Fig. 9. Method II: U-matrix for one-word sentences

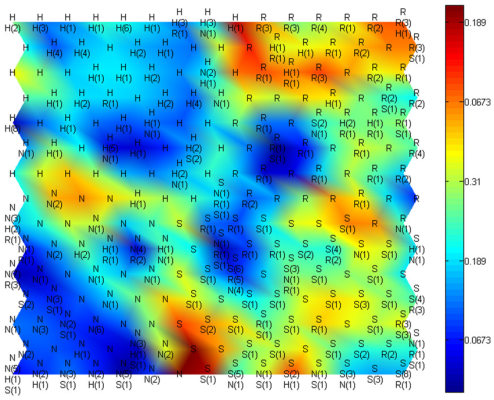


Fig. 10. Method II: U-matrix for multi-word sentences

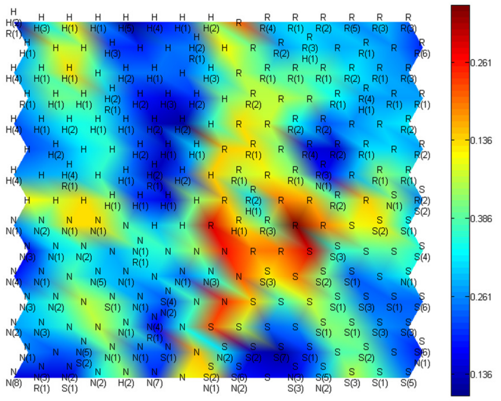


Fig. 11. Method III: U-matrix for one-word sentences

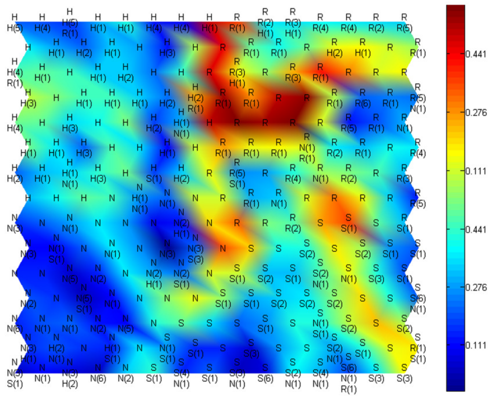


Fig. 12. Method III: U-matrix for multi-word sentences

by listening. The physical properties of the acoustic wave, which are perceived as sounds, are transformed several times: first in the organ of hearing, later at the emergence of neural excitement, and last in the cerebral analysis. Therefore, the sounds perceived in listening tests do not correlate completely with the objective properties of the acoustic patterns. The listening tests must consequently form a part of experiments. We want to conclude by establishing when to use MLNN or SSOM on the base of acquired events and on the base of augmentation, of the number of experiments.

Our effort in future work will focus on our lack of knowledge regarding the possibilities of ANN application in prosody modelling and children's disordered speech analysis. These different domain of the application influence the database creation. Long sentences (called multiword sentences in this contribution) are more acceptable for prosody modelling, yet the database created by one-word sentences is suitable for the analysis of children's disordered speech (often a speech malfunction is manifested in an inability to pronounce whole sentences). We are going to apply results from the described experiments with emotional speech to the improvement of synthetic speech naturalness, but also to the domain of neurodevelopmental disturbances (above all, developmental dysphasia).

REFERENCES

[1] E. Keller, S. Werner " Automatic Intonation Extraction and Generation for French." 14th CALICO Annual Symposium. ISBN 1-890127-01-9, West Point. NY, 1997.

[2] T. Kohonen *Self-Organizing Maps.* Ed.:Huang, T. S., Kohonen, T. ,Schroeder, M. R., 3rd ed.Springer-Verlag Berlin, 2001, ISBN 3-540-67921-9.

[3] Z. Sagisaka, T. Yamashita and Y. Kokenawa " Generation and perception of F0 markedness for communicative speech synthesis." *Speech Communication*, 2005, Vol. 46, Issues 3–4, pp. 376–384.

[4] T. J. Sejnowski, C. R. Rosenberg " NETtalk: A parallel network that learns to read aloud". *Technical Report JHU/EECS-86/01, The Johns Hopkins University Technical Report*, 1986.

[5] C. Traber " F0 generation with a database of natural F0 patterns and with a neural network." G.Bailly,C.Benoit, and T.R. Sawallis, ed., *Talking Machines: Theories, Models, and Design*, pp. 287–304. Elsevier Science Publishers,1992.

[6] J. Tuckova, V. Sebesta "The Prosody Optimisation of the Czech Language Synthesizer." *Int. Journal on Neural and Mass-Parallel Computing and Information Systems "Neural Network World"*, Ed. M. Novak, ICS AS CR and CTU, FTS, vol. 4, 2008, pp. 291–308 . ISSN 1210-0552.

[7] J. Vesanto, J. Himberg, E. Alhoniemi and J. Parhankangas "SOM Toolbox for Matlab 5," SOM Toolbox Team, Helsinki Univesity of Technology, Finland, 2000, ISBN 951-22-4951-0. Homepage of SOM Toolbox: www.cis.hut.fi/projects/somtoolbox

[8] M. Cerny, *Influence of the speech signal parametrization to prosody modelling. (in Czech)*, Diploma work, Prague, CTU FEE 2009.

[9] Z. Xiao, E. Dellandrea, WW. Dou, and L. Chen "Multi-stage classification of emotional speech motivated by a dimensional emotion model,"*Multimedia Tools and Applications journal*, Springer Netherlands, vol. 46, Nu 1/January, pp. 119–145, ISSN 1380-7501.

[10] M. Shami, W. Verhelst, "Automatic Classification of Expressiveness in Speech: A Multi-corpus Study". *In Speaker Classification Ii: Selected Projects, C. Müller*, Ed. Lecture Notes In Artificial Intelligence, vol. 4441. Springer–Verlag, Berlin, Heidelberg, 2007, pp. 43–56.

[11] A. M. Mahmoud, W. H. Hassan, "Determinism in speech pitch relation to emotion". *Proceedings of the 2nd international Conference on interaction Sciences: information Technology, Culture and Human* Seoul, Korea, November 24–26, 2009, vol. 403, ACM, New York, NY, pp. 32–37.

[12] S. McGilloway, R. Cowie, Ed. Cowie, S. Gielen, M. Westerdijk, S. Stroeve, "Approaching automatic recognition of emotion from voice: a rough benchmark," *Proceedings of the ISCA workshop on Speech and Emotion*, pp. 207–212, Newcastle, Northern Ireland, 2000.

[13] T. Polzin, A. Waibel, "Emotion-sensitive human-computer interfaces," *Proc.of the ISCA workshop on Speech and Emotion*, pp. 201–206, Newcastle, Northern Ireland, 2000.

[14] D. Ververidis, C. Koltopoulos, "Automatic speech classification to five emotional states based on gender information," *Proc. of 12th European Signal Processing Conference*, pp. 341–344, Austria, 2004.

[15] D. Ververidis, C. Koltopoulos, I. Pitas, "Automatic emotional speech classification". *Proc. of ICASSP 2004*, pp. 593–596, Montreal, Canada, 2004.

[16] *MATLAB Help* version 2009a Natick, Massachusetts: The MathWorks Inc., 2009.

[17] M. Krcmova, *Phonetics and phonology* in Czech *Fonetika a fonologie* [online]. Brno : Masarykova univerzita, 2008 [cit. 2010-04-04]. http://is.muni.cz/elportal/?id=766384. ISSN 1802-128X.

[18] M. F. Møller, "A scaled conjugate gradient algorithm for fast supervised learning."*NEURAL NETWORKS*, vol. 6, nu 4, pp. 525–533, 1993.

[19] http://www.mathworks.com/access/helpdesk_r13/help/toolbox/nnet/trainscg.html

ISBN 978-83-60810-27-9
ISSN 1896-7094

Usage of reflection in .NET to inference of knowledge base

Marek Vajgl
Ostravská Univerzita v Ostravě,
Přírodovědecká fakulta, katedra
informatiky a počítačů. 30. dubna
22, Ostrava, Czech Republic.
Email: marek.vajgl@osu

***Abstract*—This document describes how information generated by integrated development environment (namely Visual Studio 2008) can be used to generate and usage of knowledge base. The main aim is to explain, how the approach of extension of currently implemented software project can achieve knowledge representation, and how already created and implemented data types can be used to create knowledge bases using modern language's development environment and behavior. The article is aimed to the Description Logic formal system, but can be applied to any formal deduction mechanism.**

I. Introduction

Today, lot of information systems are built in non-research area and most of them use for data or knowledge storage standard relational databases, object databases or other storage places, like XML files.

This, often very large structures, contain a lot of information about company (or any targeted subject), which can be manipulated as knowledge and shared. Information systems or applications from design development phase through fully implementations contains well considered, well organized, and well described and documented schemas (like database models, class models), which are not taken and mentioned as knowledge base and does not contains any kind of inference mechanism. Nevertheless a lot of that information can be taken as concepts/roles/instances and used to create full knowledge base with inference mechanism. Moreover, this process can be done automatically due to source format – implemented diagrams, classes in object oriented programming, etc. However, only minor part use some kind of inference or deduction mechanism, like expert system, or some kind of formal deduction based over knowledge bases.

Although the main reason is not sufficient spread of knowledge mechanism approaches (like expert systems are mostly oriented in technological industry or medicine), one of the another reasons is inability to check, test, or prove those systems on created project, that will simply show the advantages (and disadvantages) of those approaches. If aimed on knowledge bases, and one of the most today's formal system, description logic, the existing implementations often requires to install third party runtime environment (e.g. Java Virtual Machine for .NET users), or are integrated as server applications listening on special ports with special syntax requested for input and output for test. True is, that most of those systems are supported by today's standard for knowledge base/ontology editor and designer Protégé [6], there is again a load on a developer to 1. Find, install and open knowledge base system, 2. Install and execute Protégé system, 3. Learn how to insert data from existing information system and 4. Learn how to profit from those combinations against the classical approaches mentioned above. A lot of other approaches are built on client-server principles, where information system as client has to send its requests to the server realizing inference over knowledge base. Those approaches again includes requirement to create and use any implementation of format of communication protocol(s) between client and server.

This approach is for "common" developer almost impassable, so there are open doors for other approaches, which brings those formal systems into the existing system of developers. The main idea is to extend developer's existing project. The added part will consist of a solution, which can (partially) automatically analyze existing project, generate knowledge base and offer a mechanism to inheritance or deduction over gained data.

Lot of integrated development environments (like Microsoft Visual Studio) offers mechanism providing automatic generating of the content into some universal format (like XML files), moreover, modern programming languages contains mechanism allowing runtime analysis of any compiled code, called "Reflection". Those two built stones open a way how can (and should) knowledge system use those data to offer a developer a ways to use knowledge representational approach in application development.

Here presented solution – XReasoner system – manages three main parts of operation with data presented: a) creation of the knowledge base – T-Box and A-Box, b) managing knowledge base, c) sharing information with other sources based on other modern phenomenon, web services. This article describes first part of this behavior. (System is built over description logic DL1 [1][3] and uses semantic tableau algorithm for inference. Idea of approach presented in this article is mainly commonly applicable over other systems; at least, for description logic (above common constructors) is required constructor of negation (to implement exclusiveness of derived classes), value restriction (for properties and relations) and sometimes quantified number restriction (for "property" behavior in .NET languages) –$\mathcal{ALCQ}$).

One more condition is important. Developer should not (at least in testing phase) adapt or rewrite the existing code. Therefore, the introduced solution, where able, presents two

978-83-60810-27-9/09/$25.00 © 2010 IEEE

approaches: first one with changes to origin source code; second one which remains origin source code intact.

II. Description logic

At the beginning of this contribution, only the short introduction of the formal system of description logic will be mentioned. This formal system is very popular today, for its simplicity, expressivity. Moreover, we prefer it due to its close approach to the object oriented paradigm of programming.

Description logic (presented according to [7]) is a formalism which arises from concept languages. Concept languages are built on description of abstract objects of real world – called concepts, and roles, representing relations between those concepts. In description logic, the concepts are ordered into hierarchical structures. Knowledge base of description logic consists of definition of concepts and their roles (both those items represent intensional knowledge) and instances of the concepts (instances represents extensional knowledge about concrete object of the real world).

From the viewpoint of formal systems for knowledge representation is description logic built on formalism supporting frames, associative networks and object oriented knowledge representation.

Knowledge base of description logic is formally divided into two parts. First part is called T-Box (terminological box) and contains terminological intensional knowledge in the form of concepts and definition of relations between those concepts. Second part is called A-Box (assertional box) and consists of facts about instances – individuals – where each instance is related to the concrete concept or role. Each instance's domain is one or more concepts in T-Box.

Moreover, as mentioned before, concepts in T-Box are automatically ordered into hierarchical structure according to theirs definition. They are classified and placed (subsumed) into taxonomy – this behavior is called *subsumption*.

III. IDE, Reflection and Automatization

As mentioned before, the main aim is to utilize the behavior of modern programming languages and modern integrated development environment. The contribution is aimed on Microsoft's .NET technology (and examples will be presented in C# programming language), because it is in close development with the XReasoner tool – that is description logic implementation over the .NET platform, created to support this behavior. The two main tools of the modern software development in .NET bring opportunity to automatic knowledge base generation from existing application or information system.

The first construct is automatization. The modern IDE (like mentioned Visual Studio) offer tools to generate some universal data (like xml files) from existing classes into structures also called *DataSets*. IDE is able to capture classes, relation between classes (with occurrences count), inheritance, interface implementations into the xml files. This information is not only generated from classes, but databases or source XML files can be used too. So, IDE offers easy way how to extract terminological part of software's "knowledge base" into the xml file, which can be later analyzed.

The second construct is more powerful. It does not rely on development environment, but is part of programming language, and is called *reflection*.

Reflection [5] is ability of modern programming language (including languages used to development over the Microsoft .NET platform) to achieve, process and invoke compiled source code dynamically during application run. Simple said, for example, during application run can a programmer choose library, load a type from it, create an instance and invoke type members over this instance. In both most used programming languages, C# (.NET) and Java uses in compilation mechanism, which includes all information about data types, its members and relations in the compiled file (e.g. .dll assembly file in .NET platform). Due to this behavior reflection does not need any additional information from other files (like C++, where the .h files are required). This part of information are saved in the *manifest* of the assembly (assembly is .NET compiled file). In .NET, assembly is simply loaded (on request by programmer), its manifest is analyzed and the runtime environment "knows" which types can be found in the library, which members can be called and with which parameters. When the object oriented programming is taken into account, there must be mentioned that reflection can access not only public types and members, but also protected, private and internal (means visible only in current assembly) ones are visible and accessible with this mechanism.

When aimed to the behavior of modern programming, one more very powerful and important feature (accessible during reflection) should be mentioned. It is called *attributes*.

Attributes allows a developer to add some special kind of declaration (!) information to a class definition. This information can be later used by compiler or other executing code to adjust its behavior against marked class. For example, class declared this way:

```
[Serializable()]
public class X
{
   public int A;
   [NonSerialized()]
   public int B;
   …
}
```

Class X will be during serialization process successfully stored to a target location; moreover, the field B will be skipped and will not be saved.

Attributes are not built-in part of the language, it the view of language they are special cases of classes. So, the developer can create its own attributes and use them with classes. Other developers can then use those attributes to mark their classes.

This mechanism is now used in coordination with persistence into relational database systems (RDBS); each class has specified into which table, with which properties and which data-types, is stored/loaded. The same way can be used to specify which classes and which its properties can be used to knowledge base generating.

IV. Creation of Knowledge Base

Generally, the mechanism uses set of assemblies containing XReasoner system used to manage description logic knowledge base and set of assemblies containing attributes and classes used to generate knowledge base by adjusted properties. As mentioned in the introduction, two approaches can be used: external, which does not affect code of the existing code, and internal, which affects this code. External approach uses xml file to define requested behavior, however, sometimes internal behavior can be more readable and understandable. Both approaches can be used together, in case of conflict internal approach has higher priority and will be used.

Physically, implementation requires adding references to assemblies of XReasoner into current project (or creating new project and adding references to both, analysed project and XReasoner). If external approach is used, moreover xml file needs to be created and passed as parameter for XReasoner system.

The mechanism of generating knowledge base including concepts, roles and also instances will be divided and introduced in two parts: a) creation of T-Box, b) adding instances into A-Box.

A. T-Box - introduction

The main contribution of reflection technology is its "knowledge" of project classes and relations between them. This information can be presented by reflection simply by analysis of existing data types, their fields, properties and type inference hierarchy.

T-Box creation request to answer three questions: a) what data are available through reflection, b) which data are relevant and public and should be included, c) how to identity and identify different definitions representing the same object.

The first step is to retrieve all defined data types in selected assembly or assemblies. As mentioned before, reflection can provide (and provides) mechanism that returns simple listing of all data types defined in assembly. In .NET, there are a lot of data types, but only classes (reference types) and structures (value types) will be taken into account (that is important, i.a. interfaces are also omitted).

Next step is to analyze inference hierarchy between those data types. .NET languages are strictly object oriented; and therefore inheritance usage is very common. Information about data type also contains references to all its predecessors. With this information tree of hierarchy can be created.

Last part of T-Box analysis aims to properties and fields of the data types. Fields are not very often used to present some information value directly, because it in contrast with encapsulation principle of object oriented paradigm. Therefore properties are used instead. Properties contain information about their data-type and define strictly binary relations between two data types.

All those information can be used to define concepts and roles between them in T-Box. Basically, data-types will represent concepts in description logic knowledge base. Subsumption of concepts will be equal to data-type inheritance hierarchy. This creates only "simple subsumption" in meaning that there is only "Concept A is subsumption of concept B" relation, without any additional conditions. Programming language and object oriented paradigm ensure there will be no cyclic dependencies in definition.

As mentioned before, properties or fields define binary relations between two concepts. Those relations represent roles between concepts. First data type (the one containing the property or field) defines concept which is domain of the created role. Second data type (the one which is property or field data type) represents range of created role. Domain and range can be the same type; the behavior of properties or fields ensures that only binary roles are created.

B. T-Box - implementation

Before implementation description, one more behavior must be explained. Previous paragraph explains that roles always create relation between two objects. In C# language, this is correct. But in knowledge base a slightly different behavior should be required. Let's state an example. We have class "Person" and it has property representing a set of e-mail addresses. Data type of this set will be some collection (in C# language probably a list of strings (List<string>)). The relation is that person has its own list of strings, which contains his e-mail addresses. But the aim of knowledge base is aim to capture the knowledge "person and his e-mails", not "person and his list, containing emails". Therefore this decomposition can be made into multiple roles realizing relation between concrete person and one of its e-mail addresses. But, in this behavior the exact mapping between origin C# code and created knowledge base is lost. Therefore, developer can choose which one of those behavior he will prefer.

As same as in object oriented paradigm not all data types are public and visible to everybody, developer can make decision which of the source data-types will be used to create concept added into knowledge base. Some types can be omitted due to their irrelevance; some may be too complex and will make the knowledge base unnecessarily difficult. Therefore, developer has to mark which types he would like to include in the knowledge base.

As mentioned in the introduction, one of the aims is to have the source code intact. Therefore there are two ways how to achieve that. First one, external, uses xml file to define requested behavior, second one inserts requested info directly into source code. Internal approach is more intuitive and will be explained as first.

This approach uses predefined attributes to define which classes and how will be transformed into concepts; and also, which of its properties will be used. XReasoner defines two attributes:

- EKnowledgeClassAttribute – only classes with this attribute will be converted into concepts. This attributes also defines three properties: a) name – which defines name of the created concept (default name created from class name will be created if not specified); b) IdentityProperties – string defining which properties uniquely define the instances of the concept (s.t. like primary key in databases) – it is used later, in A-Box instantialization; c) URIs – string defining the unique resource identifier(s) for the project – it will be used in knowledge base sharing and is expected if two different concepts (from different sources) have the same URI, then they represents the same object in the real world.

- EKnowledgePropertyAttribute – only properties marked with this attribute will be used to represents roles of the parent class/concept. Again, it has two properties: a) IsIdentity – with the same meaning as above, but is specified for property directly; b) URIs with the same meaning as above, but for role.

Simple example of this approach follows (irrelevant parts of file are dotted):

```
using ENG.XReasoner.AssemblyAnalyser.Attributes;

namespace Solution
{
  [EKnowledgeClass(
    Name="Person",
    IdentityProperties="BornNumber",
    URIs="http://.../Person")]
  public class Person
  {
    [EKnowledgeProperty(IsIdentity=true)]
    public string BornNumber { get; set; }
    [EKnowledgeProperty()]
    public string Name { get; set; }
  }
}
```

This, intensional, approach is more difficult, because all required classes/concepts have to be marked with the attribute, but a developer has exact control over the process.

Extensional approach uses xml file to define requested behavior. Upper file can be created accordingly to DTD file (supplied with XReasoner solution), and (briefly) may looks like this (irrelevant parts of file are dotted):

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE …>
<EKnowledge>
  <TBox>
    <Assembly nameRgx=".+">
      <Behavior
        conceptNaming="fullClassName"
        includePropertiesRgx=".*"
        expandEnumerations="false" />
      <Concepts>
        <Concept
          typeName="Person" explicitName="Person" >
          <Property name="BornNumber"
          isIdentity="true" />
          <Property name="Name" isIdentity="false" />
          <Property name="Addresses" isIdentity="false" />
        </Concept>
        <ConceptRgx typeNameRgx=".*" >
        </ConceptRgx>
      </Concepts>
      <Others></Others>
    </Assembly>
  </TBox>
…
</EKnowledge>
```

Simple explanation of presented elements follows.

EKnowledge element covers the whole behavior of the XReasoner system, including not only how the knowledge base is defined, but also how is shared via web services (if any).

TBox element defines how the T-Box is created. This element contains of definition for one or more assembly.

Assembly element defines how the assembly will be treated. The “nameRgx” attribute defines which assemblies will be processed with this element’s setting. It is regular expression, which is matched against full assembly name. If assembly name matches more than one definition, first suitable is taken.

In the assembly, default behavior can be set. This element specifies, how the concepts will be named (is used only for classes/concepts without explicit name specification). Options are to use class name without namespace (here can arise problem with two classes with the same names, which cannot be later treated anyway!), or full class name used, including preceding namespaces. Second attribute, “includePropertiesRgx”, defines which properties will be included as roles of concept (if there is no explicit definition for concept). Again, this is regular expression which will be matched against property name (those names are distinct within class). Last parameter refers to the first paragraph of this chapter and defines, if enumerations (or collections) are expanded into special concept instance, or if multiple roles are created for each element of enumeration (collection). This dilemma has been explained above.

Next part contains definition for concept and conceptRgx elements, both nested under Concepts element.

Concept element describes behavior of one exact class/concept. It defines name of type (obligatory), explicit name for created concept (if any, default will be used), may contains behavior specification (same as explained before) and set of elements “Properties”. Those elements explicitly define which properties will be used to define role.

ConceptRgx element is definition which will be applied on all concepts matching “typeNameRgx” regural expression. Again, expression is matched against full type name. Moreover, ConceptRgx element can contain behavior specification (in the same format as mentioned for assembly element), which is expected to be more specific than any upper one.

There is implementation in the XReasoner solution, the classes “Analyser” and “Converter”. First one takes into its public static function “Analyse” two parameters: 1) name of assembly/list of names of the assemblies, which will be proceeded; 2) name of the xml file containing the definition mentioned in explicit approach explanation. The second one can be omitted (can be “null”). The result of the process will be set of some intermediate info defines which classes and how will be transformed. Second class, “Converter”, contains static function “Convert”, which accepts one argument in the input. It is the result of the “Analyse” function. This transformation produces a set of formulas (in the set of datatypes of XReasoner solution), which can be directly added into XReasoner knowledge base instance.

C. A-Box introduction

A-Box creation is the process where the instances of classes of the project are taken and transformed into instances of concepts and relations between those instances. From the previous part of transformation, there are defined concepts and roles in the terminological box of the knowledge base.

Obtaining instances of classes is a lot tricky, because it may include many different sources (database, xml files, multiple collections in the memory), in which may be the same instances (that means, instances representing the same real object) multiple, sometimes with changed properties, or may contain a lot of unnecessary individuals. Therefore it is required to involve the developer in this process. He has to specify how the instances are achieved and passes for the XReasoner. Its task is to achieve the instances only, analysis and parsing of the instances is made by XReasoner’s classes and is done according to created concepts and roles in the previous part of process.

Physically, the task for the developer is to create a class containing two static methods. They have a simple purpose, but, of course, implementation can be difficult.

First method is used to return names for all instances of current concept. The input for the method is name of the concept (this name must match with the created concept name in the previous part, obviously). The reason for this method is really only to return instance names (that means identifiers) only, it is not necessary to load those objects fully with their properties. This method is used to create list of all instances of current concept.

Second one is used to return all roles for the current instance. It accepts one parameter, defining the name of the instance, and returns a fully loaded object, which will be converted into the instances and roles for the knowledge base.

There is one major issue which is very important and must be taken into account. The presented mechanism, created by the developer, has to always assign the same identifier to the same object, regardless of the state or time of the program. That means, if e.g. person “Michal” changes its name (between calling those two methods) to “Marek”, it have to return still the same identifier. There are two intuitive ways how to assure this: a) create identifier by combination of identity properties (or primary keys). This can assure that instance can have recognizable identifier, bud brings more work for the developer, because he needs implement this for every class/concept in the project. Second approach profits (in the Microsoft .NET technology) from the method derived from highest abstract class “Object” to all its descendants (that are all classes). Every class has method “GetHashCode()”, which should be unique for every instance and therefore can be used to return the unique identifier. However, hash codes are quite long and absolutely unreadable for humans, so person analyzing the knowledge base will not see what every instance is on the first sight.

D. A-Box implementation

The main principle was already explained in previous part of the article.

The implementation of the developer (the created class) has not to implement any specific interface or derive from any class. Therefore, this class can be included in original project. However, developer can inherit from interface presented in XReasoner solution to ensure he uses correct parameters and return types.

As the implementation suggests, XReasoner mechanism again uses reflection to invoke defined methods with requested parameters and achieve requested information. It is done by class named “InstanceExtractor“. This class offers some methods to create knowledge base. According to invoked method and passed parameter it is able to return instances’ names only, as well full instance definition with all roles. The result of this methods are again processed by the “Converter” class, which creates set of formulas (defining instances in the set of data-types of XReasoner solution), which can be directly added into XReasoner knowledge base.

V. Conclusion

There are a lot of information systems, or at least programs, which does not use any of knowledge representation approaches. There can be a motivation for developers to extend already implemented systems, or include this approach to currently developed systems. This article brings simple way how the existing solution can be extended to offer knowledge representation without any penetrative changes in the origin source code. What is not covered by this article, this extension can later be used to next deduction over created knowledge base, and also for sharing partial or full knowledge base using web services and definitions of URIs for selected concepts.

References

[1] Vajgl, M., Lukasová, A. A, *Semi-Dedicable Semantic Tableau Proof System in a Description Logic DL1.* Proceedings of CSE 2008 International Scientific Conference on Computer Science and Engineering. Košice: Department of Computers and Informatics FEEI TU of Košice, 2008. s. 277-284. [2008-09-24]. ISBN 978-80-8086-092-9

[2] Vajgl, M., Lukasová, A. *RDF-model as Associative Network.* DATAKON 2009 - Sborník databázové konference. Praha: Vysoká škola ekonomická v Praze, 2009. s. 95-103. [2009-10-10]. ISBN 978-80-245-1568-7

[3] Lukasová, A., Vajgl, M. *Genzen-like Proofs in Description Logic DL1.* Proceedings of the Tenth International Conference on Informatics 2009. Košice: Technical Univerzity of Košice, 2009. s. 160-166. [2009-11-23]. ISBN 978-80-8086-126-1

[4] W3C. Extensible Markup Language. [Cited 13. 6. 2010] Web resource. http://www.w3.org/XML/

[5] Microsoft – *System Reflection namespace*. [Cited 5. 4. 2010]. Web resource. http://msdn.microsoft.com/en-us/library/136wx94f.aspx

[6] The Protégé *Ontology Editor and Knowledge Acquisition System* – [Cited 5. 4. 2010] Web resource. http://protege.standford.edu

[7] F Baader et col. (eds.): *The Description Logic Handbook – Theory, Interpretation and Applications.* Cambridge University Press, 2003. ISBN 0-521-78176-0.

ISBN 978-83-60810-27-9
ISSN 1896-7094

On the evaluation of the linguistic summarization of temporally focused time series using a measure of informativeness

Anna Wilbik and Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
Email: wilbik@ibspan.waw.pl, kacprzyk@ibspan.waw.pl

***Abstract*—We extend our previous works of deriving linguistic summaries of time series using a fuzzy logic approach to linguistic summarization. We proceed towards a multicriteria analysis of summaries by assuming as a quality criterion Yager's measure of informativeness of classic and temporal protoforms that combines in a natural way the measures of truth, focus and specificity, to obtain a more advanced evaluation of summaries. The use of the informativeness measure for the purpose of a multicriteria evaluation of linguistic summaries of time series seems to be an effective and efficient approach, yet simple enough for practical applications. Results on the summarization of quotations of an investment (mutual) fund are very encouraging.**

I. Introduction

WITH this paper we continue our previous works (cf. Kacprzyk, Wilbik, Zadrożny [1], [2], [3] or Kacprzyk, Wilbik [4], [5], [6]) which deal with the problem of how to effectively and efficiently support humans in making decisions concerning investments in some financial, notably in investment (mutual) funds. Decision makers are here mainly interested in future gains/losses. However, we follow the decision support paradigm, that is, we assume that the users are autonomous and we only support, not replace, him/her. We do not intend to forecast the future daily prices.

The available information concerns the history, or past, and this implies some problems. Basically in all investment decisions the future is the most important, and the past is irrelevant. But, we know only the the past, and the future is completely unknown. Behavior of the human being is to a large extent driven by his/her (already known) past experience. People usually tend to assume that what has happened in the past will also happen (to some, maybe large extent) in the future. By the way, this is the underlying assumption behind the statistical methods too! That attitude clearly implies that the past can be employed to help the human decision maker find a good solution. We follow here this path, i.e. we present a method to subsume the past, to be more specific the past performance of an investment (mutual) fund, by presenting results in a very human consistent way, using natural language statements.

This line of reasoning has often been articulated by many well known investment practitioners, and one can quote here some more relevant opinions. In any information leaflets of investment funds, one may always notice a disclaimer stating that "Past performance is no indication of future returns" which is true. However, on the other hand, for instance, in a well known posting "Past Performance Does Not Predict Future Performance" [7], they state something that may look strange in this context, namely: "…according to an Investment Company Institute study, about 75% of all mutual fund investors mistakenly use short-term past performance as their primary reason for buying a specific fund". But, in an equally well known posting "Past performance is not everything" [8], they state: "…disclaimers apart, as a practice investors continue to make investments based on a schemes past performance. To make matters worse, fund houses are only too pleased to toe the line by actively advertising the past performance of their schemes leading investors to conclude that it is the single-most important parameter (if not the most important one) to be considered while investing in a mutual fund scheme".

As strange as this may be, we may ask ourselves why it is so. Again, in a well known posting "New Year's Eve: Past performance is no indication of future return" [9], they say "…if there is no correlation between past performance and future return, why are we so drawn to looking at charts and looking at past performance? I believe it is because it is in our nature as human beings …because we don't know what the future holds, we look toward the past …".

And, continuing along this line of reasoning, we can find many other examples of similar statements supporting our position. For instance, Myers [10] says: "…Does this mean you should ignore past performance data in selecting a mutual fund? No. But it does mean that you should be wary of how you use that information …*Lousy performance in the past is indicative of lousy performance in the future…*". And, further: Bogle [11] states: "... there is an important role that past performance can play in helping you to make your fund selections. While you should disregard a single aggregate number showing a fund's past long-term return, you can learn a great deal by studying the *nature of its past returns*. Above all, look for consistency.". In [12], we find: "While

978-83-60810-27-9/09/$25.00 © 2010 IEEE

past performance does not necessarily predict future returns, it can tell you how volatile a fund has been". In the popular "A 10-step guide to evaluating mutual funds" [13], they say in the last, tenth, advise: "Evaluate the funds performance. Every fund is benchmarked against an index like the S&P500, BSE 200, etc. Investors should compare fund performance over varying time frames vis-a-vis both the benchmark index and peers. Carefully evaluate the funds performance across market cycles particularly the downturns".

We can quote more, and basically all of them stress the importance of looking at the past to help make future decisions, and also generally advocate a more comprehensive look not focused on single values but a very essence of past behavior and returns.

We have followed this line of reasoning in our past papers (cf. Kacprzyk, Wilbik, Zadrożny [1], [2], [3] or Kacprzyk, Wilbik [4], [5], [6]), i.e. to try to find a human consistent, fuzzy quantifier based scheme for a linguistic summarization of the past in terms of various aspects of how the time series representing daily quotations of the investment fund(s) behave. However, we have mainly concentrated on a sheer absolute performance, i.e. the time evolution of the quotations themselves. This may be relevant, and sometimes attractive to the users who can see a summary of their gains/loses and their temporal evolution. One can also use a maybe more realistic approach to take into account benchmarks of the particular funds as points of departure which does not change the essence.

Though the use of linguistic data summaries of past performance of the time series representing mutual fund quotations does take into account the importance (or "value") of time, in this paper we will go deeper into this issue by using some results from psychology, cognitive sciences and human decision making. Basically, we will employ some results by Ariely and Zakay [14] who consider the role of time in decision making.

In our case, those psychological analyses served the purpose of suggesting, and/or justifying a new types of protoforms of linguistic summaries of time series. Basically, in most of our works (cf. Kacprzyk, Wilbik, Zadrożny [1], [2], [3] or Kacprzyk, Wilbik [4], [5], [6]) we have used the following protoforms of the linguistic summaries of times series: "Among all y's, Q are P", exemplified by "among all segments (of the time series) most are slowly increasing", and "Among all R segments, Q are P", exemplified by "among all short segments almost all are quickly decreasing". Notice that we took into account, and so to say assign the same weights, the entire time series, i.e. all the segment.

Since in our case the analysis of time series is a highly human focused activity because its very purpose is to provide a human decision maker with some support for making (future) decision, we should take into account some inherent characteristics of time series and their evaluations that are consistent with the human perception of their relevance for the decision making process. One of the crucial aspects in this respect, which will be considered here is the importance of time in the sense that means and ends, like decisions and outcomes, have a carrying relevance and impact depending on the time moment when they occur. Basically, in virtually all cases what occurs in a more immediate past is more relevant and meaningful that what has occurred earlier. This temporal relationships change both the decisions and their evaluation as has been shown in psychology (cf. Ariely and Zakay [14] or Rachlin [15]). Among many approaches one can mention, for instance, a so called *temporal construal theory* by Liberman and Trope [16] who have shown that options are evaluated differently depending on time instants they come into question. They introduce the two main characteristics of options: desirability, which refers to long time wishes or intentions that are far away of their implementation of a decision option, and feasibility, which refers to a short term, close to the implementation characteristics. One can mention other works concerned with similar issues. It should be noted that this fact has already been reflected in (dynamic, or multistage) decision making and control models in which discounting is widely used.

In our context, we proposed(cf. Kacprzyk, Wilbik [17]) to take into account some of those psychological findings related to the importance of time by using different protoforms of linguistic summaries of times series, called *temporal linguistic summaries*. We consider two types of temporal protoforms: "E_T among all y's Q are P", exemplified by "Recently, among all segments, most are slowly increasing", and "E_T among all Ry's Q are P", exemplified by "Initially, among all short segments, most are quickly decreasing"; they both go beyond the classic Zadeh's protoforms.

The analysis of time series data involves different elements but we concentrate on the specifics of our approach. First, we have to to identify the consecutive parts of time series within which the data exhibit some uniformity as to their variability. Some variability must here be neglected, under an assumed granularity. Here, these consecutive parts of a time series are called trends (or segments), and described by straight line segments. That is, we perform first a piece-wise linear approximation of a time series and present time series data as a sequence of trends. The (linguistic) summaries of time series refer to the (linguistic) summaries of (partial) trends as meant above. For the construction of a piecewise linear approximation, we use a modified version of the Sklansky and Gonzalez algorithm (cf. [18]) though many other methods can be used cf. Keogh et al. [19], [20].

The next step is an aggregation of the (characteristic features of) consecutive trends over an entire time span (horizon) assumed. We follow the idea initiated by Yager [21], [22] and then shown more profoundly and in an implementable way in Kacprzyk and Yager [23], and Kacprzyk, Yager and Zadrożny [24], [25], that the most comprehensive and meaningful will be a linguistic quantifier driven aggregation resulting in linguistic summaries of classic protoform exemplified by "Most trends are short" or "Most long trends are increasing" and temporal protoform "Recently most trends are increasing" or "Recently most short trends are increasing".

These summaries are easily derived and interpreted using Zadehs fuzzy logic based calculus of linguistically quantified propositions. A new quality, and an increased generality was obtained by using Zadehs [26] protoforms as proposed by Kacprzyk and Zadrożny [27].

Here we employ the classic Zadehs fuzzy logic based calculus of linguistically quantified propositions in which the degree of truth (validity) is the most obvious and important quality criterion. Some other quality criteria like a degree of specificity, focus, fuzziness, etc. have also been proposed by Kacprzyk and Wilbik [28], [6], [5], [29]. The results obtain clearly indicate that multiple quality criteria of linguistic summaries of time series should be taken into account, and this makes the analysis obviously much more difficult.

As the first step towards an intended comprehensive multi-criteria assessment of linguistic summaries of time series, we propose here a very simple, effective and efficient approach, namely to use quite an old, maybe classic Yagers [30] proposal on an informativeness measure of a linguistic summary which combines, via an appropriate aggregation operator, the degree of truth, focus and specificity.

We illustrate our analysis on a linguistic summarization of daily quotations over an 8 year period of an investment (mutual) fund. We present the characteristic features of trends derived under some reasonable granulations, variability, trend duration, etc.

The paper is in line with some other modern approaches to linguistic summarization of time series. First, one should refer to the *SumTime* project coordinated by the University of Aberdeen, an EPSRC Funded Project for Generating Summaries of Time Series Data[1] in which English summary descriptions of a time series data set are sought by using advanced time series and NLG (natural language generation) technologies [31]. However, the linguistic descriptions obtained do not reflect an inherent imprecision (fuzziness) as in our approach. A relation between linguistic data summaries and NLG is discussed by Kacprzyk and Zadrożny [32], [33].

II. Linguistic Data Summaries

As a *linguistic summary of data (base)* we understand a (usually short) sentence (or a few sentences) that captures the very essence of the set of data, that is numeric, large, and because of its size is not comprehensible for human being.

We use Yager's basic approach [21]. A linguistic summary includes: (1) a summarizer P (e.g. *low* for attribute *salary*), (2) a quantity in agreement Q, i.e. a linguistic quantifier (e.g. *most*), (3) truth (validity) $\mathcal{T}$ of the summary and optionally, (4) a qualifier R (e.g. *young* for attribute *age*).

Thus, a linguistic summary may be exemplified by

$$\mathcal{T}(most \text{ of employees earn } low \text{ salary}) = 0.7 \qquad (1)$$

or in richer (extended) form, including a qualifier (e.g. *young*), by

$$\mathcal{T}(most \text{ of } young \text{ employees earn } low \text{ salary}) = 0.82 \quad (2)$$

Thus, basically the core of a linguistic summary is a linguistically quantified proposition in the sense of Zadeh [26] which may be written, respectively as

$$Qy's \text{ are } P \qquad\qquad QRy's \text{ are } P \qquad (3)$$

III. Linguistic Summaries of Trends

In our first approach we summarize the trends (segments) extracted from time series. Therefore as the first step we need to extract the segments. We assume that segment is represented by a fragment of straight line, because such segments are easy for interpretation.

There are many algorithms for the piecewise linear segmentation of time series data, including e.g. on-line (sliding window) algorithms, bottom-up or top-down strategies (cf. Keogh [19], [20]).

We consider the following three features of (global) trends in time series: (1) dynamics of change, (2) duration, and (3) variability. By *dynamics of change* we understand the speed of change of the consecutive values of time series. It may be described by the slope of a line representing the trend, represented by a linguistic variable. *Duration* is the length of a single trend, and is also represented by a linguistic variable. *Variability* describes how "spread out" a group of data is. We compute it as a weighted average of values taken by some measures used in statistics: (1) the range, (2) the interquartile range (IQR), (3) the variance, (4) the standard deviation, and (5) the mean absolute deviation (MAD). This is also treated as a linguistic variable.

For practical reasons for all we use a fuzzy granulation (cf. Bathyrshin at al. [34], [35]) to represent the values by a small set of linguistic labels as, e.g.: increasing, slowly increasing, constant, slowly decreasing, decreasing. These values are equated with fuzzy sets.

For clarity and convenience we employ Zadeh's [36] protoforms for dealing with linguistic summaries [27]. A protoform is defined as a more or less abstract prototype (template) of a linguistically quantified proposition. We have two types of protoforms of linguistic summaries of trends:
– a short form:

$$\text{Among all segments, } Q \text{ are } P \qquad (4)$$

e.g.: "Among all segments, *most* are *slowly increasing*".
– an extended form:

$$\text{Among all } R \text{ segments, } Q \text{ are } P \qquad (5)$$

e.g.: "Among all *short* segments, *most* are *slowly increasing*".

We can extend our protoforms given in (4) and (5) by adding a temporal expression, E_T, like: "recently", "in the very beginning" or "in May 2010", "initially", etc. (cf. Kacprzyk, Wilbik [17]). The temporal protoforms can have the following forms:

- a simple (short) protoform:

$$E_T \text{ among all segments, } Q \text{ are } P \qquad (6)$$

[1] cf. www.csd.abdn.ac.uk/research/sumtime/

e.g.: "*Recently*, among all segments, *most* are *slowly increasing*".

- an extended protoform:

$$E_T \text{ among all } R \text{ segments, } Q \text{ are } P \tag{7}$$

e.g.: "*Initially*, among all *short* segments, *most* are *slowly increasing*".

The quality of linguistic summaries can be evaluated in many different ways, eg. using the degree of truth, specificity, appropriateness or others.

Yager [30] proposed measure of informativeness, a measure that evaluates the amount of information hidden in the summary. This measure is interesting as it aggregates some of previously mentioned quality criteria, namely the truth value, degree of specificity and degree of focus in the case of extended form summaries. Now we will present shortly those 3 measures.

A. Truth Value

The truth value (a degree of truth or validity), introduced by Yager in [21], is the basic criterion describing the degree of truth (from $[0, 1]$) to which a linguistically quantified proposition equated with a linguistic summary is true.

Using Zadeh's calculus of linguistically quantified propositions [26] it is calculated in dynamic context using the same formulas as in the static case. Thus, the truth value is calculated for the simple and extended form as, respectively:

$$\mathcal{T}(\text{Among all } y\text{'s, } Q \text{ are } P) = \mu_Q\left(\frac{1}{n}\sum_{i=1}^{n}\mu_P(y_i)\right) \tag{8}$$

$$\begin{aligned}&\mathcal{T}(\text{Among all } Ry\text{'s, } Q \text{ are } P) =\\ &= \mu_Q\left(\frac{\sum_{i=1}^{n}\mu_R(y_i)\wedge\mu_P(y_i)}{\sum_{i=1}^{n}\mu_R(y_i)}\right)\end{aligned} \tag{9}$$

where μ_P, μ_R and μ_Q are membership functions of fuzzy set representing summarizer, qualifier and linguistic quantifier, respectively. $\wedge$ is the minimum operation (more generally it can be another appropriate operator, notably a t-norm). In Kacprzyk, Wilbik and Zadrożny [37] results obtained by using different t-norms were compared. Various t-norms can be in principle used in Zadeh's calculus but clearly their use may result in different results of the linguistic quantifier driven aggregation. It seems that the minimum operation is a good choice since it can be easily interpreted and the numerical values correspond to the intuition.

The computation of truth values of temporal summaries is very similar to the previous case. We only need to consider a temporal expression as an additional external qualifier, as the temporal expression limits the universe of interest to those trends (segments) only that occur on the time axis described by a fuzzy set modeling the expression E_T. We compute the proportion of segments in which "trend is P" and occurs in E_T to those that occur in E_T. Next, we compute the degree to which this proportion is Q.

The truth value of the simple temporal protoform (6) is computed as:

$$\begin{aligned}&\mathcal{T}(E_T \text{ among all } y\text{'s, } Q \text{ are } P) =\\ &= \mu_Q\left(\frac{\sum_{i=1}^{n}\mu_{E_T}(y_i)\wedge\mu_P(y_i)}{\sum_{i=1}^{n}\mu_{E_T}(y_i)}\right)\end{aligned} \tag{10}$$

where $\mu_{E_T}(y_i)$ is the degree to which a trend (segment) occurs during the time span described by E_T.

Similarly we compute the truth of the extended temporal protoform (7) as:

$$\begin{aligned}&\mathcal{T}(E_T\text{Among all } Ry\text{'s, } Q \text{ are } P) =\\ &= \mu_Q\left(\frac{\sum_{i=1}^{n}\mu_{E_T}(y_i)\wedge\mu_R(y_i)\wedge\mu_P(y_i)}{\sum_{i=1}^{n}\mu_{E_T}(y_i)\wedge\mu_R(y_i)}\right)\end{aligned} \tag{11}$$

A natural question emerges of how to compute $\mu_{E_T}(y_i)$. Let $\mu_{E_T}(t)$ be a membership function of a fuzzy set representing a linguistic variable E_T. We assume that the time span considered is normalized, i.e. $t \in [0, 1]$, the first observation is made for $t = 0$ and the last for $t = 1$. Let us consider a segment y_i, starting at time a and terminating at time b, $0 \leq a < b \leq 1$. Then

$$\mu_{E_T}(y_i) = \frac{1}{b-a}\int_a^b \mu_{E_T}(t)dt \tag{12}$$

and we can interpret this value as the average membership degree of E_T in $[a, b]$. Graphically it can be represented as the gray stripped area divided by the stripped area in Figure 1.

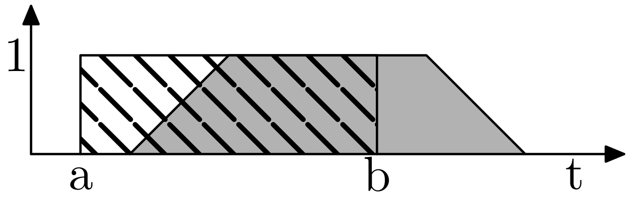


Fig. 1. Graphical presentation of $\mu_{E_T}(y_i)$

B. Degree of Specificity

The concept of specificity provides a measure of the amount of information contained in a fuzzy subset or possibility distribution. The specificity measure evaluates the degree to which a fuzzy subset points to one and only one element as its member [38].

We will consider the original Yagers proposal [38], in which specificity measures the degree to which a fuzzy subset contains one and only one element. The measure of specificity is a measure $Sp : I^X \longrightarrow I$, $I \in [0, 1]$ if it has the following properties: (1) $Sp(A) = 1$ if and only if $A = \{x\}$, (is a singleton set), (2) $Sp(\varnothing) = 0$, and (3) $\frac{\partial Sp(A)}{\partial a_1} > 0$ and $\frac{\partial Sp(A)}{\partial a_j} \leq 0$ for all $j \geq 2$.

In [39] Yager proposed a measure of specificity as

$$Sp(A) = \int_0^{\alpha_{max}} \frac{1}{card(A_\alpha)}d\alpha \tag{13}$$

where α_{max} is the largest membership grade in A, A_α is the α-level set of A, (i.e. $A_\alpha = \{x : \mu_A(x) \geq \alpha\}$) and $card(A_\alpha)$ is the number of elements in A_α.

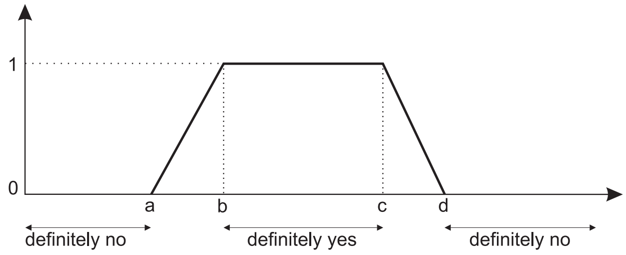


Fig. 2. A trapezoidal membership function of a set

In our summaries to define the membership functions of the linguistic values we use trapezoidal functions, as they are sufficient in most applications [40]. Moreover, they can be very easily interpreted and defined by a user not familiar with fuzzy sets and logic, as in Figure 2. To represent a fuzzy set with a trapezoidal membership function we need to store only four numbers, a, b, c and d. Usage such a definition of a fuzzy set is a compromise between cointension and computational complexity. In such a case measure of specificity of a fuzzy set A

$$Sp(A) = 1 - \frac{c + d - (a + b)}{2} \tag{14}$$

C. Degree of Focus

The very purpose of a degree of focus is to limit the search for the best linguistic summaries by taking into account some additional information in addition to truth values. The extended protoform linguistic summaries (5) does limit by itself the search space as the search is performed in a limited subspace of all (most) trends that fulfill an additional condition specified by qualifier R. The very essence of the degree of focus introduced in this paper is to give the proportion of trends satisfying property R to all trends extracted from the time series. It provides a measure that, in addition to the basic truth value, can help control the process of discarding non-promising linguistic summaries.

The *degree of focus* is similar in spirit to a degree of covering, described above, but it measures how many trends fulfill property R. The degree of focus makes obviously sense for the extended protoform summaries only, and is calculated as (cf. Kacprzyk and Wilbik [29]):

$$d_{foc}(\text{Among all } Ry\text{'s, } Q \text{ are } P) = \frac{1}{n}\sum_{i=1}^{n} \mu_R(y_i) \tag{15}$$

In our context, the degree of focus describes how many trends extracted from a given time series fulfill qualifier R in comparison to all extracted trends. If the degree of focus is high, then we can be sure that such a summary concerns many trends, so that it is more general. However, if the degree of focus is low, we may be sure that such a summary describes a (local) pattern seldom occurring.

The formula for the degree of focus for the extended temporal protoform requires small changes. The temporal expression may be treated as the external qualifier, and we can compute the proportion of trends satisfying property R in the E_T time span to all trends occurring in that time span. So the degree of focus of extended temporal protoform summaries (7) is computed as:

$$\begin{aligned} & d_{foc}(E_T \text{ among all } Ry\text{'s, } Q \text{ are } P) = \\ & = \frac{\sum_{i=1}^{n} \mu_{E_T}(y_i) \wedge \mu_R(y_i)}{\sum_{i=1}^{n} \mu_{E_T}(y_i)} \end{aligned} \tag{16}$$

Here also the degree of focus help us distinguish more general summaries from those describing a (local) pattern seldom occurring.

As we wish to discover a more general, global relationship, we can eliminate the linguistic summaries that concern a small number of trends only. The degree of focus may be used to eliminate the whole groups of extended form summaries for which qualifier R limits the set of possible trends to, for instance, 5%. Such summaries, although they may be very true, will not be representative.

We could think also about an additional measure similar to the degree of focus for the temporal protoforms – a degree of focus of temporal expression. This degree could measure how many trends extracted from a given time series occurs in the time span described by E_T in comparison to all extracted trends. Hence, for the simple and extended temporal protoform summaries we have:

$$d_{E_T}(E_T \text{ among all } Ry\text{'s, } Q \text{ are } P) = \frac{1}{n}\sum_{i=1}^{n} \mu_{E_T}(y_i) \tag{17}$$

D. Measure of Informativeness

The idea of a measure of informativeness (cf. Yager, Ford and Canas [30]) may be summarized as follows. Suppose we have a data set, whose elements are from measurement space X. One can say that the data set itself is its own most informative description, and any other summary implies a loss of information. So, a natural question is whether a particular summary is informative, and to what extent.

Yager et. al [30] proposed the following *measure of informativeness* of a simple protoform summary

$$\begin{aligned} & I(\text{Among all } y\text{'s } Q \text{ are } P) = \\ & = (\mathcal{T} \cdot Sp(Q) \cdot Sp(P)) \vee ((1 - \mathcal{T}) \cdot Sp(Q^c) \cdot Sp(P^c)) \end{aligned} \tag{18}$$

where P^c is the negation of P, i.e. $\mu_{P^c}(\cdot) = 1 - \mu_P(\cdot)$ and Q^c is the negation of Q, i.e. $\mu_{Q^c}(\cdot) = 1 - \mu_Q(\cdot)$. $Sp(Q)$ is specificity of Q, similarly it is calculated for Q^c, P and P^c.

For the extended protoform summary we propose the following measure (cf. Kacprzyk and Wilbik [41]):

$$\begin{aligned} I(\text{Among all } Ry\text{'s } Q \text{ are } P) &= \\ = \; & (\mathcal{T} \cdot Sp(Q) \cdot Sp(P) \cdot Sp(R) \cdot d_{foc}) \\ & \vee ((1 - \mathcal{T}) \cdot Sp(Q^c) \cdot Sp(P^c) \cdot Sp(R) \cdot d_{foc}) \end{aligned} \tag{19}$$

where d_{foc} is the degree of focus of the summary, $Sp(R)$ is specificity of qualifier R and the rest is defined as previously.

The measure of informativeness of the simple temporal protoform summary is calculated as:

$$\begin{aligned} &I(E_T \text{ among all } y\text{'s } Q \text{ are } P) = \\ &= \ (\mathcal{T} \cdot Sp(Q) \cdot Sp(P) \cdot Sp(E_T) \cdot d_{E_T}) \\ &\quad \vee ((1-\mathcal{T}) \cdot Sp(Q^c) \cdot Sp(P^c) \cdot Sp(E_T) \cdot d_{E_T}) \end{aligned} \quad (20)$$

where $Sp(E_T)$ is the specificity of the temporal expression and d_{E_T} is the degree of focus of temporal expression defined as in Eq. (17).

The measure of informativeness of the extended temporal protoform summary is calculated as:

$$\begin{aligned} &I(E_T \text{ among all } Ry\text{'s } Q \text{ are } P) = \\ &= \ (\mathcal{T} \cdot Sp(Q) \cdot Sp(P) \cdot Sp(E_T) \cdot Sp(R) \cdot d_{foc} \cdot d_{E_T}) \\ &\quad \vee ((1-\mathcal{T}) \cdot Sp(Q^c) \cdot Sp(P^c) \cdot Sp(E_T) \cdot Sp(R) \cdot \\ &\qquad \cdot d_{foc} \cdot d_{E_T}) \end{aligned} \quad (21)$$

Here in those formulas different values are aggregated by the product. We could think of using instead of the product other t-norms. However, for example, the minimum would ignore all values that are smaller than the largest one, and the Łukasiewicz t-norm tends to be very small if we aggregate many numbers. Moreover, the product may be a natural choice taking into account many results from, for instance, decision analysis and mathematical economics.

IV. Numerical Results

The method proposed was tested on data on quotations of an investment (mutual) fund that invests at least 50% of assets in shares listed at the Warsaw Stock Exchange.

Data shown in Figure 3 were collected from January 2002 until the December 2009 with the value of one share equal to PLN 12.06 in the beginning of the period to PLN 35.82 at the end of the time span considered (PLN stands for the Polish Zloty). The minimal value recorded was PLN 9.35 while the maximal one during this period was PLN 57.85. The biggest daily increase was equal to PLN 2.32, while the biggest daily decrease was equal to PLN 3.46. We illustrate the method proposed by analyzing the absolute performance of a given investment fund, and not against benchmarks, for illustrativeness.

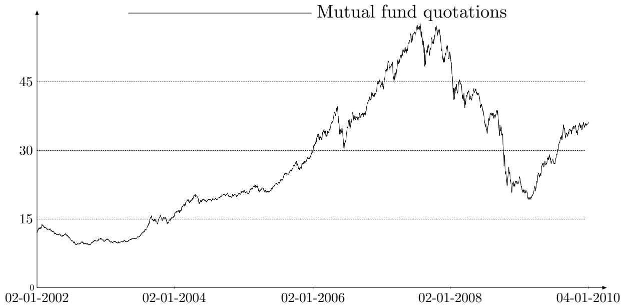


Fig. 3. Daily quotations of an investment fund in question

We obtain 362 extracted trends, with the shortest of 1 time unit only, and the longest – 71 time units. We assume 3 labels only for each attribute: short, medium and long for duration, increasing, constant and decreasing for dynamics and low, moderate and high for variability. The use of linguistic values in the summaries is clearly a reflection of a natural information granulation.

In Table I there are presented the most valid summaries of the classic protoforms. They are ordered according to the degree of truth and then the degree of focus.

TABLE I
Summaries of the classic protoform

linguistic summary	$\mathcal{T}$	d_{foc}	I
Among all y, most are short	1.0000	1.0000	0.4675
Among all moderate y, most are short	1.0000	0.3453	0.0969
Among all decreasing y, most are short	1.0000	0.2267	0.0604
Among all increasing y, most are short	1.0000	0.1688	0.0450
Among all medium y, most are constant	1.0000	0.1394	0.0429
Among all medium y, most are constant and high	1.0000	0.1394	0.0778
Among all medium y, most are high	1.0000	0.1394	0.0349
Among all medium and constant y, most are high	1.0000	0.1366	0.0794
Among all medium and high y, most are constant	1.0000	0.1222	0.0780
Among all high y, most are short	0.8453	0.5789	0.1601
Among all constant y, most are short	0.8341	0.6045	0.2027
Among all high y, most are constant	0.8164	0.5789	0.1565
Among all constant y, most are high	0.7564	0.6045	0.1514

We may notice that the first summary has a very big value of the measure of the informativeness, and this summary is of a simple protoform. It is very informative. Also the last four summaries presented in Table I are interesting, as their values of this measure are quite high. Those summaries do not have the truth value equal to 1, but nevertheless they are also true, moreover they have very high values of degree of focus, indicating that these summaries describe pattern which are quite often occuring.

In Table II we may see the temporal summaries decribing that time series. Me may notice the summaries of the first few years after the the fund wa established (initially), then the middle time (in the middle), and the last two summaries describe more or less ime qotations from autumn 2007, when the finantial crisis started.

The obtained summaries are divide into 2 groups, each describing separate period. The first group describes what was happening initially. First 4 summaries have high values of the measure of informativeness. Especially interesting is the summary "initially among all y, most are constant and very high", it has the value of this measure higher than a summary with only one of the linguistic values used in the summary.

In the second group, only one summary stand out –"middle among all y, most are constant", with a value of the measure of informativeness over twice as big than the other values. We may notice that this value is also the biggest for all summaries presented in Table II. It is partially so, because the temporal expression "in the middle" describes the longest period.

In the last group describing the last 2 years – the time of financial crisis, we obtained just 2 summaries, from which the

TABLE II
SUMMARIES OF THE TEMPORAL PROTOFORM

linguistic summary	$\mathcal{T}$	d_{foc}	I
initially among all constant y, most are very high	1.0000	1.0000	0.0329
initially among all y, most are constant	1.0000	1.0000	0.0387
initially among all y, most are constant and very high	1.0000	1.0000	0.0770
initially among all y, most are very high	1.0000	1.0000	0.0383
initially among all very high y, most are constant	1.0000	0.8089	0.0266
initially among all long and constant y, most are very high	1.0000	0.3566	0.0213
initially among all long y, most are constant	1.0000	0.3566	0.0097
initially among all long y, most are constant and very high	1.0000	0.3566	0.0192
initially among all long y, most are very high	1.0000	0.3566	0.0096
initially among all medium and constant y, most are very high	1.0000	0.3292	0.0214
initially among all medium and very high y, most are constant	1.0000	0.3292	0.0215
initially among all medium y, most are constant	1.0000	0.3292	0.0107
initially among all medium y, most are constant and very high	1.0000	0.3292	0.0212
initially among all medium y, most are very high	1.0000	0.3292	0.0106
initially among all very long y, most are constant	1.0000	0.3230	0.0069
initially among all long and very high y, most are constant	1.0000	0.2947	0.0177
initially among all very long and very high y, most are constant	1.0000	0.1939	0.0105
initially among all high y, most are constant	1.0000	0.1911	0.0059
initially among all very long and high y, most are constant	1.0000	0.1291	0.0068
initially among all high y, most are very long	0.7518	0.1911	0.0028
initially among all high y, most are very long and constant	0.7518	0.1911	0.0073
middle among all very high y, most are constant	1.0000	0.4860	0.0481
middle among all medium y, most are constant	1.0000	0.3205	0.0312
middle among all high y, most are short	1.0000	0.2436	0.0249
middle among all medium and very high y, most are constant	1.0000	0.2422	0.0476
middle among all short and very high y, most are constant	1.0000	0.1094	0.0228
middle among all y, most are constant	0.9659	1.0	0.1124
middle among all medium y, most are very high	0.9113	0.3205	0.0281
middle among all short y, most are constant	0.8506	0.4794	0.0447
middle among all medium and constant y, most are very high	0.8484	0.2840	0.0470
middle among all moderate y, most are short	0.8188	0.2741	0.0200
middle among all short and moderate y, most are constant	0.8084	0.1945	0.0301
middle among all moderate y, most are constant	0.8010	0.2741	0.0179
from the crisis begin among all medium y, most are constant	1.0000	0.2273	0.0513
from the crisis begin among all decreasing y, most are very short	0.9887	0.1438	0.0294

first one is more informative than the other, nevertheless both seems to be interesting for the experts.

V. CONCLUDING REMARKS

We extended our approach to the linguistic summarization of time series towards a multicriteria analysis of classic and temporal summaries by assuming as a quality criterion Yager's measure of informativeness that combines in a natural way the measures of truth, focus and specificity. Results on the summarization of quotations of an investment (mutual) fund are very encouraging.

REFERENCES

[1] J. Kacprzyk, A. Wilbik, and S. Zadrożny, "Linguistic summarization of trends: a fuzzy logic based approach," in *Proceedings of the 11th International Conference Information Processing and Management of Uncertainty in Knowledge-based Systems*, 2006, pp. 2166–2172.

[2] ——, "On some types of linguistic summaries of time series," in *Proceedings of the 3rd International IEEE Conference "Intelligent Systems"*. IEEE Press, 2006, pp. 373–378.

[3] ——, "Linguistic summarization of time series using a fuzzy quantifier driven aggregation," *Fuzzy Sets and Systems*, vol. 159, no. 12, pp. 1485–1499, 2008.

[4] J. Kacprzyk and A. Wilbik, "An extended, specificity based approach to linguistic summarization of time series," in *Proceedings of the 12th International Conference Information Processing and Management of Uncertainty in Knowledge-based Systems*, 2008, pp. 551–559.

[5] ——, "A new insight into the linguistic summarization of time series via a degree of support: Elimination of infrequent patterns," in *Soft Methods for Handling Variability and Imprecision*, D. Dubois, M. Lubiano, H. Prade, M. A. Gil, P. Grzegorzewski, and O. Hryniewicz, Eds. Springer-Verlag, Berlin and Heidelberg, 2008, pp. 393–400.

[6] ——, "Linguistic summarization of time series using linguistic quantifiers: augmenting the analysis by a degree of fuzziness," in *Proceedings of 2008 IEEE World Congress on Computational Intelligence*. IEEE Press, 2008, pp. 1146–1153.

[7] Past performance does not predict future performance."

[8] ——, "www.personalfn.com/detail.asp?date=9/1/2007&story=3, Past performance is not everything."

[9] ——, "stockcasting.blogspot.com/2005/12/new-years-evepast-performance-is-no.html, new year's eve:past performance is no indication of future return."

[10] R. Myers, "Using past performance to pick mutual funds," *Nation's Business*, 1997.

[11] J. C. Bogle, *Common Sense on Mutual Funds: New Imperatives for the Intelligent Investor*. New York: Wiley, 1999.

[12] investing: Look at more than a fund's past performance, U.S. Securities and Exchange Commission."

[13] ——, "www.personalfn.com/detail.asp?date=5/18/2007&story=2, A 10-step guide to evaluating mutual funds."

[14] D. Ariely and D. Zakay, "A timely account of the role of duration in decision making," *Acta Psychologica*, vol. 108, no. 2, pp. 187–207, 2001.

[15] H. Rachlin, *Judgment, Decision, and Choice: A Cognitive/Behavioral Synthesis*. W.H. Freeman & Company, 1989.

[16] N. Liberman and Y. Trope, "The role of feasibility and desirability considerations in near and distant future decisions: A test of temporal construal theory," *Journal of Personality and Social Psychology*, vol. 75, pp. 5–18, 1998.

[17] J. Kacprzyk and A. Wilbik, "Temporal linguistic summaries of time series using fuzzy logic," in *Proceedings of IPMU2010 (in press)*, 2010.

[18] J. Sklansky and V. Gonzalez, "Fast polygonal approximation of digitized curves," *Pattern Recognition*, vol. 12, no. 5, pp. 327–331, 1980.

[19] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "An online algorithm for segmenting time series," in *Proceedings of the 2001 IEEE International Conference on Data Mining*, 2001.

[20] ——, "Segmenting time series: A survey and novel approach," in *Data Mining in Time Series Databases*, M. Last, A. Kandel, and H. Bunke, Eds. World Scientific Publishing, 2004.

[21] R. R. Yager, “A new approach to the summarization of data,” *Information Sciences*, vol. 28, pp. 69–86, 1982.

[22] ——, “On linguistic summaries in data,” in *Knowledge Discovery in Databases*, G. Piatetsky-Shapiro and W. J. Frawley, Eds. MIT Press, Cambridge, USA, 1991, pp. 347–363.

[23] J. Kacprzyk and R. R. Yager, “Linguistic summaries of data using fuzzy logic,” *International Journal of General Systems*, vol. 30, pp. 33–154, 2001.

[24] J. Kacprzyk, R. R. Yager, and S. Zadrożny, “A fuzzy logic based approach to linguistic summaries of databases,” *International Journal of Applied Mathematics and Computer Science*, vol. 10, pp. 813–834, 2000.

[25] ——, “Fuzzy linguistic summaries of databases for an efficient business data analysis and decision support,” in *Knowledge Discovery for Business Information Systems*, J. Z. W. Abramowicz, Ed. Boston: Kluwer, 2001, pp. 129–152.

[26] L. A. Zadeh, “Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic,” *Fuzzy Sets and Systems*, vol. 9, no. 2, pp. 111–127, 1983.

[27] J. Kacprzyk and S. Zadrożny, “Linguistic database summaries and their protoforms: toward natural language based knowledge discovery tools,” *Information Sciences*, vol. 173, pp. 281–304, 2005.

[28] J. Kacprzyk and A. Wilbik, “Linguistic summarization of time series using fuzzy logic with linguistic quantifiers: a truth and specificity based approach,” in *Artificial Intelligence and Soft Computing ICAISC 2008*, L. Rutkowski, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, Eds. Springer-Verlag, Berlin and Heidelberg, 2008, pp. 241–252.

[29] ——, “Towards an efficient generation of linguistic summaries of time series using a degree of focus,” in *Proceedings of the 28th North American Fuzzy Information Processing Society Annual Conference – NAFIPS 2009*, 2009.

[30] R. R. Yager, K. M. Ford, and A. J. Cañas, “An approach to the linguistic summarization of data,” in *Uncertainty in Knowledge Bases, 3rd International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU ’90, Paris, France, July 2-6, 1990, Proceedings*, B. Bouchon-Meunier, R. R. Yager, and L. A. Zadeh, Eds. Springer, 1990, pp. 456–468.

[31] S. G. Sripada, E. Reiter, and I. Davy, “SumTime-Mousam: Configurable marine weather forecast generator,” *Expert Update*, vol. 6, no. 3, pp. 4–10, 2003.

[32] J. Kacprzyk and S. Zadrożny, “Data mining via protoform based linguistic summaries: Some possible relations to natural language generation,” in *2009 IEEE Symposium Series on Computational Intelligence Proceedings*, Nashville, TN, 2009, pp. 217–224.

[33] ——, “Computing with words is an implementable paradigm: fuzzy queries, linguistic data summaries and natural language generation,” *IEEE Transactions on Fuzzy Systems*, 2010, (forthcoming).

[34] I. Batyrshin, “On granular derivatives and the solution of a granular initial value problem,” *International Journal Applied Mathematics and Computer Science*, vol. 12, no. 3, pp. 403–410, 2002.

[35] I. Batyrshin and L. Sheremetov, “Perception based functions in qualitative forecasting,” in *Perception-based Data Mining and Decision Making in Economics and Finance*, I. Batyrshin, J. Kacprzyk, L. Sheremetov, and L. A. Zadeh, Eds. Springer-Verlag, Berlin and Heidelberg, 2006.

[36] L. A. Zadeh, “A prototype-centered approach to adding deduction capabilities to search engines – the concept of a protoform,” in *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS 2002)*, 2002, pp. 523–525.

[37] J. Kacprzyk, A. Wilbik, and S. Zadrożny, “Linguistic summarization of time series under different granulation of describing features,” in *Rough Sets and Intelligent Systems Paradigms - RSEISP 2007*, M. Kryszkiewicz, J. F. Peters, H. Rybinski, and A. Skowron, Eds. Springer-Verlag, Berlin and Heidelberg, 2007, pp. 230–240.

[38] R. R. Yager, “On measures of specificity,” in *Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration with Applications*, O. Kaynak, L. A. Zadeh, B. Türksen, and I. J. Rudas, Eds. Springer-Verlag: Berlin, 1998, pp. 94–113.

[39] ——, “Measuring tranquility and anxiety in decision making: An application of fuzzy sets,” *International Journal of General Systems*, vol. 8, pp. 139–146, 1982.

[40] L. A. Zadeh, “Computation with imprecise probabilities,” in *Proceedings of the 12th International Conference Information Processing and Management of Uncertainty in Knowledge-based Systems*, 2008.

[41] J. Kacprzyk and A. Wilbik, “A multi-criteria evaluation of linguistic summaries of time series via a measure of informativeness,” in *Proceedings of ICAISC2010 (in press)*, 2010.

Workshop on Agent Based Computing: from Model to Implementation VII

The 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 ad-on components, further increasing value of existing solutions.

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 logics
- 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

Paolo Bresciani, European Commission – DG Information Society and Media, Belgium

Zoran Budimac, University of Novi Sad, Republic of Serbia

Giacomo Cabri, University of Modena and Reggio Emilia, Italy

Bengt Carlsson, Blekinge Institute of Technology, Sweden

Radovan Cervenka, Whitestein Technologies, Slovak Republic

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 Universita' di Napoli, Italy

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 Politécnica de Valencia, Spain

Dominic Greenwood, Whitestein Technologies, Switzerland

Maurice Grinberg, New Bulgarian University, Bulgaria

Andreas Herzig, IRIT, France

Mike Hinchey, Lero-the Irish Software Engineering Research Centre, Ireland

Piotr Jedrzejowicz, Gdynia Maritime University, Poland

Gordan Jezic, University of Zagreb, Croatia/Hrvatska

Tomas Klos, Delft University of Technology, Netherlands

Matthias Klusch, German Research Center for Artificial Intelligence, Germany

Igor Kotenko, SPIIRAS, Russian Federation

Zofia Kruczkiewicz, Technical University of Wroclaw, Poland

Michal Laclavik, Institute of Informatics, Slovak Academy of Sciences, Slovak Republic

Jiming Liu, Hong Kong Baptist University, China

Vincenzo Loia, University of Salermo, Italy

Michele Loreti, University of Florence, Italy

Giuseppe Mangioni, University of Catania, Italy

Viorel Negru, West University of Timisoara, Romania

Andrea Omicini, Alma Mater Studiorum, Università di Bologna, Italy

Nir Oren, King's College London, United Kingdom

Sascha Ossowski, University Rey Juan Carlos, Spain

Joaquin Peña, University of Seville, Spain

Agostino Poggi, DII – University of Parma, Italy

Alexander Pokahr, University of Hamburg, Germany

Thomas Potok, Oak Ridge National Lab, Afghanistan

Jarogniew Rykowski, Poznan University of Economics, Poland

Marija Slavkovik, 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

Paolo Torroni, University of Bologna, Italy

Walter Truszkowski, NASA, USA

Rainer Unland, University of Duisburg-Essen, ICB, Germany

Andrzej Uszok, IHMC, USA

Salvatore Venticinque, Second University of Naples, Italy

Paulus Wahjudi, Marshall University, USA

Yu Zhang, Trinity University, USA

Antoine Zimmermann, Digital Enterprise Research Institute, Ireland

Organizing (Steering) Committee

Costin Badica, University of Craiova, Romania

Maria Ganzha, University of Gdansk and IBS PAN, Poland

Marcin Paprzycki (Chairman), WSM and IBS PAN, Poland

Shahram Rahimi, Southern Illinois University, USA

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 165–172

ISBN 978-83-60810-27-9
ISSN 1896-7094

Java-based Mobile Agent Platforms for Wireless Sensor Networks

Francesco Aiello, Alessio Carbone, Giancarlo Fortino*, Stefano Galzarano
DEIS, Università della Calabria, Via P.Bucci cubo 41c, 87036 Rende (CS), Italy
Email: {faiello, acarbone, galzarano}@si.deis.unical.it, g.fortino@unical.it

***Abstract*—This paper proposes an overview and comparison of mobile agent platforms for the development of wireless sensor network applications. In particular, the architecture, programming model and basic performance of two Java-based agent platforms, Mobile Agent Platform for Sun SPOT (MAPS) and Agent Factory Micro Edition (AFME), are discussed and evaluated. Finally, a simple yet effective case study concerning a mobile agent-based monitoring system for remote sensing and aggregation is proposed. The proposed case study is developed both in MAPS and AFME so allowing to analyze the differences of their programming models.**

***Keywords*: Mobile agent platforms, wireless sensor networks, Java Sun SPOT, finite state machines, intentional agents.**

I. Introduction

Due to recent advances in electronics and communication technologies, Wireless Sensor Networks (WSNs) are currently emerging as one of the most disruptive technologies enabling and supporting next generation ubiquitous and pervasive computing scenarios. A WSN is a network of RF transceivers, sensors, machine controllers, microcontrollers, and user interface devices with at least two nodes communicating by means of wireless transmissions. WSNs have a high potential to support a variety of high-impact applications such as disaster/crime prevention and military applications, environmental applications, health care applications, and smart spaces. However programming WSNs is a complex task due to the limited capabilities (processing, memory and transmission range) and energy resources of each sensor node as well as the lack of reliability of the radio channel. Moreover, WSN programming is usually application-specific (or more generally domain-specific) and requires tradeoffs in terms of task complexity, resource usage, and communication patterns. Therefore the developed software which usually integrates routing mechanisms, time synchronization, node localization and data aggregation is tightly dependent on the specific application and scarcely reusable. Thus to support rapid development and deployment of WSN applications flexible, WSN-aware programming paradigms are needed which directly provide proactive and on-demand code deployment at run-time as well as ease software programming at application, middleware and network layer. Among the programming paradigms proposed for the development of WSN applications [1], the mobile agent-based paradigm [2], which has already demonstrated its effectiveness in conventional distributed systems as well as in highly dynamic distributed environments, can effectively deal with the programming issues that WSNs have posed. In particular, a mobile agent is a software entity encapsulating dynamic behavior and able to migrate from one computing node to another to fulfill distributed tasks. We believe that mobile agents can provide more benefits in the context of WSNs than in conventional distributed environments. In particular, mobile agents can support the programming of WSNs at application, middleware and network levels.

In this paper we present the currently available mobile agent platforms for WSNs which are based either on TinyOS [3] or on Java Sun SPOT [4]. In particular, we focus on MAPS and AFME, the only two available Java-based platforms, by analyzing their architecture, programming model and core performances. Finally they are also exemplified through an effective case study showing the different programming mechanisms of the two platforms.

The rest of this paper is organized as follows. Section II introduces mobile agents, presenting their characteristics and benefits by focusing on the WSN context; moreover, work related to currently available mobile agent systems for WSNs is discussed. Section III describes two Java-based agent platforms (MAPS and AFME) which are also compared with respect to their architecture, programming model and performance. Section IV presents an example application for showing how differently agent behavior can be defined through MAPS and AFME. Finally, conclusions are drawn and on-going research efforts delineated.

II. Mobile Agents for Wireless Sensor Networks

In the context of distributed computing systems and highly dynamic distributed environments, mobile agents are a suitable and effective computing paradigm for supporting the development of distributed applications, services, and protocols [1]. Mobile agents are software processes able to migrate among computing nodes by retaining their execution state (strong mobility). In their seminal paper [2], Lange and Oshima advertised at least seven good reasons for using mobile agents in generic distributed systems. In the following we discuss them by focusing on the WSN context.

1. Network load reduction. Mobile agents are able to access remote resources, as well as communicate with any remote entity, by directly moving to their physical locations and interacting to them locally so that to save bandwidth resources. A mobile agent incorporating data processing capabilities can migrate to a sensor node, perform the needed operations on the sensed data and transmit the results to a sink

[*] Corresponding author

node. This is more desirable, rather than a periodic transmission of raw sensed data from the sensor node to the sink node and the computation of data processing on the latter.

2. Network latency overcoming. An agent provided with proper control logic may move to a sensor/actuator node to locally perform the required control tasks. This overcomes the network latency which will not affect the real-time control operations also in case of lack of network connectivity with the base station.

3. Protocol encapsulation. Suppose that a specific routing protocol supporting multi-hop paths should be deployed in a given zone of a WSN. A set of cooperating mobile agents encapsulating the routing protocol can be dynamically created and distributed into the proper sensor nodes without any regard for standardization matter. Also in case of protocol upgrading, a new set of mobile agents can easily replace the old one at run-time.

4. Asynchronous and autonomous execution. These are distinctive properties of mobile agents and very important in dynamic environments like WSNs where connections may not be stable and network topology may change rapidly. A mobile agent, upon a request, can autonomously travel across the network to gather needed information "node by node" or to carry out the programmed tasks and, finally, can asynchronously report the results to the requester.

5. Dynamic adaptation. Mobile agents can perceive their execution environment and react autonomously to changes. This behavioral dynamic adaptation is well suited for operating on long-running systems like WSNs where environment conditions are very likely to change over time.

6. Orientation to heterogeneity. Mobile agents can act as wrappers among systems based on different hardware and software. This ability can well fit the need for integrating heterogeneous WSNs supporting different sensor platforms or connecting WSN and other networks (like IP-based networks). An agent may be able to translate requests coming from a system into specific suitable requests to submit to another different system.

7. Robustness and fault-tolerance. The ability of mobile agents to dynamically react to unfavorable situations and events (e.g. low battery level) can lead to a better robust and fault tolerant distributed systems (e.g. by migrating all executing agents to an equivalent sensor node so that to continue their operations).

Mobile agents are supported by mobile agent systems (MASs) which basically provide an API for developing agent-based applications, and an agent server able to execute agents by providing them with basic services such as migration, communication, and node resource access. Developing flexible and efficient MASs for WSNs is a challenging and very complex task due to the currently available resource-constrained sensor nodes and related operating systems. Very few MASs for WSNs have been to date proposed and actually implemented. In the following, we discuss Agilla and actorNet, the most significant available research prototypes based on TinyOS

Agilla [5] is an agent-based middleware developed on TinyOS [3] and supporting multiple agents on each node. As shown by its software architecture (see Fig. 1), Agilla provides two fundamental resources on each node: a tuplespace and a neighbors list. The tuplespace represents a shared memory space where structured data (tuples) can be stored and retrieved, allowing agents to exchange information in a decoupling way. A tuplespace can be also accessed remotely. The neighbors list contains the address of all one-hop nodes, needed when an agent has to migrate.

Agents can migrate carrying their code and state, but do not carry their tuples locally stored on a tuplespace. Packets used for nodes communication (e.g. for agent migration/cloning, remote tuples accessing) are very small to minimize messages loss, whereas retransmission techniques are also adopted.

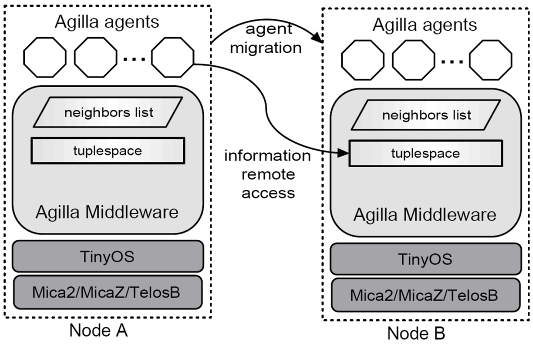


Fig. 1 Agilla software architecture

ActorNet [6] is an agent-based platform specifically designed for Mica2/TinyOS sensor nodes. To overcome the difficulties in allowing code migration and interoperability due to the strict coupling between applications and sensor node architectures, actorNet exposes services like virtual memory, context switching, and multi-tasking. Thanks to these features, it effectively supports agents programming by providing a uniform computing environment for all agents, regardless of hardware or operating system differences. The actorNet architecture is depicted in Fig. 2. The actorNet language used for high-level agent programming, has a syntax and a semantic similar to those of Scheme [7] with proper instructions extension.

Both Agilla and actorNet are designed for TinyOS which is based on the nesC language that is not an object-oriented language but an event- and component-based extension of C. The Java language, through which Sun SPOT [4] and Sentilla JCreate [18] sensors can be programmed, due to its object-oriented features, could provide more flexibility and extendibility for an effective implementation of agent-based platforms. The most significant available Java-based agent platforms for WSNs are MAPS [8, 9, 10] and AFME [11, 12, 13], which are discussed in Section III.

In Table 1, a comparison among the aforementioned agent platform with respect to 7 characteristics (migration, multitasking, communication model, programming language, remote configuration, intentional agents, and sensor platforms) is reported.

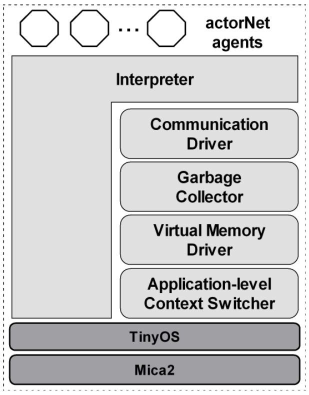


Fig. 2 ActorNet software architecture

TABLE 1
COMPARISON AMONG DIFFERENT WSN MASS

	Agilla	*actorNet*	*MAPS*	*AFME*
Migration	Y	Y	Y	Y
Multitasking	Y	Y	Y	Y
Communication Model	tuple space	messages	messages	messages
Programming Language	proprietary ISA	Scheme-like	Java	Java
Agent Model	Assembler-like	Functional	Finite State Machine	BDI
Intentional Agents	N	N	N	Y
Sensor Platforms	Mica2, MicaZ, TelosB	Mica2	Sun SPOT	Sun SPOT

Agent migration and multitasking, which allows for the execution of multiple agents on the same node, is supported by all the systems. The communication model of Agilla is centered on local tuplespace where agent can asynchronously insert tuples and take tuples left by other agents. Conversely the communication model of the other systems is based on (unicast and broadcast) message-passing. The programming language and model is different among the systems. Agilla is based on a proprietary low-level language composed of an assembler-like instruction set which makes programming of complex agents very difficult. ActorNet is based on a functional Scheme-like language whereas MAPS and AFME on the Java language. Indeed, MAPS uses a finite state machine model to define agent behaviour whereas AFME employs a more complex BDI-like model. Intentional agents are therefore only offered by AFME. Agilla and actorNet run on motes; in particular Agilla on Mica2, MicaZ, and TelosB, whereas actorNet currently only on Mica2. On the contrary MAPS and AFME are based on Sun SPOTs.

III. JAVA-BASED MOBILE AGENT PLATFORMS FOR WSNS

A great variety of Java-based agent platforms have been to date developed for the standard Java virtual machine atop conventional distributed systems. However, in the context of wireless sensor networks, only two Java-based platforms are currently available: MAPS (Mobile Agent Platform for Sun SPOT) and AFME (Agent Factory Micro Edition). While MAPS was specifically conceived for Sun SPOTs, AFME was developed for J2ME enabled PDAs and then ported onto Sun SPOTs. In the following we describe and compare MAPS and AFME with respect to their architecture, programming model and performance.

A. Mobile Agent Platform for Sun SPOTs

MAPS [8, 9, 10] is an innovative Java-based framework expressly developed on Sun SPOT technology for enabling agent-oriented programming of WSN applications. It has been defined according to the following requirement:

- 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) agent/s.
- Use of Java language for defining the mobile agent behavior.

MAPS architecture (see Fig. 3) is based on components which interact through events and offer a set of services to mobile agents, including message transmission, agent creation, agent cloning, agent migration, timer handling, and easy access to the sensor node resources. In particular, the main components are:

- *Mobile Agent (MA)*. MAs are the basic high-level component defined by user for constituting 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 [4]. The MAs hibernation and serialization involve data and execution state whereas the code should 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.

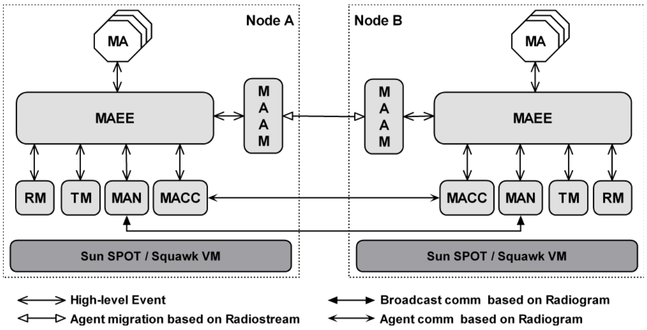


Fig. 3 MAPS software architecture

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 based on the 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.

B. Agent Factory Micro Edition

AFME [11, 12, 13] is an open-source lightweight J2ME MIDP compliant agent platform based upon the preexisting Agent Factory framework [14] and intended for wireless pervasive systems. Thus, AFME has not been specifically designed for sensor networks but, thanks to a recent support of J2ME onto the Sun SPOT sensor platform, it can be adopted for developing agent-based WSN applications.

AFME is strongly based on the *Believe-Desire-Intention* (*BDI*) paradigm [15], in which agents follow a sense-deliberate-act cycle. To facilitate the creation of BDI agents the framework supports a number of system components which developers have to extend when building their applications: perceptors, actuators, modules, and services. Perceptors and actuators enable agents to sense and to act upon their environment respectively. Modules represent a shared information space between actuators and perceptors of the same agent, and are used, for example, when a perceptor may perceive the resultant effect of an actuator affecting the state of an object instance internal to the agent. Services are shared information space between agents used for data agent exchange.

The agents are periodically executed using a scheduler, and four functions are performed when an agent is executed. First, the perceptors are fired and their sensing operations generate beliefs, which are added to the agent's belief set. A belief is a symbolic representation of information related to the agent's state or to the environment. Second, the agent's desires are identified using resolution-based reasoning, a goal-based querying mechanism commonly employed within Prolog interpreters. Third, the agent's commitments (a subset of desires) are identified using a knapsack procedure. Fourth, depending on the nature of the commitments adopted, various actuators are fired.

In AFME agents are defined through a mixed declarative/imperative programming model. The declarative Agent Factory Agent Programming Language (AFAPL), based on a logical formalism of belief and commitment, is used to encode an agent's behavior by specifying rules defining the conditions under which commitments are adopted. The imperative Java code is instead used to encode perceptors and actuators. A declarative rule is expressed in the following form:

$$b1, b2, \ldots, bn > doX;$$

where *b1... bn* represent beliefs, whereas *doX* is an action. The rule is evaluated during the agent execution, and if all the specified beliefs are currently included into the agent's beliefs set, the imperative code enclosed into the actuator associated to the symbolic string *doX* is executed.

The AFME platform architecture is shown in Fig. 4. It comprises a scheduler, a group of agents, and several platform services needed for supporting, among the others, agents communication and migration.

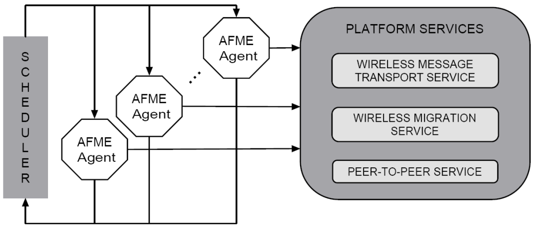


Fig. 4 AFME platform architecture

To improve reuse and modularity within AFME, actuators, perceptors, and services are prevented from containing direct object references to each other. Actuators and perceptors developed for interacting with a platform service in one application can be used, without any changes to their imperative code, to interact with a different service in a different application. In the other way round, the implementation of platform services can be completely modified without having to modify the actuators and the perceptors. Additionally, the same platform service may be used within two different applications to interact with a different set of actuators and perceptors. So, all system components of the AFME platform are interchangeable because they interact without directly referencing one another.

C. MAPS vs. AFME: A comparison

Both MAPS and AFME offer similar services for developing WSN agent-based application. Nevertheless, the definition of agents is based on different approaches. MAPS uses state machines to model the agent behavior and directly the Java language to program guards and actions. AFME uses a more complex model centered on perceptors, actuators, rules, modules, and services that define the agent behavior. They are both effective in modeling agent behavior even though MAPS is more straightforward as it relies on a programming style based on state machines widely known by programmers of embedded systems. Moreover, differently from AFME, MAPS is specifically designed for WSNs and fully exploits the release 5.0 red of the Sun SPOT library to provide advanced functionality of communication, migration, sensing/actuation, timing, and flash memory storage. An example is represented by the implementation of mobile agents through isolates, whose migration mechanism is directly offered by the SPOT Squawk JVM. Isolates are not used in AFME due to its employment as a more generic agent platform for CLDC-compliant devices. Apart from the agent implementation mechanism adopted, both platforms suffers from the current limitation of the Sun SPOTs which do not allow dynamic class loading, preventing from the possibility to support code migration (i.e. any classes required by the agent must already be present at the destination). Finally, MAPS allows developers to program agent-based applications in Java according to its rules so no translator and/or interpreter need to be developed and no new language has to be learnt.

To evaluate and compare the performance of MAPS and AFME two benchmarks have been defined according to [16] for the following mechanisms:

Agent communication. The agent communication time is computed for two agents running onto different nodes and communicating in a client/server fashion (request/reply). Two different request/reply schemes are used: (i) *data Back and Forward (B&F)*, in which both request and reply contain the same amount of data; (ii) *data B*, in which only the reply contains data. Comparison results are shown in Fig. 5. For agents with light data payload AFME performs better than MAPS; however, when the agent data payload overtakes 700 bytes MAPS starts performing better in the case *data B&F*.

Agent migration. The agent migration time is calculated for agent ping-pong among two single-hop-distant sensor nodes. Migration times are computed by varying the data cargo of the ping-pong agent. The obtained migration times are high due to the slowness of the SquawkVM operations supporting the migration process. Comparison results are shown in Fig. 6. AFME retains a higher performance migration mechanism.

IV. An Agent-Based Application Example

To demonstrate the effectiveness of agent-based platforms to support programming of WSN applications, a simple and exemplificative remote monitoring application has been developed. In particular, this section will show how differently the two Java-based platform MAPS and AFME allow defining the agent behavior. The proposed application example involves two sensor nodes and consists in the following three interacting agents:

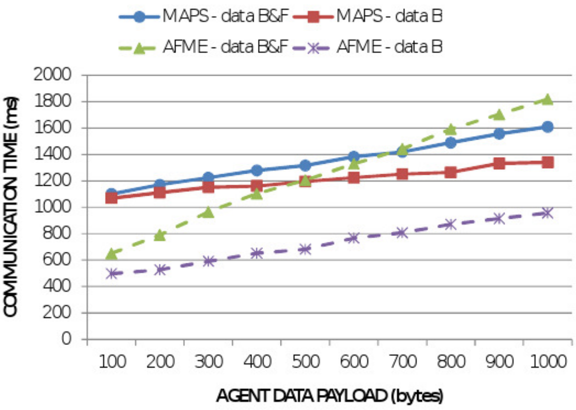


Fig. 5 Agent communication time comparison

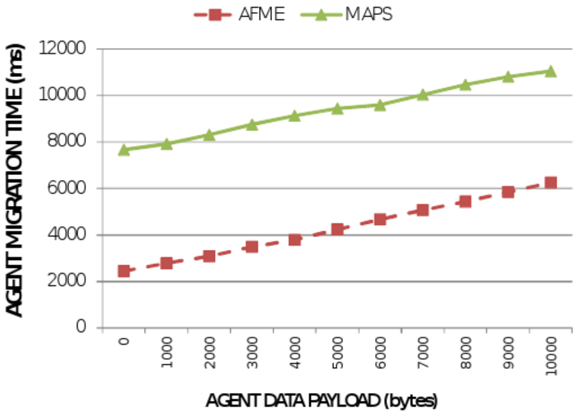


Fig. 6 Agent migration time comparison

- *DataCollectorAgent*, which collects data related to the Sun SPOT node sensors (accelerometer, temperature, light);
- *DataMessengerAgent*, which carries collected sensed data from the sensing node to the basestation;
- *DataViewerAgent*, which displays the received collected data.

The sequence of interactions among the three defined agents is shown in Fig. 7 through an M-UML sequence diagram [17]. The application execution is driven by the user by pressing a switch on the Sun SPOT on which the DataViewerAgent is running. Upon the user event the DataViewerAgent sends a remote message to the DataCollectorAgent (running on the other node) for starting its sensing operations. The agent therefore starts an internal timer to a particular value to begin its collecting activity: on timer expiration the agent acquires data from the onboard node sensors and collects them. As soon as the agent has acquired *numData* samples, it calculates a set of features (e.g. max, min and mean) for each of the sensor data types. Afterwards, the DataCollectorAgent creates the DataMessengerAgent which, carrying the computed features migrates to the DataViewerAgent node for data visualization. In case the user presses the switch on the node where the DataCollectorAgent is running, the agent sends an instantaneous message having the values of the last computed features. Finally, the remote monitoring activity terminates when the user presses again a switch of the Sun SPOT on which the DataViewerAgent is running.

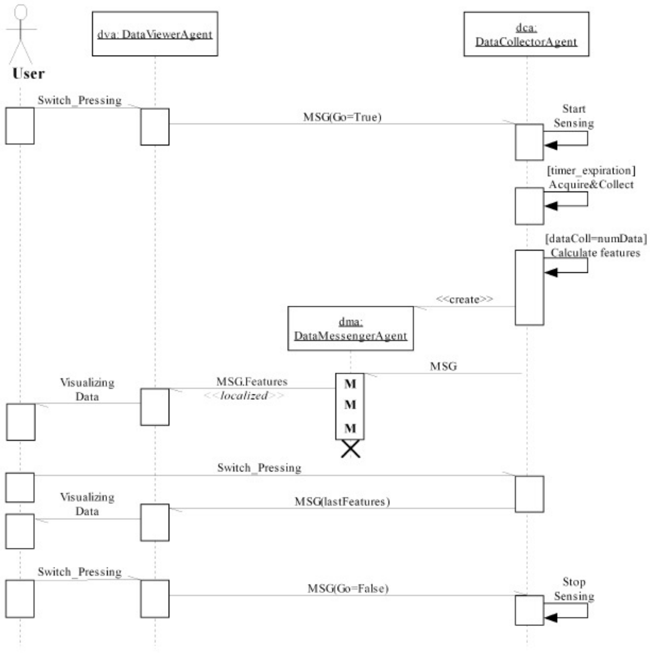


Fig. 7 M-UML sequence diagram for agents interactions

In the following subsections we describe how agent programming models offered by MAPS and AFME can be used to define the DataCollectorAgent behavior.

A. Agent definition in MAPS

As before illustrated, MAPS agents are modeled through a multi-plane state machine. In Fig. 8 the plane related to the DataCollectorAgent is depicted whereas a brief explanation is provided in the following. The *AGN_Start* event causes the transition from the agent creation state to the *IDLE* state whit the execution of an initializing code represented by the action *A0* (e.g. data structures initialization). In the *IDLE* state, when the network message (*MSG*) sent by the DataViewerAgent arrives and the guard *[go==true]* holds, the timer is configure and started for timing the sensors reading (action *A1*) and the agent transits to the WAITSENSING state. When the timer fires (see *TMR_Expired* event), the sensing operations are requested (action *A2*) to the three onboard sensors. When each of the three sensor data are available (see ACC_Tilt, TMP_Current and LGH_current events), their corresponding actions (*A5, A6, A7*) store the values on appropriate buffers. If *numData* samples for each sensor type have not been collected yet, the guard *[dataColl!=numData]* holds so that the agent returns to the WAITSENSING state waiting for the next sensing operations on the timer expiration. If at the contrary the necessary samples number is reached, sensor data are ready for being transmitted to the second sensor node. So, the set of the features are compute and the DataMessengerAgent is created (action *A9*). When the *AGN_Id* event, containing the agent id of the created agent, is received the set of the features values are passed to it (action *A10*). Regarding the WAITSENSING and the DATACOLLECTING states, if the user presses the switch, the last set of computed features are immediately sends to the DataViewerAgent through a remote message (action *A3*). Finally, when the event *MSG* is received and the guard *[go==false]* holds, the agent is terminated (action *A4*).

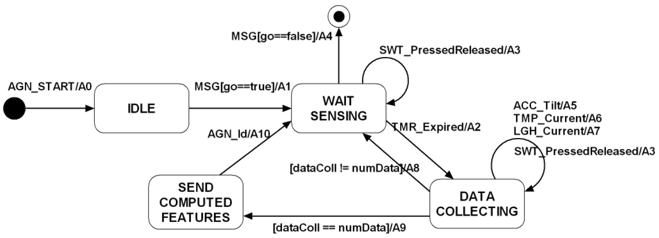


Fig. 8 MAPS-based DataCollectorAgent model

After having described the DataCollectorAgent plane, the code of some actions is provided in Fig. 9. In particular, the most significant part of the Java code related to the plane consists in the operations included into the set of the actions previously defined for the plane state machine.

```
A1:  Event timer = new Event(this.agent.getId(),
        this.agent.getId(), Event.TMR_EXPIRED, Event.NOW);
     timerID = this.agent.setTimer(true, 3000, timer);
A2:  Event temperature = new Event(this.agent.getId(),
        this.agent.getId(), Event.TMP_CURRENT, Event.NOW);
     temperature.setParam(ParamsLabel.TMP_CELSIUS,"true");
     this.agent.sense(temperature);
        //similar code for accelerometer and light
A3:  Event msg = newEvent(this.agent.getId(),
             dataViewerAgentID,Event.MSG, Event.NOW);
     msg.setParam("lastFeatures", this.computedFeatures);
     this.agent.send(this.agent.getId(),dataViewerAgentID,
                                             msg, true);
A6:  this.collectedData += event.getParam(
               ParamsLabel.TMP_TEMPERATURE_VALUE)+"-";
     dataCollTemp++;
A9:     // code for feature computation...
     this.agent.create("applications.demo.Messenger",
        null,this.agent.getMyIEEEAddress().asDottedHex());
A10: Event msg = new Event(this.agent.getId(),
              messengerAgentID, Event.MSG, Event.NOW);
     msg.setParam("features", this.features);
     this.agent.send(this.agent.getId(), messengerAgentID,
                                             msg, true);
```

Fig. 9 Java code for some of the DataCollectorAgent plane actions

B. Agent definition in AFME

The DataCollectorAgent specified trough the AFME design model is depicted in Fig. 10.

The components constituting the previous model are described in the following:

- 7 *Perceptors*. AccPerc, LightPerc and TempPerc are used to acquire data from the SunSPOT sensors. VerifyNumDataSamplesPerc is needed for perceiving if all necessary sensor data have been collected for features computation whereas VerifyFeaturesComputedPerc checks that all features (min, max, mean) have been computed on the collected sensor samples. Finally, TimerPerc checks when the

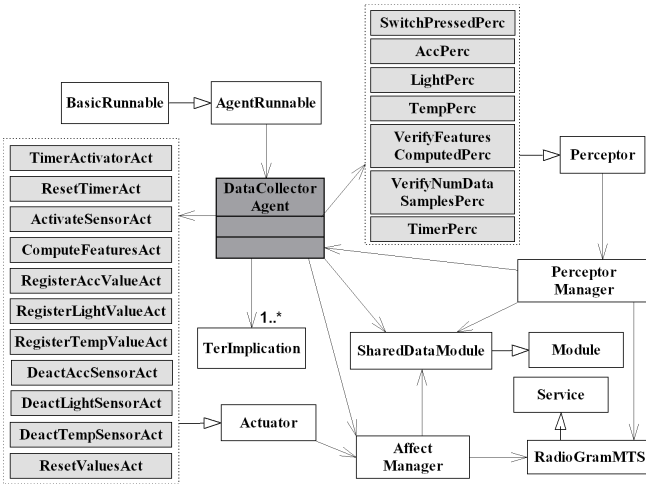


Fig. 10 AFME-based DataCollectorAgent model

timer expires and SwitchPressedPerc recognizes the user switch pressing.

- 11 *Actuators*. TimerActivatorAct and ResetTimerAct are used to activate/reset the timer of the sensor node for timing the sensor acquisition operations. RegisterAccValueAct, RegisterLigthValueAct, and RegisterTempValueAct allow storing the sensed data into the SharedDataModule for sharing them with the proper aforementioned perceptors. DeactAccSensorAct, DeactLigthSensorAct, and DeactTempSensorAct actuators are activated after the sensor readings and are needed to prevent the sensors perceptors from reading sensor data before the timer expiration. In particular, since all the agent perceptors, driven by the AFME runtime system, cyclically perceive the agent environment, and since the sensor data acquisition is made through the use of perceptors, the continuous data acquisition has to be avoided. For this reason, ActivateSensorAct is used to re-enable sensors reading after timer expiration. ComputeFeatureAct is responsible for the actual feature computation on the collected sensor data whereas the ResetValuesAct actuator is used for reinitializing all necessary data structure after a completed feature computation.

- *Rules*. TerImplication contains the auto-generated Java description of the agent behavior rules defined into a script file (see below).

- 1 *Module*. SharedDataModule is the shared memory space which stores the data samples sensed from the node sensors (accelerometer, temperature, light) and other useful data to be shared among perceptors and actuators.

- 1 *Service*. RadiogramMTS represents the transport service for data transmission to remote nodes or basestation.

In the following the basic rules defining the DataCollectorAgent behavior are described:

1. message(inform, sender(dataViewer, addresses("radiogram://”+DataViewerAgentNodeAddr)), ?content), !goFalse > par(timerActivatorAct, adoptBelief(always(goFalse)));
2. timerExpired > par(timerActivatorAct, activateSensorsAct);
3. temperature(?value) > par(deactTempSensorAct, registerTempValueAct(?value));
4. light(?value) > par(deactLightSensorAct, registerLightValueAct(?value));
5. acc(?accX, ?accY, ?accZ) > par(deactAccSensorAct, registerAccValueAct(?accX, ?accY, ?accZ));
6. numDataSampled > computeFeatures
7. featuresComputed(?computedFeatures) > par(resetValuesAct, inform(agentID(messenger, addresses("radiogram://”+ DataMessengerAgentNodeAddr)), values(?computedFeatures)));
8. switch_pressed(?computedFeatures) > inform(agentID(dataViewer, addresses("radiogram://”+ DataViewerAgentNodeAddr)), values(? computedFeatures));
9. message(inform, sender(dataViewer, addresses("radiogram://” +*DataViewerAgentNodeAddr*)), ?content), goFalse > resetTimerAct, deactAccSensorAct, deactLigthSensorAct, deactTempSensorAct;

The rule (1) enables the timer upon the reception of the message coming from the DataViewerAgent and creates the *goFalse* belief. Rule (2) states that after timer expiration the sensor reading operations are activated and also the timer is reactivated. Rules (3, 4, 5) are used for storing the sensor

```
    TempPerc:
....
public void perceive(){
   FOS currentFOS = manager.perManage("shareddata", 5);
   int activated= integer.parseInt(currentFOS.toString()
);
   if(activated == 1) {
      try {
         tempValue= EDemoBoard.getInstance().
                 getADCTemperature().getCelsius();
      } catch(IOException e){ e.printStackTrace(); }
      this.adoptBelief("temperature("+ value +")");
   }
}
....
    ActivateSensorAct:
....
public boolean act(FOS arg0) {
   System.out.println("ActivateSensorsAct..");
   m.actOn("shareddata", 2, null);
   return true;
}
....
    Shareddatamodule:
....
   // data structures declaration
   ...
public FOS processPer(int id)throws MalformedLogicExcep{
   ...
   else if(id == 5){
      return FOS.createFOS(""+temperatureSensorActivated);
   }
   ...
}
public FOS processAction(int id, FOS data)
                                  throws
                               MalformedLogicExcep{
   ...
   else if(id == 2){
      temperatureSensorActivated= 1;
      lightSensorActivated= 1;
      accSensorActivated= 1;
      return null;
   }
   ...
}
....
```

Fig. 11 Java code excerpt from the AFME DataCollectorAgent

data when are available and at the same time they disable the sensing perceptors. The rule (6) checks if *numData* samples have been acquired for each sensor type, and, in that case, features computation starts. Rule (7) verify that features have been correctly computed and if this is the case, the DataMessengerAgent is notified through a message. Rule (8) states that if the user presses the switch a message with the last computed features is sent to the DataViewerAgent. Finally, the last rule is for checking the reception of a message from the DataViewerAgent and since the *goFalse* belief has been previously inserted by the first rule, the sensing operations are stopped (timer is reset and sensors reading are disabled).

In Fig.11, a code excerpt of the AFME agent is provided and in particular, an example of *perceive* method (from the TempPerc perceptor), *act* method (from the ActivateSensorsAct actuator) and the SharedDataModule are shown.

V. Conclusion

In this paper we have presented several TinyOS-based and Java-based agent platforms for WSNs. In particular, we have described and compared MAPS and AFME in more details, two Java-based platforms offering similar services but through different agent programming models. The MAPS programming model is based on finite state machines and events for defining agent's behavior, whereas AFME provides a BDI-like agent model based on perceptors, actuators and rules. Experimentation experience with MAPS and AFME in the proposed case study and in other application domains suggests that both platforms are effective in supporting agent-based development of WSN application. However, agent programming with AFME is less straightforward than MAPS programming due to the BDI-like AFME agent model, which is more complex than the finite state machine model offered by MAPS. Moreover, finite state machine programming is a model often used in programming embedded systems so it may be more easily exploited by low-level programmers. Ongoing work is devoted to defining a solution for agent communication interoperability between MAPS and AFME agents which would enable the development of heterogeneous agent-based WSN applications.

References

[1] Eiko Yoneki and Jean Bacon, "A survey of Wireless Sensor Network technologies: research trends and middleware's role," Technical Report UCAM-CL-TR-646, University of Cambridge, UK, Sept. 2005.

[2] Danny B. Lange and Mitsuru Oshima, "Seven Good Reasons for Mobile Agents," *Communications of the ACM*, Vol. 42, No. 3 March 1999.

[3] TinyOS, documentation and software, http://www.tinyos.net, (2010).

[4] Sun™ Small Programmable Object Technology (Sun SPOT), http://www.sunspotworld.com/ (June 2010).

[5] C-L. Fok, G-C. Roman, C. Lu, "Rapid Development and Flexible Deployment of Adaptive Wireless Sensor Network Applications," in *Proc. of the 24th Int'l Conf. on Distributed Computing Systems* (ICDCS'05), Columbus (OH), Jun 6-10, pp. 653-662, 2005.

[6] Y Kwon, S. Sundresh, K. Mechitov, G. Agha, "ActorNet: An Actor Platform for Wireless Sensor Networks", in *Proc. of the 5th Int'l Joint Conference on Autonomous Agents and Multiagent Systems* (AAMAS), pages 1297-1300, 2006.

[7] R. Kent Dybvig, *The Scheme programming language*. Prentice-Hall, 1987.

[8] F. Aiello, G. Fortino, R. Gravina, A. Guerrieri, "MAPS: a Mobile Agent Platform for Java Sun SPOTs," In *Proceedings of the 3rd International Workshop on Agent Technology for Sensor Networks* (ATSN-09), jointly held with the 8th International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS-09), 12th May, Budapest, Hungary, 2009.

[9] Aiello, F., Fortino, G., Gravina, R. and Guerrieri, A, "A Java-based Agent Platform for Programming Wireless Sensor Networks," *The Computer Journal*. to appear (2010).

[10] Mobile Agent Platform for Sun SPOT (MAPS), documentation and software at: http://maps.deis.unical.it/ (June 2010).

[11] Muldoon, C., O'Hare, G. M. P., O'Grady, M. J. and Tynan, R., "Agent Migration and Communication in WSNs", in *Proc. of the 9th International Conference on Parallel and Distributed Computing, Applications and Technologies* (2008).

[12] Agent Factory Micro Edition (AFME), documentation and software at http://sourceforge.net/projects/agentfactory/files/ (June 2010).

[13] C. Muldoon, G. M. P. O Hare, R.W. Collier, and M. J. O'Grady, "Agent Factory Micro Edition: A Framework for Ambient Applications," in *Proc. of Intelligent Agents in Computing Systems*, ser. Lecture Notes in Computer Science, vol. 3993. Reading, UK: Springer, 28-31 May 2006, pp. 727–734.

[14] http://www.agentfactory.com (last access June 2010).

[15] A. S. Rao and M. P. Georgeff, "BDI Agents: from theory to practice," *Proceedings of the First International Conference on Multi-Agent Systems* (ICMAS'95), pp. 312–319, June 1995.

[16] Dikaiakos, M., Kyriakou, M., and Samaras, G., "Performance evaluation of mobile-agent middleware: A hierarchical approach," *Proc. of the 5th IEEE Int'l Conference on Mobile Agents*, Atlanta, Georgia, 2–4 December, LNCS 2240, Springer Verlag, Berlin, 2005, pp. 244–259.

[17] Saleh, K., El-Morr, C., "M-UML: An extension to UML for the modeling of mobile agent-based software systems," *Information and Software Technology*, 46(4), pp. 219-227.

[18] Sentilla Developer Community website, http://www.sentilla.com/developer.html (June 2010).

ISBN 978-83-60810-27-9
ISSN 1896-7094

BeesyBees – Efficient and Reliable Execution of Service-based Workflow Applications for BeesyCluster using Distributed Agents

Paweł Czarnul, Mariusz Matuszek, Michał Wójcik and Karol Zalewski
Faculty of Electronics Telecommunications and Informatics, Gdansk University of Technology
Email:{pczarnul,mrm}@eti.pg.gda.pl

***Abstract*—The paper presents an architecture and implementation that allows distributed execution of workflow applications in BeesyCluster using agents. BeesyCluster is a middleware that allows users to access distributed resources as well as publish applications as services, define service costs, grant access to other users and consume services published by others. Workflows created in the BeesyCluster middleware are exported to BPEL and executed by agents in a distributed environment. As a proof of concept, we have implemented a real workflow for parallel processing of digital images and tested it in a real cluster-based environment. Firstly, we demonstrate that engaging several agents for distributed execution is more efficient than a centralized approach. We also show increasing negotiation time in case of too many agents. Secondly, we demonstrate that execution in the proposed environment is reliable even in case of failures. If a service fails, a task agent picks a new equivalent service at runtime. If one of task agents fails, another of remaining agents takes over its responsibilities. The communication between the middleware, agents and services is encrypted.**

I. Introduction

INTEGRATION of various systems and components is one of the most crucial challenges that needs to be addressed today. In service-based environments this requires efficient algorithms for scheduling workflows i.e. selection of services for particular tasks while meeting optimisation goals. In the literature [1], [2] the workflow is usually defined as a directed acyclic graph $G(V, E)$ where vertices V denote tasks to be executed while edges denote time dependencies between the tasks. A workflow can be concrete if there is one service to be executed for each task or abstract, in which case there is a set of functionally equivalent services for each task, possibly with different QoS parameters. The latter almost always include execution time and cost and if needed others, such as reliability, reputation of the provider etc. One service needs to be chosen for each task such that a global goal is optimized with possibly meeting other QoS constraints. Typical optimisation goals include minimisation of execution time with an upper bound on the total cost of selected services which makes the problem NP-hard. Such workflows are executed by workflow engines which invoke services, control their statuses and transfer data.

II. Problem Statement

Workflows represent an integration of tasks, in which a set of alternative services may be defined (either manually or automatically) for each task. Alternatives may be statically linked to a task or dynamically assigned at runtime. Regardless of which service is selected for a task, the workflow should be executed in such a way that efficiency and reliability of workflow execution should be maximised. We assume that the workflow definition as well as the optimisation criteria and solver are already given [3] which means that services selected for particular tasks are known.

At this point, the problem becomes how to manage the execution of a distributed workflow application so that the following goals are optimized and maintained:

- efficiency — because of potentially high communication latency and low bandwidth of WANs, communication between services through agents close to interacting services is faster than through a distant centralised execution engine;
- reliability — since the node(s) on which the workflow execution engine runs as well as service locations may be geographically distributed, it is likely that at times connections may be lost; the solution should ensure reliable and continuous execution in such cases;
- security — all service invocations and data flows should be performed in a fully secure manner.

We present an agent-based distributed management mechanism that addresses all of the above and examine its performance in a practical example executed in a real environment.

III. Related Work

A. Workflow Types and Workflow Management Environments

Literature studies bring examples of many middlewares and environments for workflow execution. First and foremost, these do differ based on the target workflows to be executed. Scientific workflows usually focus on integration of computational services into chains executed on HPC resources such as clusters and supercomputers, or analysis of data acquired from input sensors or devices by services installed on powerful machines. Such workflows can span over various geographically distributed sites and virtual organisations forming grid systems [4], [5] and usually focus on data flow. There are several systems [4] that support such applications including Kepler [6], Gridbus, Triana [7], Pegasus [8], P-GRADE [9], Directed Acyclic Graph Manager (DAGMan), ICENI [10], UNICORE, Taverna, GridFlow, GrADS, Askalon, GridAnt.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Conversely, business workflows focus on discovery, selection and integration of services offered by various parties so that certain QoS parameters are met. Such workflows focus more on controlling the flow and often incorporate constructs such as choice, loop and other conditionals. Business workflows usually consider many more QoS parameters apart from the execution time and cost such as reliability, accessibility, fidelity, conformance, security etc. Systems such as Meteor-S [11], [12] focus on automating the process of service discovery and selection using an ontology-based approach. Selected services are executed automatically making service discovery, selection and execution almost fully automatic, based on the given workflow specification and given services and their descriptions. BPEL [13] is often used for describing business oriented workflows and includes control and data flows as well as service invocations. There are several execution engines for workflows specified in BPEL such as The ActiveBPEL Engine [14], bexee [15] or Silver [16].

Thirdly, in the context of pervasive and ubiquitous computing, workflows often react to events asynchronously and more importantly the context is considered for invocation of a service and changing the state. The latter is defined as information defining the state of an object [17]. As an example, uWDL [18] is used for describing ubiquitous workflows and allows specification of both the services as well as the context and service flow through the `node` and the `link` elements respectively.

B. *Existing Service-based Solution in BeesyCluster*

The already existing workflow support module in BeesyCluster developed by our research group [3], [19] contains a workflow editor, an optimiser and execution engine and has already been tested on a variety of scientific and business workflow applications, such as multimedia processing, numerical simulations [19] or business workflows [3].

BeesyCluster is a middleware that allows its users to invoke sequential or parallel applications exposed as services on registered system accounts on various clusters and servers. Such services can be assigned to particular workflow tasks in BeesyCluster's editor. The workflow editor is implemented by an applet. Created workflows are saved in BeesyCluster's database. Based on service execution times and costs, BeesyCluster's optimiser selects one service for each task so that the given criteria are optimised. It can optimise either a linear combination of the total cost of selected services and the workflow execution time or e.g. the execution time with a global constraint on the total cost of selected services. The execution engine is implemented within a Java EE server. For each workflow node a `SIMessageBean` which is a message driven bean is responsible for executing the service chosen for a particular workflow task by the optimizer is used. This is done within the `onMessage` method which is executed upon receiving a JMS message. For the given task, after the service has been executed, the bean copies output files to dedicated directories of services chosen for successor tasks and initiates execution of following tasks by sending JMS messages. The execution status of a particular node instance is updated in the database dynamically.

The drawback of this solution is centralized management of execution. If large data is passed between services, it needs to be passed through BeesyCluster which increases the workflow execution time. We propose to optimize this by launching several distributed JADE agents that would launch services for workflow tasks and act as local proxies for communication between the services. Comparison to other agent-based approaches is presented in Section V.

IV. Proposed Agent-based Solution

As stated in the previous section, BeesyCluster's execution engine is implemented within a Java EE server. This approach allows for easy execution management, but it can also create a bottleneck for efficient execution and a single point of failure. Should a problem with either the server or its network connectivity occur, the whole workflow execution may be at risk. This in itself is a serious limitation.

By using a set of well defined, industry standard conformant interfaces, we migrated the task of execution of a prepared workflow to an agent-based environment, which we nicknamed *BeesyBees*. By separating the two tasks we also gained a possibility of having a pluggable architecture, allowing for experimentation with different approaches to workflow execution. By implementing the execution management in mobile agents we take advantage of enormous flexibility of this environment.

In this paper, we concentrate on increased efficiency and reliability of the workflow execution, which is a main difference to original BeesyCluster's solution.

A. *Architecture*

The BeesyBees system is based on the JADE (Java Agent DEvelopment Framework) [20] agent system implementing the FIPA [21] communication and the OMG MASIF [22] management standards. It provides distributed and decentralized workflow execution for BeesyCluster using agents.

BeesyCluster itself can be regarded as a middleware that offers an easy-to-use WWW interface to access various accounts on various geographically distributed clusters and servers through a single account in BeesyCluster. This allows management of files and directories on such clusters and servers, compiling and running sequential and parallel applications with support for various queuing systems, a team work environment with sharing data and others. Furthermore, users can publish own applications as services within BeesyCluster and make them available to other BeesyCluster users. A provider can specify a cost the client would need to pay for invoking the service. BeesyCluster contains a subsystem for handling virtual payments between users. Access to clusters and servers as well as running particular commands, searching for and running services is also possible through the Web Service interface [23], [24].

Figure 1 shows a generalised diagram for BeesyBees system architecture. There are four types of agents implemented in the system:

- GateWayAgent — receives workflow descriptions in BPEL and spawns TaskAgents,
- GraphAgent — monitors the progress of workflow execution and gives a graphical representation of agent instances and decisions made,
- StatusAgent — collects reports on task execution states from TaskAgents and persist actual status of workflow realization,
- TaskAgent — executes a single workflow task. A group of TaskAgents executes the workflow cooperatively.

A workflow is composed using BeesyCluster's SIEditor applet, a tool for modeling workflows that consist of tasks with services assigned out of those defined in BeesyCluster. For the each task from one to few services can be chosen. First, the definition of the workflow to be run is fetched from BeesyCluster's database and saved in BPEL, which then is handed to the BessyBees system where it is received by the GateWayAgent. All the communication between BeesyCluster and BeesyBees is handled with appropriate web services. Then the GateWayAgent spawns TaskAgents in order to execute workflow's tasks.

TasksAgents negotiate the assignment of each task [25]. TaskAgent interested in executing a particular task sends its proposal including its matching score to all other agents. The score is an object which can be filled with various comparable metrics, like for example a load of a machine an agent resides on. When the agent receives other agent's execution proposals it agrees only when it is not interested in that particular task or it's matching score is lower. This process is represented in Figure 2. Finally, the TaskAgent which received approvals only, begins task execution. When the task is done the TaskAgent sends notification to all agents executing particular workflow, containing information which task has been done and using which service. Example of notification is presented in Figure 3.

The execution state of workflow's tasks is monitored by a StatusAgent. Messages sent by TaskAgents during workflow execution are monitored. These include information whether a task has been executed, or if there were problems with calling services. That data is saved persistently by the StatusAgent so it could possibly be used for recovery after the whole system crash. Additionally a GraphAgent which shows a graphical representation of a workflow execution can be turned on.

B. Efficiency

The implementation launches a certain number of software agents which negotiate which agents are responsible for execution of particular tasks. Secondly, the agents may run on a defined number of containers. This allows deployment and testing of both a fully centralized architecture in which one agent acts as a central management point or a fully distributed approach with several agents and containers. Obviously, the optimal number of agents and containers may depend on the size of the workflow as well as locations of actual services. Starting too many agents results in too much overhead for negotiation.

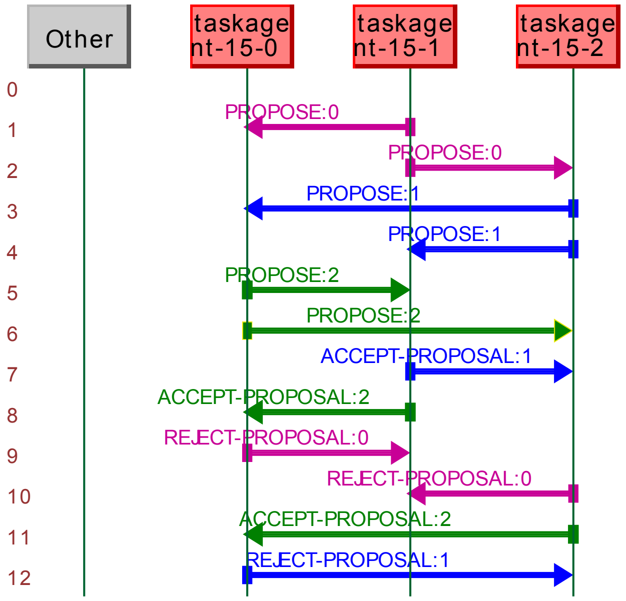


Figure 2. Negotiation between Agents

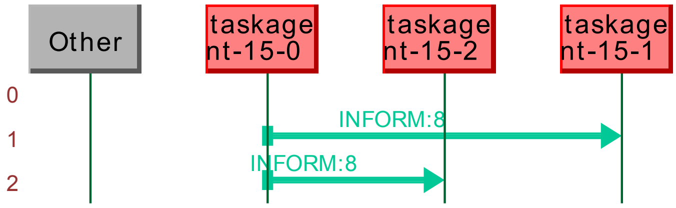


Figure 3. Notifications about Current Task Status

C. Fault tolerance

BeesyBees was designed with fault tolerance in mind. Currently, it is tolerant of service and TaskAgent failures. When either fails, the failure is recognised and either the respective flow path is restarted, or an alternative service is being sought and utilised. During task execution TaskAgent is blocking access to its assigned task by rejecting proposals concerning execution of that task. Thanks to this feature failure tolerance is achieved. When TaskAgent fails, nothing else is blocking an uncompleted task, so one of remaining agents picks it and proposes its execution to other agents. A service failure is detected after a specified number of unsuccessful retries. When this condition occurs an alternative service is chosen if available.

D. Security

At this point, we assume that the agent environment is managed by a group of trusted entities. We use security mechanisms implemented in JADE. Each container has its own certificate signed by our Certification Authority (CA) and it communicates with other containers using SSL.

Originally, BeesyCluster executes services on computing nodes using SSH. Each computing node has its own record

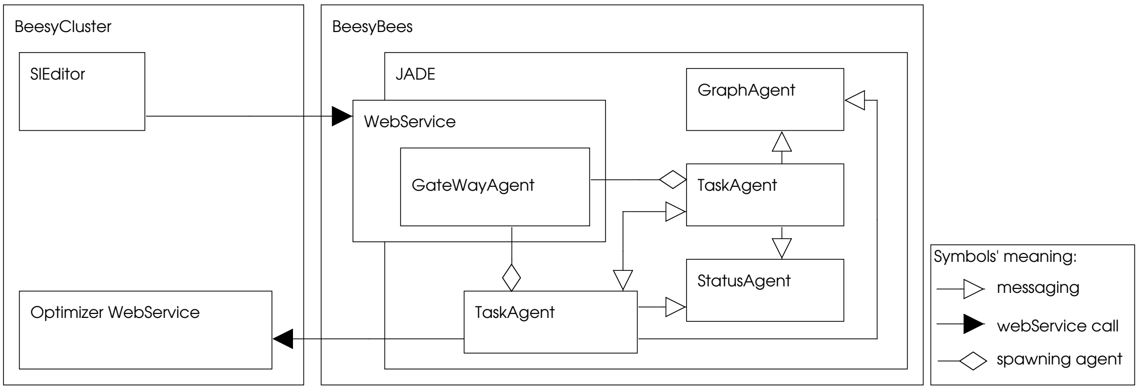


Figure 1. System Architecture

on the BeesyCluster's main server that contains its public key. In the agent-based approach we used a similar method. Each machine that contains an agent container has its own list of trusted computing nodes. Each agent can establish an SSH connection from a machine where it actually runs to communicate with computing nodes and invoke services.

The gateway agent is exposed as a Web Service and serves as a proxy between BeesyCluster and BeesyBees. The communication between BeesyCluster and the gateway agent is secured using HTTPS and certificates. There is always a possibility of attacks using holes that exist in software or solutions that we are depending on. In [26] such a problem as well as a way to mitigate its adverse effects is presented.

V. Other Agent-based Approaches to Workflow Management

One of similar projects we can mention is JBees [27] — a workflow management system based on agent technology combining collaboration agents and the coloured Petri net. As opposed to BeesyBees using BeesyCluster as a system for workflow creation, JBees makes use of built-in management agents providing user interface for the human workflow manager. Moreover resources in JBees, which can be compared to BeesyCluster services, are strictly integrated with the system and are represented by resource agents. The idea of process agents and storage agents is similar to BeesyBees task agents and state agents. Process agents are responsible for the execution of particular cases. Storage agents collect information from process agents (in JBees it is approached through the monitor agent, as opposed to BeesyBees where task agents communicate directly with the state agent) and make it persistent. As mentioned, resources in JBees are accessed by process agents requesting task execution through resource agents. In BeesyBees services are called directly by task agents. As for workflow description, JBees makes use of coloured Petri nets [28].

WS2JADE presented in [29] is a tool for runtime deployment and control of web services. It is of interest, because it uses the same agent system, JADE. The main goal of this project is integration of agent systems and web services, it is approached by representing each web service by a specialised agent. Similarly to JBees, web services are called through their associated agents. In order to call a web service, a client agent searches DF (Directory Facilitator, JADE built in service directory agent) for it, then if the service is not present, DF can trigger WS2JADE to look it up in the web service environment. If an appropriate service is found, the web service agent, which registers its service in DF, is created. Finally, the service is called by exchanging messages between agents.

SwinDeW-A [30] integrates services using WS2JADE by enhancing SwinDeW (Swinburne Decentralised Workflow) with agents. As opposed to BeesyBees using BeesyCluster's optimiser in order to choose best services for specified task, SwiNDew-A makes use of negotiation agents which are able to negotiate with a number of service agents. In order to choose the most suitable service, a Service Level Agreement (SLA) needs to be formed between the negotiation agent and each service provider. Agents can negotiate non functional parameters (such as time, cost, availability), which are similar to those proposed in BeesyCluster [3].

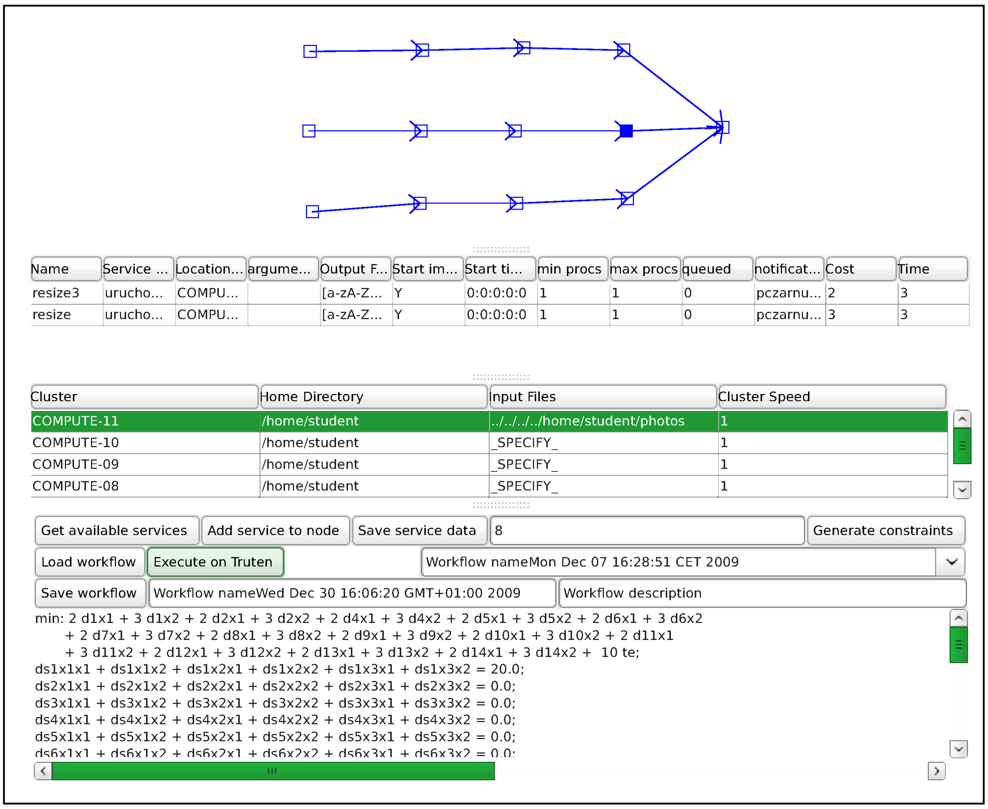
Finally, a relatively recent development is WADE (Workflow and Agents Development Environment) [31], which is an extension of JADE agent framework, facilitating both the possibility to define agent tasks according to the workflow metaphor and an architecture with mechanisms allowing administration of distributed WADE based applications. Workflows in WADE are expressed as Java classes.

VI. Simulations

A. Testbed Workflow and Environment

As an example, we have run a workflow application for parallel processing of RAW digital images in order to produce a Web album. The process uses standard steps in professional photo editing (`$1` is the input file name):

- `rawtotiff` — conversion from RAW to lossless TIFF. Implemented using the `dcraw` converter as `dcraw -T $1`



Name	Service ...	Location...	argume...	Output F...	Start im...	Start ti...	min procs	max procs	queued	notificat...	Cost	Time
resize3	urucho...	COMPU...		[a-zA-Z...	Y	0:0:0:0:0	1	1	0	pczarnu...	2	3
resize	urucho...	COMPU...		[a-zA-Z...	Y	0:0:0:0:0	1	1	0	pczarnu...	3	3

Cluster	Home Directory	Input Files	Cluster Speed
COMPUTE-11	/home/student	../../../../home/student/photos	1
COMPUTE-10	/home/student	_SPECIFY_	1
COMPUTE-09	/home/student	_SPECIFY_	1
COMPUTE-08	/home/student	_SPECIFY_	1

8

Workflow nameMon Dec 07 16:28:51 CET 2009

Workflow nameWed Dec 30 16:06:20 GMT+01:00 2009

Workflow description

```
min: 2 d1x1 + 3 d1x2 + 2 d2x1 + 3 d2x2 + 2 d4x1 + 3 d4x2 + 2 d5x1 + 3 d5x2 + 2 d6x1 + 3 d6x2
     + 2 d7x1 + 3 d7x2 + 2 d8x1 + 3 d8x2 + 2 d9x1 + 3 d9x2 + 2 d10x1 + 3 d10x2 + 2 d11x1
     + 3 d11x2 + 2 d12x1 + 3 d12x2 + 2 d13x1 + 3 d13x2 + 2 d14x1 + 3 d14x2 +  10 te;
ds1x1x1 + ds1x1x2 + ds1x2x1 + ds1x2x2 + ds1x3x1 + ds1x3x2 = 20.0;
ds2x1x1 + ds2x1x2 + ds2x2x1 + ds2x2x2 + ds2x3x1 + ds2x3x2 = 0.0;
ds3x1x1 + ds3x1x2 + ds3x2x1 + ds3x2x2 + ds3x3x1 + ds3x3x2 = 0.0;
ds4x1x1 + ds4x1x2 + ds4x2x1 + ds4x2x2 + ds4x3x1 + ds4x3x2 = 0.0;
ds5x1x1 + ds5x1x2 + ds5x2x1 + ds5x2x2 + ds5x3x1 + ds5x3x2 = 0.0;
ds6x1x1 + ds6x1x2 + ds6x2x1 + ds6x2x2 + ds6x3x1 + ds6x3x2 = 0.0;
```

Figure 7. Testbed Workflow in BeesyCluster's Editor

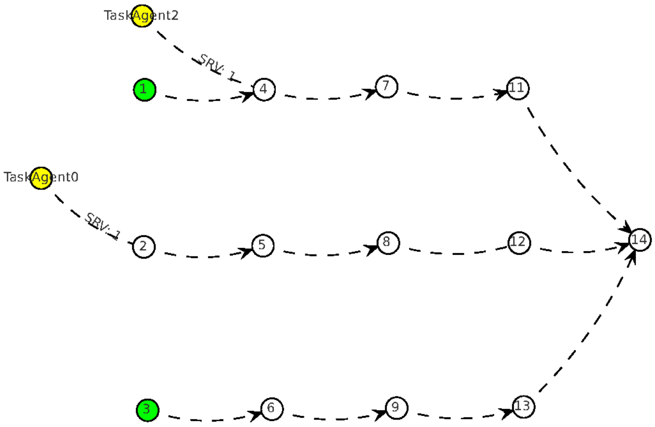


Figure 4. Visualisation of Workflow Execution

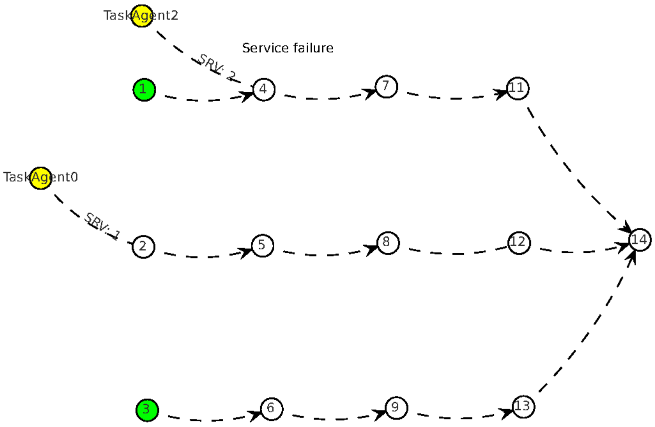


Figure 5. Visualisation of Workflow Execution with Service Failure

- `normalize` — normalisation of the image using ImageMagick's `convert` as `convert $1 -normalize $1`
- `sharpen` — sharpening the image implemented as `convert $1 -sharpen 1x1.2 $1.jpg`
- `resize` — resizing the image implemented as `convert $1 -resize 600x400 $1.jpg`
- `albumgeneration` — implemented by either `jigl` or `album`.

All those services are bash scripts executed through ssh. Input for each service contains pictures being processed.

Figure 7 presents the testbed workflow created in BeesyCluster's editor. Parallel paths include nodes running `rawtotiff`, `normalize`, `resize` and `sharpen` filters while the final node gathers resulting jpg images and produces

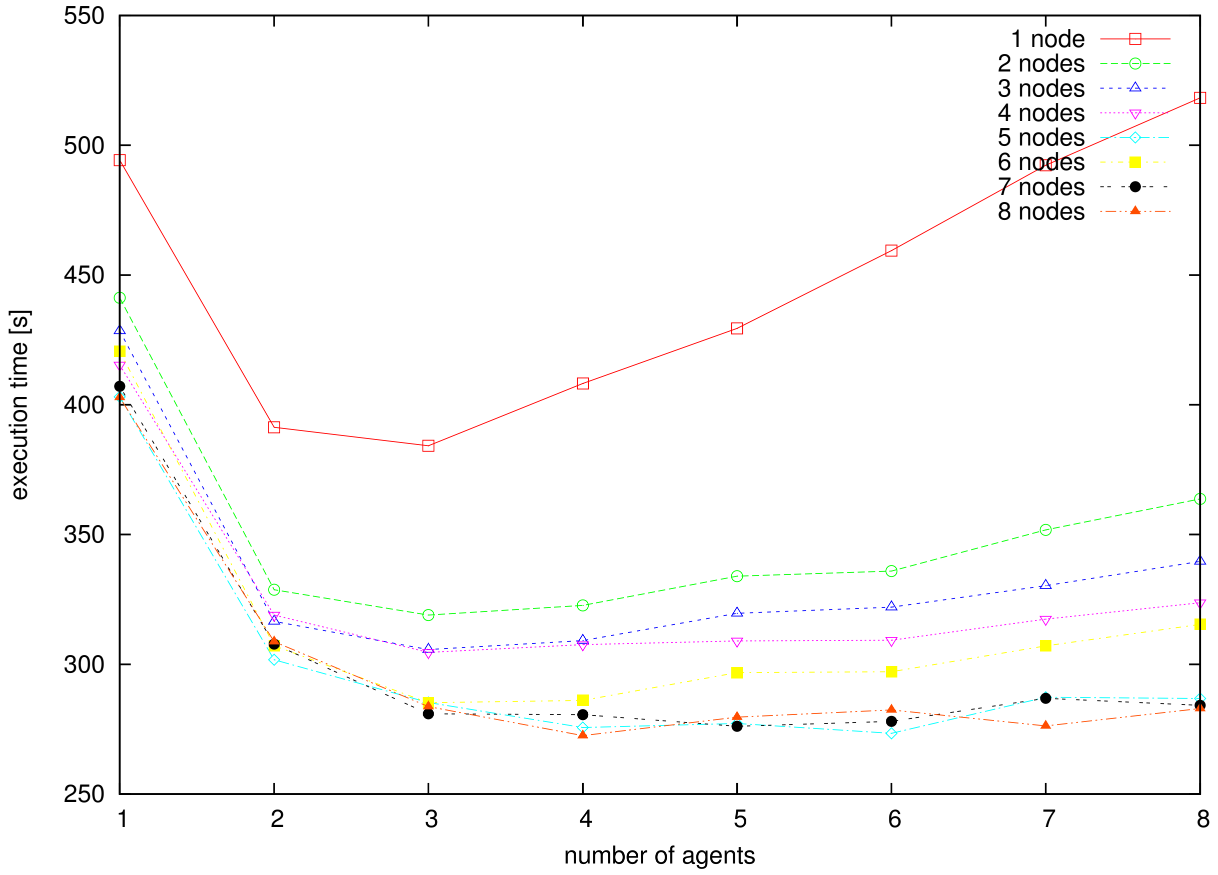


Figure 8. Execution Time

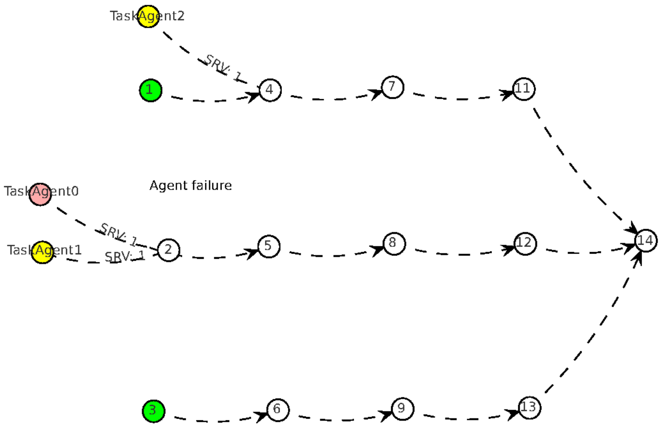


Figure 6. Visualisation of Workflow Execution with Agent Failure

a web album. For each node a primary and a backup services were deployed on a dedicated computer. Such a simple workflow was chosen on purpose, in order to facilitate the process of verification of experiment results with theoretical expectations.

There are several approaches to choosing appropriate services for tasks' execution. One is that following static optimisation done by BeesyCluster's optimiser before the run, task agents responsible for the given task call BeesyCluster's optimiser and fetch the selected service. Optimizer can consider such metrics like execution cost and time. Another approach, where knowledge about services is gathered during their execution and subsequently used to choose the best service considering metrics given, for example a probability of a service failure, was presented in [32]. If a particular service is not available or has a runtime failure, the agent selects the next best service for the task.

We have implemented a monitoring mechanism that shows the status of the workflow execution visually and updates it as the execution progresses. Figure 4 shows a snapshot of the testbed workflow being executed. Nodes 1–3 execute `rawtotiff`, nodes 4–6 execute `normalize`, nodes 7–9 execute `resize`, nodes 11–13 execute `sharpen` and node 14 executes `albumgeneration`. Figures 2 and 3 show communication between TaskAgents obtained by the JADE sniffing tool. They show the negotiation process described earlier and notifications about current task status sent to other TaskAgents respectively.

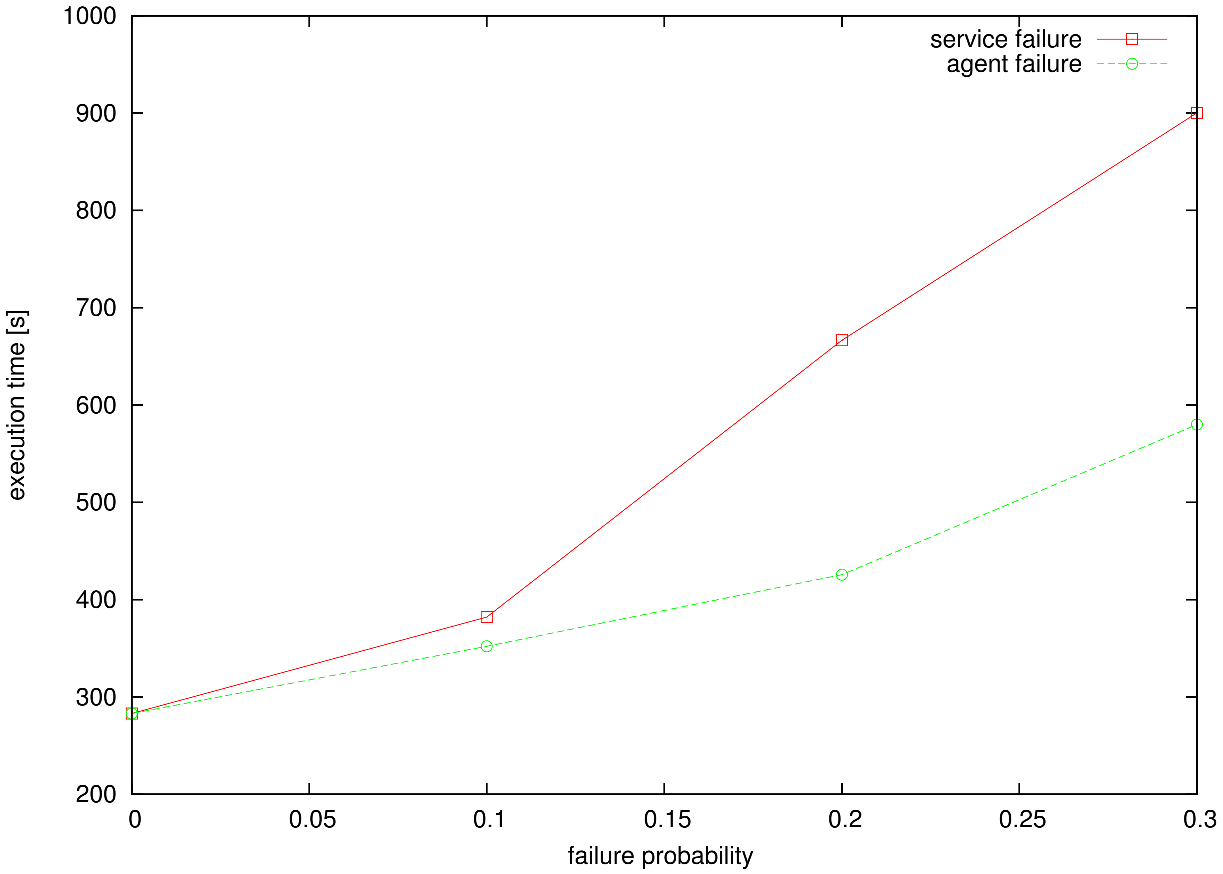


Figure 9. Execution Time in case of Failures

B. Optimization of Execution Time through Distributed Agent Processing

Figure 8 presents workflow execution time depending on the number of participating agents. They show execution times for processing the workflow with big files. There were 4 big files of 34MB each. Each of the series corresponds to a different number of system nodes on which agents work. The situation with only one agent and one agent node corresponds to a centralized environment, which performance should be similar to BeesyCluster's one because of using the same mechanism to invoke services. We can see that increasing the number of agents and nodes reduces the execution time, but only up to a certain number of agents. Further increase in a number of agents causes negotiation costs to offset any gains in execution time. In the testbed worklow there are three parallel paths and it can be seen that the results are best also for the number of working agents equal to three.

C. Mechanism for Fault-tolerant Execution and its Performance

We have been able to demonstrate that the execution environment is tolerant of faults in communication, service execution and agent execution. The workflow application can be completed successfully even if the following failures occur:

1) a service fails to complete after it has been invoked. As shown in Figure 5, if `taskagent2` detects a failure of service 1, it automatically switches to a functionally equivalent service 2 that executes the given task;
2) if a failure of a task agent occurs then another task agent takes over the responsibility of the former automatically. As shown in Figure 6, if `taskagent0` fails then `taskagent1` takes over its responsibility and the workflow execution can continue.

Figure 9 shows the total execution time of the worklow application assuming a certain probability of agent failure or service failure. It can be seen that execution time is getting higher with a higher probability of failure. If an agent fails, another one is given a chance for taking over the responsibility over its task and service. Again, it may fail with the given probability. For a given probability of agent or service failure, the final execution time is higher for the latter case. This is because for each task there is one agent but the service needs to be invoked four times because of four files being processed in each task. If any of the service invocations fails, a new service needs to be chosen, input files copied and again the service is invoked for the input files for the task.

VII. Conclusions

We described the architecture and its implementation that allows efficient, fault-tolerant and secure execution of workflows using agents. A standard execution management module would invoke remote services from one central server which would result in performance bottleneck and could fail if the server loses connections with the services. We have shown that for an optimal number of agents and containers, distributed management of workflow execution by agents is more efficient than a centralized approach. The distributed approach can complete successfully as long as agents supposedly located much closer to the services are able to reach the latter. We have been able to show that execution of a testbed workflow application for parallel processing of digital images completes successfully even if a task agent or a service fails.

Acknowledgment

Work sponsored by research grant N N516 383534 "Strategies for management of information services in distributed environments".

References

[1] J. Yu, R. Buyya, and C.-K. Tham, "Cost-based scheduling of workflow applications on utility grids," in *Proceedings of the 1st IEEE International Conference on e-Science and Grid Computing (e-Science 2005), IEEE CS Press*, Melbourne, Australia, December 2005.

[2] J. Yu and R. Buyya, "Scheduling scientific workflow applications with deadline and budget constraints using genetic algorithms," *Scientific Programming Journal*, 2006, iSSN: 1058-9244, IOS Press, Amsterdam, The Netherlands.

[3] P. Czarnul, "A jee-based modelling and execution environment for workflow applications with just-in-time service selection," in *Proceeding of 4th International Workshop on Workflow Management (ICWM2009), 4th International Conference on Grid and Pervasive Computing, GPC 2009*, Geneva, Switzerland, May 2009.

[4] J. Yu and R. Buyya, "A taxonomy of workflow management systems for grid computing," *Journal of Grid Computing*, vol. 3, no. 3-4, pp. 171–200, September 2005. [Online]. Available: http://dx.doi.org/10.1007/s10723-005-9010-8

[5] M. Wieczorek, A. Hoheisel, and R. Prodan, "Towards a general model of the multi-criteria workflow scheduling on the grid," *Future Generation Comp. Syst.*, vol. 25, no. 3, pp. 237–256, 2009.

[6] B. Ludascher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger-Frank, M. Jones, E. Lee, J. Tao, and Y. Zhao, "Scientific Workflow Management and the Kepler System," *Concurrency and Computation: Practice & Experience, Special Issue on Scientific Workflows*, 2005. [Online]. Available: http://www.sdsc.edu/%7Eludaesch/Paper/kepler-swf.pdf

[7] S. Majithia, M. S. Shields, I. J. Taylor, and I. Wang, "Triana: A Graphical Web Service Composition and Execution Toolkit," in *IEEE International Conference on Web Services (ICWS'04)*. IEEE Computer Society, 2004, pp. 512–524. [Online]. Available: http://www.trianacode.org/

[8] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, S. Patil, M.-H. Su, K. Vahi, and M. Livny, "Pegasus : Mapping Scientific Workflows onto the Grid," in *Across Grids Conference*, Nicosia, Cyprus, 2004. [Online]. Available: http://pegasus.isis.edu

[9] *Parallel Grid Runtime and Application Development Environment, User's Manual, ver. 8.4.2*, Laboratory of Parallel and Distributed Systems, MTA SZTAKI, Hungary. [Online]. Available: http://www.lpds.sztaki.hu/∼smith/pgrade-manual/manual.html

[10] L. Young, S. McGough, S. Newhouse, and J. Darlington, "Scheduling architecture and algorithms within the iceni grid middleware," 2003. [Online]. Available: citeseer.ist.psu.edu/young03scheduling.html

[11] R. Aggarwal, K. Verma, J. Miller, and W. Milnor, "Constraint driven web service composition in meteor-s," in *Proceedings of IEEE International Conference on Services Computing (SCC'04)*, 2004, pp. 23–30.

[12] ——, "Dynamic web service composition in meteor-s," LSDIS Lab, Computer Science Dept., UGA, Technical Report, May 2004.

[13] Andrews, T., et al., "Business Process Execution Language for Web Services," 2003, version 1.1, BEA, IBM, Microsoft, SAP, Siebel.

[14] T. A. Engine, 2009, active Endpoints. [Online]. Available: http://www.activevos.com/community-open-source.php

[15] bexee, "BPEL Execution Engine," 2004, berne University of Applied Sciences. [Online]. Available: http://bexee.sourceforge.net/index.html

[16] G. Hackmann, M. Haitjema, C. Gill, and G. Roman, "Sliver: A bpel workflow process execution engine for mobile devices," in *in: Proceedings of 4th International Conference on Service Oriented Computing (ICSOC*. Springer Verlag, 2006, pp. 503–508.

[17] S. Ben Mokhtar, D. Fournier, N. Georgantas, and V. Issarny, "Context-Aware Service Composition in Pervasive Computing Environments," in *Rapid Integration of Software Engineering Techniques, Second International Workshop : RISE 2005*, Heraklion, Crete Greece, 2006, pp. 129–144. [Online]. Available: http://hal.archives-ouvertes.fr/inria-00415111/en/

[18] J. Han, E. Kim, and J. Choi, "Workflow language based on web services for autonomic services in ubiquitous computing," in *Proceedings of International Conference on Artificial Reality and Telexistence, ICAT*, Coex, Korea, 2004.

[19] P. Czarnul, "Integration of compute-intensive tasks into scientific workflows in beesycluster," in *Computational Science – ICCS 2006*, ser. LNCS, vol. 3993. Springer, 2006, pp. 944–947.

[20] "JADE (Java Agent DEvelopemnt Framework) Online Documentation," Telecom Italia Lab. [Online]. Available: http://jade.tilab.com/doc/index.html

[21] "FIPA specifications," The Fundation of Intelligent Physical Agents. [Online]. Available: http://www.fipa.org/

[22] "Mobile Agent System Interoperability Facilities Specification," Object Management Group. [Online]. Available: http://www.omg.org/cgi-bin/doc?orbos/97-10-05

[23] P. Czarnul, "Integration of compute-intensive tasks into scientific workflows in beesycluster," in *Proceedings of ICCS 2006 Conference,*. University of Reading, UK: Springer Verlag, May 2006, lecture Notes in Computer Science, LNCS 3993.

[24] P. Czarnul, M. Bajor, M. Fraczak, A. Banaszczyk, M. Fiszer, and K. Ramczykowska, "Remote task submission and publishing in beesycluster : Security and efficiency of web service interface," in *Proceedings of PPAM 2005*, Springer-Verlag, Ed., vol. in press in LNCS, Poznan, Poland, Sept. 2005.

[25] M. Matuszek, "Agent cooperation strategies in execution of complex distributed services," Ph.D. dissertation, Gdansk University of Technology, 2007.

[26] "Cve-2009-3555." [Online]. Available: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-3555

[27] L. Ehrler, M. Fleurke, M. Purvis, and B. T. R. Savarimuthu, "Agent-based workflow management systems (wfmss)," *Agent-based workflow management systems (WfMSs)*, vol. 4, no. 1, pp. 5–23, 2006.

[28] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, Apr 1989.

[29] X. T. Nguyen, R. Kowalczyk, M. B. Chhetri, and A. Grant, "Ws2jade: A tool for run-time deployment and control of web services as jade agent services," *Software Agent-Based Applications, Platforms and Development Kits*, pp. 223–251, 2006.

[30] J. Yan, Y. Yang, R. Kowalczyk, and X. Nguyen, "A service workflow management framework based on peer-to-peer and agent technologies," in *Quality Software, 2005. (QSIC 2005). Fifth International Conference on*, Sept. 2005, pp. 373–380.

[31] "Workflows and Agents Development Environment," Telecom Italia Lab. [Online]. Available: http://jade.tilab.com/wade/

[32] P. Czarnul, M. Matuszek, M. Wójcik, and K. Zalewski, "Beesybees – agent-based, adaptive & learning workflow execution module for beesycluster," in *Faculty of ETI Annals, Information Technologies vol. 18*, 2010.

ISBN 978-83-60810-27-9
ISSN 1896-7094

A Technique based on Recursive Hierarchical State Machines for Application-level Capture of Agent Execution State

Giancarlo Fortino* and Francesco Rango
DEIS – University of Calabria, Via P. Bucci cubo 41c, 87036 Rende (CS), Italy
Email: g.fortino@unical.it, frango@si.deis.unical.it

***Abstract*—The capture of the execution state of agents in agent-based and multi-agent systems is a system feature needed to enable agent checkpointing, persistency and strong mobility that are basic mechanisms supporting more complex, distributed policies and algorithms for fault tolerance, load balancing, and transparent migration. Unfortunately, the majority of the currently available platforms for agents, particularly those based on the standard Java Virtual Machine, do not provide this important feature at the system-level. Several system-level and application-level approaches have been to date proposed for agent state execution capture. Although system-level approaches are effective they modify the underlying virtual machine so endangering compatibility. Conversely, application-level approaches do not modify any system layer but they provide sophisticated agent programming models and/or agent converters that only allow a coarse-grain capture of agent state execution.**

In this paper, we propose an application-level technique that allows for a programmable-grain capture of the execution state of agents ranging from a per-instruction to a statement-driven state capture. The technique is based on the Distilled StateCharts Star (DSC*) formalism that makes it available an agent-oriented type of recursive hierarchical state machines. According to the proposed technique a single-threaded agent program can be translated into a DSC* machine by preserving its original semantics. Although the proposed technique can be applied to any agent program written through an imperative-style programming language, it is currently implemented in Java and integrated into the JADE framework, being JADE one of the most diffused agent platforms. In particular, agents, which are specified through a generic Java-like agent language, are translated into JADE agents according to the JADE DSCStarBehaviour framework. A simple yet effective example is used to illustrate the proposed technique.

I. Introduction

AGENT-BASED and Multi-Agent Systems are developed around the concept of agent, a goal-directed, computational and interacting entity which acts on behalf of another entity (or entities) [20]. Such agent systems are supported by agent platforms that basically provide agent programming libraries and system-level services to support agent execution. An important system-level feature that an agent platform needs to include for the development of robust and fault-tolerant agent systems is agent state execution capture. In fact, agent checkpointing, persistency and strong mobility are mechanisms fully enabled by agent state execution capture. Agent checkpointing [13] is a technique for inserting fault tolerance into agent systems. In particular, it basically consists of storing a snapshot of the agent execution state and, later on, using it for restarting the agent execution in case of its failure. Agent persistency is a mechanism allowing saving an agent along with its state into the mass memory as a file. Archived agents can be later on loaded and resumed. Strong mobility [10] is an important feature of an agent that enables the migration of the agent state, data and code. Such mechanisms constitute the basis for supporting more complex, system-wide agent policies and algorithms for fault tolerance, load balancing, and transparent migration.

Unfortunately, even after more than one decade of research on agent systems, the majority of the currently available agent platforms, particularly those based on the standard Java Virtual Machine (JVM) do not provide this important feature at system-level. In fact, currently the standard JVM does not yet provide mechanisms to capture the state execution of Java processes. Nevertheless, several approaches have been to date proposed to overcome such an issue. They can be roughly taxonomized into three categories: system-level [4, 5, 14, 17], converters [3, 16, 15, 11, 12] and model-based [18, 9, 19]. The system-level approach modifies the underlying virtual machine to capture the execution state at process/thread level; however, the modified virtual machine is usually not compatible for agents previously developed. Conversely, converters and model-based approaches are application-level approaches that rely on specific, sometimes sophisticated, agent programming models and/or on converters at bytecode-level or source-code-level to allow for agent execution state capture. Although they do not require any virtual machine modification, the grain of capture of the agent execution state is usually coarse so that capture can be carried out only at specific points of agent execution. In fact, capture is either automatically driven by specific statements (in agent converters) or manually programmed by exploiting the reference agent programming models. Both techniques can be used only for agent-driven capture and not for an effective system-driven capture.

In this paper, an application-level technique for agent-driven and system-driven capture of agent state execution is proposed. Agent-driven capture is based on three key statements (checkpoint, persistence and move), whereas system-

* Corresponding author

driven capture is fine-grain and carried out on a per-instruction basis. A distinctive feature of the defined technique is the capture of the execution state in recursive and mutually recursive methods that can also contain key statements. The proposed technique is based on the Distilled StateCharts Star (DSC*) formalism that provides an agent-oriented type of recursive hierarchical state machine through which the agent program and its execution state can be represented. According to the technique, a single-threaded agent program formalized as a main method and ancillary methods is translated into a DSC* object, preserving the original agent program semantics. The technique is currently implemented in Java and integrated into the JADE framework [2]. In particular, an agent program, which is formalized through a simple Java-based Agent Language (JAL), is converted into a JADE agent compliant to a newly defined JADE behavior named DSCStarBehaviour. Conversion can be carried out in two modes: (i) driven by the statements *checkpoint*, *persistence* and *move*; (ii) instruction by instruction. The so obtained JADE behavior not only has the same semantics as the original agent program but also incorporates its execution state at run-time. A simple yet effective example, concerning an agent-based version of the Fibonacci program, is used to exemplify the proposed technique.

The rest of this paper is organized as follows. In Section II the DSC* formalism is defined as an enhancement of the Distilled StateCharts formalism [9]. Section III presents the proposed technique that is then exemplified in section IV by means of the Fibonacci agent example. Section V provides some details about the implementation of the proposed technique through Java and the JADE framework. Finally, conclusions are drawn and future work anticipated.

II. The Distilled StateCharts Star Formalism

The Distilled StateCharts Star (DSC*) formalism is based on of the Distilled StateCharts (DSC) formalism [9, 8] that was specifically defined for effective modeling of single-threaded agent behavior through hierarchical state machines based on ECA rules, OR-decomposition, history entrance mechanisms and UML-like execution semantics based on the run-to-completion step. In addition, DSC* introduce typical mechanisms of recursive functions/procedures so that DSC* machines are recursive hierarchical state machines [1]. A DSC* machine is formalized by the following tuple:

DSC* = <Σ, s_0, *fs, L*, Φ, *deep, defaultHistoryEntrance, defaultEntrance, defaultEntranceAction, V, star, return*>, where:

- Σ is the set of states including composite states (cs=ncs$\cup$pcs), which can be normal composite states (ncs) or procedure composite states (pcs), simple states (ss) and history pseudostates (hs). State notation is defined according to the hierarchical structure of the DSC*. In particular: (i) the notation A(B) indicates that the simple state B is encapsulated in the composite state A; (ii) the notation A(•) indicates that A is a composite state; (iii) the notation A(x), where x is a placeholder variable, indicates any state included in A (e.g. x=B).
- s_0 is the initial state.
- *fs* is the set of final states.
- *L* is the set of transition labels. A label is defined as the following ECA rule: *Event[Condition]/Action*. The semantics implies that if Event is processed and Condition is true, upon transition firing Action is first atomically executed and, then, the state transition takes place.
- Φ is the set of transitions among states. In particular, it is defined as follows: $\Phi \subseteq \Sigma \times \Sigma \times L$. Thus a transition is formalized as a triple <source state, target state, label>.
- *defaultEntrance* is a function that indicates the default entrance of each composite state. The default entrance is the transition sourcing from the initial pseudostate of a composite state and targeting a state encapsulated into the composite state. Default entrances are not included in Φ as initial pseudostates are implicitly associated to composite states and are not explicitly defined in Σ.
- *defaultEntranceAction* is a function that associates an action (if any) to the default entrance of composite states.
- *deep* is an attribute of a history pseudostate indicating deep history if true, shallow history if false. History mechanisms allow a partial (through shallow history pseudostate H) or full (through deep history pseudostate H*) recovery of the state history after re-entering into a composite state previously exited.
- *defaultHistoryEntrance* is a function that indicates the default history entrance (if any) of the history pseudostates. The default history points to the state to be entered in case history is not initialized, i.e. when the composite state containing the history pseudostate is entered for the first time.
- *V* is a set of variables hierarchically scoped according to the state structure. In particular, $V(x)=\{v_1(x),\ldots,v_n(x)\}$ is the set of variables declared in the x state.
- *star* and *return* are attributes of specific transitions which respectively support activation of and return from a procedure type composite state (or procedure state) that semantically behaves like a procedure in an imperative-style programming language. Thus the *star* transition represents the procedure call whereas the *return* transition represents the procedure return. Let A and B be two composite states, when the transition (*) from A to B is drawn, the transition (<<R>>>) from B to A is to be added too (see Figure 1). The CallB event is generated inside A and triggers the (*) transition, whereas the ReturnFromB event is generated inside B and triggers the (<<R>>) transition. Procedure call parameters are passed to B through the CPE_B event that is generated inside A just after the CallB generation. Return value is passed through the RPE_B event that is generated just after the ReturnFromB event. The procedure state activation and return semantics are as follows: if the DSC* is in the A state and the CallB event is processed, the transition (*) is fired so that a new instance of the B

state is entered and the B state parameters (if any) are actualized with the parameter values contained in the CPE_B event. When the B state has to complete, the ReturnFromB event and the RPE_B event, filled with the return value, are generated. Once the ReturnFromB event is processed by the DSC*, the transition <<R>> fires so that the B state instance is removed and the A state is entered through its shallow history pseudostate. The so defined mechanism is therefore equivalent to the procedure call mechanism; parameters are only passed by value. Procedure state recursion and mutual recursion among procedure states are also fully supported (see Section III).

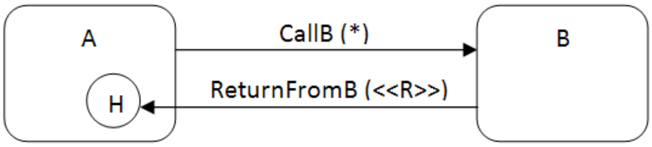


Fig. 1 Star (*) and return (<<R>>) transitions

The structure of a well-formed DSC* consists of a top state enclosing one procedure state, named main procedure state, which contains the initial state and the main final state, and zero or more secondary procedure states linked directly or indirectly to the main procedure state (see Section III for an example).

The execution semantics of a DSC* machine are defined in terms of an abstract machine, which embodies a DSC*, whose key components are:

- An event queue (EQ), which holds incoming event instances until they are dispatched;
- An event dispatching mechanism (EDM), which selects and de-queues event instances from EQ.
- An event processor (EP), which processes dispatched events.

The semantics of event processing is based on the run-to-completion (RTC) assumption implying that an event can only be de-queued and dispatched if the processing of the previous event is fully completed.

Given the last dequeued event by EDM, the main task of EP is to cyclically:

1. find out which transitions are enabled, i.e., which transitions could be fired based on the dequeued event;
2. select, among the enabled transitions, the transitions to be fired. Indeed, only one transition of a DSC* is step-by-step ready to be fired.

The algorithm executed by EP is reported in Figure 2 through a pseudocode notation.

The input to each step is given by the last recently dequeued event (current_event) and the DSC* execution control status, which includes the current state (current_state) of the DSC*, the dynamic history information associated to history pseudostates (history(hs_i), ∀i), and the stacks of the procedure states (stack(cs_j), ∀j). At the termination of each step, the output is given by the new DSC* execution control status.

```
1 :for each history pseudostate i
2 : history[i].init(defaultHistoryEntrance[i])
3 :for each procedure state j
4 : stack(j).init(null)
5 :current_state = $s_0$;
6 :while (current_state∉fs) do
7 : current_event = EQ.dequeue();
8 : enabled_transitions =
     enabled(current_state, current_event);
9 : if (enabled_transitions∉∅) then
10:  fired_transition = fire(enabled_transitions);
11:  if (return(fired_transition)) then
12:   pop(stack(source(fired_transition)));
13:  else
14:   if (source(fired_transition)∈cs)
15:    updateHistory(source(fired_transition))
16:   endIf
17:  endIf
18:  executeActionChain(fired_transition);
19:  if (star(fired_transition)) then
20:   j=target(fired_transition);
21:   push(stack[j], new instance of j);
22:  endIf
23:  current_state = nextconf(current_state,
                              fired_transition);
24: endIf
25:endWhile
```

Fig. 2 The DSC* execution semantics algorithm

Before starting the event processing loop, history pseudostates are initialized with the default history entrances, the stack of the procedure states is set to null and the current_state set to the initial state (lines 1-5). While the current_state is different from a final state (line 6), the current_event is dequeued (line 7) and successively processed. In particular the following steps are carried out: (i) the set of enabled transitions is computed on the basis of the current_state and current_event (line 8); (ii) if such a set is not empty the transition to be fired is selected among the enabled transitions (lines 9-10); (iii) if the fired transition is a <<R>> transition then the instance at the top of the stack of the procedure state which is the source of the transition is removed (lines 11-12); otherwise, if the source state of the fired transition is composite, its history pseudostates are updated (lines 14-15); (iv) the action chain of the fired transition is executed (line 18); (v) if the fired transition is a (*) transition a new instance of the target procedure state is added to its stack (lines 19-22); (vi) finally the new current state is computed on the basis of the current_state and the fired transition (line 23).

III. A DSC*-based Technique for Agent State Execution Capture at Application-level

The proposed technique translates an agent program, defined according to the schema reported in Figure 3, into a DSC* machine that is semantically equivalent to the agent program and contains its execution state at run-time. Translation can be carried out either per instruction basis (named full translation) or driven by key statements. The agent program, which represents a single-threaded agent, is composed of a main method (simply called main) from which the agent activity starts and zero, one or more supporting methods $\{m_1,..., m_N\}$, N>=0. Agent data can be declared at global level (GV) and at local level in each method (LVm_i). The supported control-flow statements are: sequential, selective (if-then-else), iterative (for and while), procedure call and return. Moreover the defined key statements are: (i) *checkpoint*, which captures the execution state of the agent and store it in memory; (ii) *persistence*, which captures the execution state of the agent and store it in a file; and (iii) *move*, which triggers the migration of the agent from the current location to a new location.

```
program Agent{
 GV //Declaration of Global Variables
 m1(param1,…,paramn_m1):return_value_m1{
  LV1 // Declaration of Local Variables
  CBm1 //code block of the method 1
 }
 …
 mN(param1,…,paramn_mN):return_value_mN{
  LVN // Declaration of Local Variables
  CBmN //code block of the method N
 }
 main(param1,…,paramn_main){
  LV_main // Declaration of Local Variables
  CBm_main //code block of the main
 }
}
```

Fig. 3 The schema of the Agent program

The structure of the Agent program is represented by the equivalent DSC* machine structure shown in Fig. 4. Each procedure is formalized as a procedure state, the global variables V(Active) are declared at the top state named Active, which is a normal composite state, the local variables of the main V(main) along with the main parameters P(main) at the main level, the local variables $V(m_i)$ of the supporting methods along with the method parameters $P(m_i)$ at the level of the corresponding procedure states. The Active composite state, which is embedded in the FIPA agent behavior template [6], is always entered through the deep history pseudostate to allow restoring the DSC* machine status to the status in which the DSC* machine was before leaving the Active state.

Each statement of the Agent program can be translated into a DSC* diagram. In the following we describe the translation patterns of the Agent program statements and, finally, we present a short version of the translation algorithm. We defined two types of statements: special statements, which include control-flow statements (if-then-else, while, for, procedure call, return) and key statements (checkpoint, persistence, move), and normal statements, which include all other statements different from the special statements.

In Figure 5 the translation of the normal statement is reported. The E event drives the execution of the `statement` bringing the DSC* machine into the S state which formalizes the control-flow status just after the execution of the `statement`. A new event E is generated to drive the control-flow to the next control-flow point. This translation is carried out if and only if the translation is full.

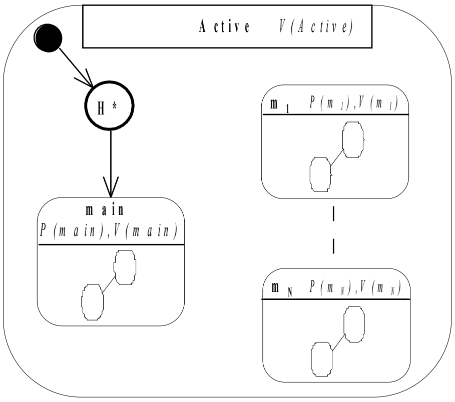


Fig. 4 The schema of DSC* machine of an Agent program

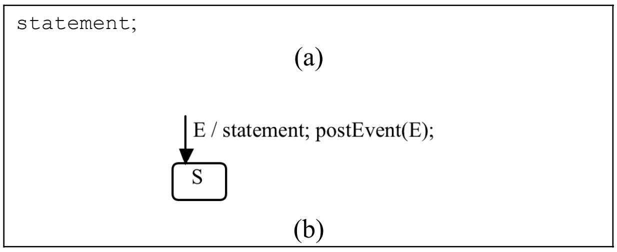


Fig. 5 Translation of the normal statement: (a) normal statement; (b) corresponding DSC* diagram

The translation of the `key_statement` is shown in Figure 6. In this case, the list of normal statements (containing one or more normal statements) which are before the `key_statement` are moved into the transition action before the key statement. This translation is carried out if and only if the translation is key-statement-driven; otherwise the translation is carried out as for a normal statement (see above).

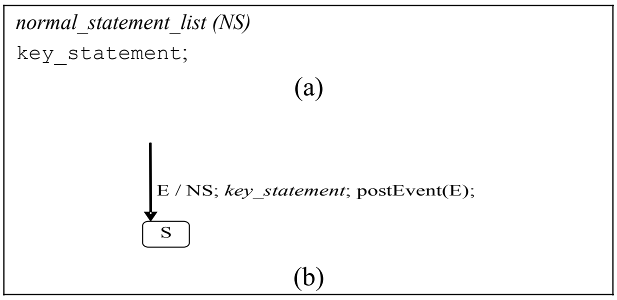
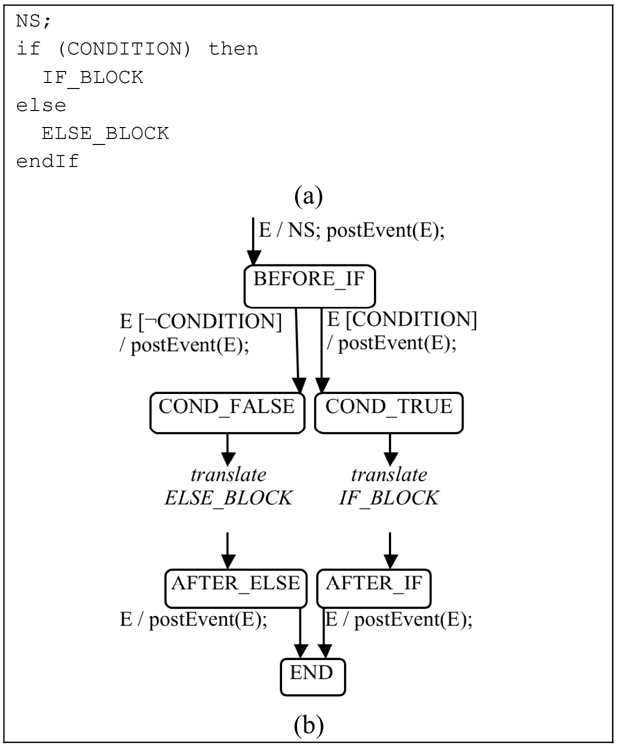

normal_statement_list (NS)
key_statement;
(a)
E / NS; key_statement; postEvent(E);
S
(b)

Fig. 6 Translation of the key statement: (a) key statement; (b) corresponding DSC* diagram

The translation of the `if-then-else` statement is reported in Figure 7. The Before_If state indicates the control flow point before the execution of the `if-then-else` statement. Cond_True and Cond_False represent the states that indicate the points just after the `CONDITION` evaluation and before the execution of the `IF_BLOCK` and the `ELSE_BLOCK`, respectively. The After_If and After_Else states represent the points after the execution of the `IF_BLOCK` and `ELSE_BLOCK`, respectively. Finally the End state represents the point after the `if-then-else` statement.



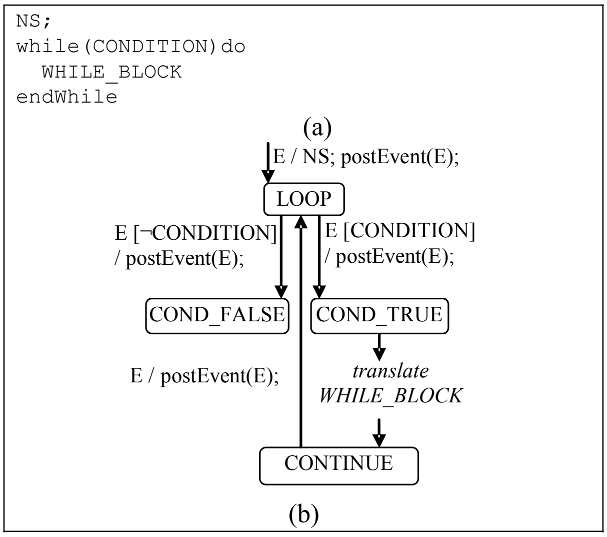
```
NS;
if (CONDITION) then
  IF_BLOCK
else
  ELSE_BLOCK
endIf
```

Fig. 7 Translation of the if-then-else statement: (a) if-then-else statement; (b) corresponding DSC* diagram

The translation of the `while` statement is reported in Figure 8. The Loop state indicates the control-flow point before the execution of the `while` statement. The Cond_True state represents the state indicating the point just after the `CONDITION` is evaluated true and before the execution of the `WHILE_BLOCK`. If the `CONDITION` does not hold the Cond_False state is reached which represents the control flow point after the `while` statement.

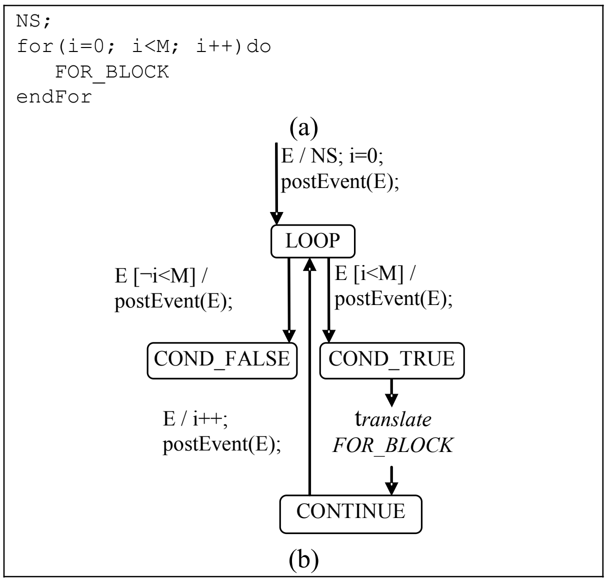
Figure 9 reports the translation of the `for` statement. The Loop state indicates the control-flow point before the execution of the `for` statement. The `for` counter `i` is initialized in the action of the transition above Loop. The Cond_True state represents the state indicating the point just after the `condition i<M` is evaluated true and before the execution of the `FOR_BLOCK`. If `condition` does not hold the Cond_False state is reached which represents the control-flow point after the `for` statement. The `for` counter `i` is incremented in the action of the transition from Continue to Loop. Of course, the `for` condition as well as the `for` counters can be easily generalized.

The translation of the `procedure call` is reported in Figure 10. To implement the call to the P2 method, the CallP2 and CPE_P2 events are sequentially posted. The former labels the (*) transition between P1 and P2. The latter contains the parameter values {PARAMETERS_P2} to be passed to P2. When the CPE_P2 event is processed, the parameters value are extracted and assigned. When the RPE_P2, generated in the P2 state (see translation of the `return` statement), is processed, the return value contained in the RPE_P2 event is extracted and assigned.



```
NS;
while(CONDITION)do
  WHILE_BLOCK
endWhile
```

Fig. 8 Translation of the while statement: (a) while statement; (b) corresponding DSC* diagram



```
NS;
for(i=0; i<M; i++)do
   FOR_BLOCK
endFor
```

Fig. 9 Translation of the for statement: (a) for statement; (b) corresponding DSC* diagram

The translation of the `return` statement is reported in Figure 11. The ReturnFromP2 and RPE_P2 events are sequentially posted. The former drives the <<R>> transition between P2 and P1. The latter contains the return parameter value to be passed back to P1.

The translator algorithm (Figure 12) converts an agent program (AP) defined as in Figure 3 into a DSC* diagram (AP*), preserving the AP semantics. In particular, for each method m of AP (line 1) the following steps are carried out:

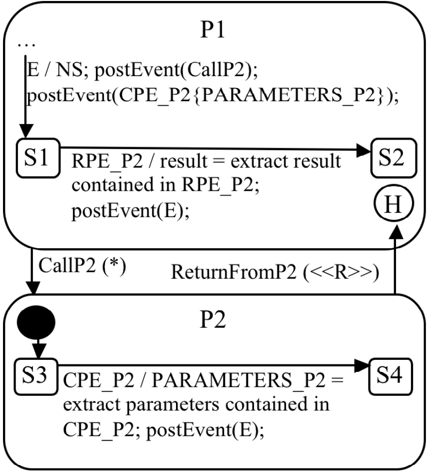
(i) a procedure state pcs corresponding to m is created and added to AP* (line 2);

(ii) the variables P(m) and V(m) are added to pcs (line 3);

(iii) the initial state of pcs is defined along with the transition labeled by the CPE event, through which the method parameters could be passed, which connects s1 and s2 repre-

```
RETURN_TYPE_P1 p1(PARAMETERS_P1){
 ...
   NS;
    RETURN_TYPE_P2  result  =  p2(PARAMETERS_P2
);
     ...
}
```

(a)

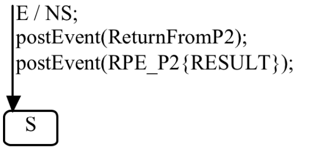


(b)

Fig. 10 Translation of the procedure call statement: (a) procedure call statement; (b) corresponding DSC* diagram

```
RETURN_TYPE_P2 p2(PARAMETERS_P2){
  ...
  NS;
  return RESULT;
  ...
}
```

(a)



(b)

Fig. 11 Translation of the return statement: (a) return statement; (b) corresponding DSC* diagram

senting the control-flow starting point of m (lines 4-7); the tc variable represents the translation cursor pointing to the last created state from which translation is to be restarted;

(iv) the body of m is finally translated through translateCodeBlock. In particular, translateCodeBlock sequentially translates the statements belonging to the code block cb from the 1st to the k-th. Given the h-th statement (s_h) the following steps are carried out:

(i) if `full_translation` is false and s_h belongs to SS and contains directly or indirectly a key_statement then s_h is translated according to the translation patterns and NS is reset (lines 3-6); otherwise, s_h is inserted in NS (line 7);

(ii) if `full_translation` is true and s_h belongs to SS then s_h is translated according to the above described translation patterns (line 10); otherwise, s_h is translated as normal statement (line 11).

```
SS      set of special statements
S(AP)   set of statements of AP
CB(AP)  set of code blocks of AP
M(AP)   set of methods of AP
tc      translation cursor
fullTranslation {true, false}

AP* translator(AP)
1 : for each m∈M(AP) do
2 :  pcs = createCompositeState(m)
3 :  declaration in pcs of variables P(m), V(m)
4 :  s1 = createSimpleState(pcs)
5 :  defaultEntrance(pcs)=s1
6 :  s2 = createSimpleState(pcs)
7 :  create CPE labeled transition between s1 and
s2
8 :  tc=s2;
9 :  translateCodeBlock(body(m), m, pcs)
10: endFor

translateCodeBlock(cb∈CB(AP), m∈M(AP), cs∈PCS)
1 : NS=<>
2 : for each sh∈cb|h=1..k do
3 :  if ¬fullTranslation then
4 :   if sh∈SS and sh⊒key_statement then
5 :    handleSpecialStatement(sh, m, cs, NS)
6 :    NS=<>
7 :   else NS = NS + sh;
8 :   endIf
9 :  else
10:   if sh∈SS handleSpecialStatement(sh, m, cs, <>
)
11:   else translateNormalStatement(sh, m, cs)
12:   endIf
13:  endIf
14: endFor
```

Fig. 12 A schema of the translation algorithm

IV. A Translation Example: Fibonacci Agent

To exemplify the technique presented in the previous section, the translation of a Fibonacci mobile agent, reported in Figure 13, is described. The agent is an extension of the Fibonacci algorithm with mobility: it moves across N locations to compute fibonacci(N) and, finally, prints out the result at the home location.

The translation was carried out both full and statement-driven; for the sake of exemplification, in the following we discuss the DSC* diagram of the Fibonacci agent obtained according to the key-statement-driven translation (see Figure 14). In particular, the `move` statement drives the translation so that the agent execution state can be captured before any migration and restored after migration. It is worth pointing out the DSC*-based modeling of the direct recursion: the Fibonacci procedure state has both a self (*) transition labeled by the CallFibonacci event and a self <<R>> transition labeled by the ReturnFromFibonacci event.

V. A JADE-based Implementation

The proposed technique is implemented in Java for the JADE framework that is one of the most diffused agent platforms [2]. The implementation comes with a translator program and a JADE add-on, named DSCStarBehaviour framework, which provides the programming abstractions supporting the definition of agent behaviors in terms of DSC* ma-

```
public class FibonacciAgent extends JALAgent {
 Location [] locations;
 public int fibonacci(int n){
  int x;
  int res1;
  int y;
  int res2;
  int res;
  if (n == 0 || n == 1) {
   move(locations[0]);
   return n;
  }
  move(locations[n]);
  x = n - 1;
  res1 = fibonacci(x);
  y = n - 2;
  res2 = fibonacci(y);
  res = res1 + res2;
  return res;
 }
 public void main(Location [] loc){
  int result;
  locations = loc;
  result = fibonacci(10);
  System.out.println("RESULT = " + result);
 }
}
```

Fig. 13 Fibonacci agent

chines. In particular, a Java-based agent program is converted by the translator into a JADE agent based on the DSCStarBehaviour. The agent program is defined through a Java-based agent language (JAL) which allows structuring agents as single-threaded Java programs using classes of the JADE framework and the three key statements (checkpoint, persistence, and move). The DSCStarBehaviour framework is an enhancement of the JADE DistilledStateChartBehaviour framework [7]; a simplified version of its class diagram is reported in Figure 15.

The DSCStarEvent extends ACLMessage so that the basic message processing mechanisms of JADE are completely reused. The DSCStarBehaviour extends CompositeBehaviour and implements mechanisms fulfilling the DSC* execution semantics. All DSC* composite states extend DSCStarBehaviour and are handled through the JADE Behaviour mechanisms. The other classes are abstractions of the main components of the DSC* formalism: variables and parameters of procedure states (DSCStarVariablesAndParameters), (*) transition (DSCStarStarTransition) and <<R>> transition (DSCStarReturnTransition) which are an extension of normal transition (DSCStarNormalTransition), and instances of a procedure state (VirtualInstance).

A simplified excerpt of the obtained FibonacciAgent class is reported in Fig 16 to provide a flavor of programming with the JADE DSCStarBehaviour framework. `FibonacciAgent` extends the basic JADE Agent class and contains the two procedure states `main` and `fibonacci` (lines 1-3). In the `setup` method all the simple states are defined along with the transitions; finally the `FibonacciAgent` behavior is started up (lines 4-28). In particular: (i) states `S1`-`S17` are declared, created and added to `main` and `fibonacci` as shown in lines 7-9; (ii) in line 10, the `root` behavior, which represents the encapsulating Active state (see Fig. 14), is created; (iii) the recursive (*) transition is defined in line 11; (iv) the transition `t2` between the `S12` and `S13` states (see Fig. 14) is defined in lines 12-24; two methods are defined for each transition: `trigger`, which defines the transition trigger *event[condition]*, and `action`, which defines the transition action, e.g. action of `t2`; (v) all transitions are added to the `root` which is added to the pool of active behaviors (lines 25-26); (vi) finally, the CPE_MAIN event, which triggers the execution of the FibonacciAgent, is created and posted (lines 27-28).

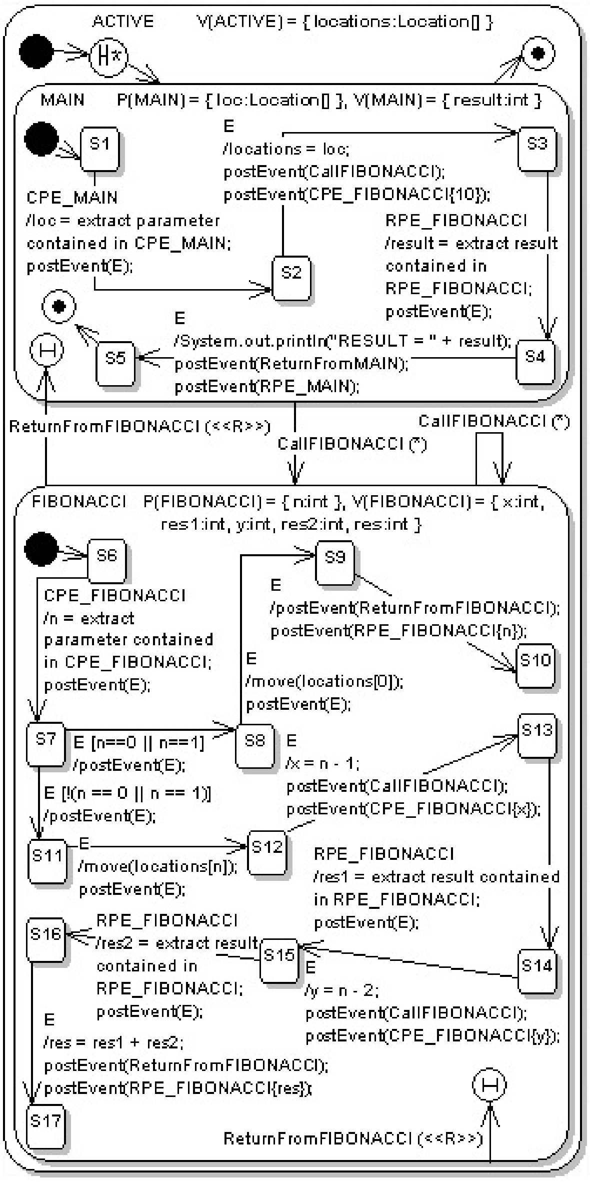


Fig. 14 DSC* diagram of the Fibonacci agent based on the *move-driven* translation

VI. Conclusion

In this paper we have presented a novel technique based on hierarchical recursive state machines for application-level capture of agent execution state. In particular, the technique uses the DSC* formalism to formalize agent programs writ-

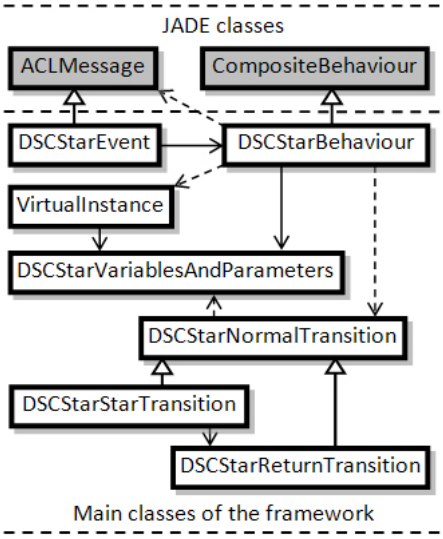


Fig. 15 The DSCStarBehaviour class diagram

ten in an imperative style language. The technique is currently implemented in Java for the JADE framework and allows obtaining JADE agents, whose behavior is based on the newly defined DSCStarBehaviour framework, which can be actively and passively checkpointed, made persistent and strongly migrated. On-going research efforts are devoted to (i) analyze the code and execution overhead introduced by the proposed application-level approach and (ii) implement the technique for other Java-based agent platforms.

```
public class FibonacciAgent extends Agent {
 private DSCStarBehaviour fibonacci;
 private DSCStarBehaviour main;
 protected void setup(){
  fibonacci=new DSCStarBehaviour(this,"FIBONACCI",..);
  main=new DSCStarBehaviour(this,"MAIN",...);
  Behaviour S1=new SimpleStateBehaviour(this,"S1");…;
  main.addInitialState(S1); main.addState(S2);…;
  fibonacci.addInitialState(S6);
  fibonacci.addState(S7);…;
  DSCStarBehaviour root =
     DSCStarBehaviour.createRootForDSCStar(main,...);
  DSCStarStarTransition t1 = new
          DSCStarStarTransition(fibonacci, fibonacci);
  DSCStarNormalTransition t2 = new
               DSCStarNormalTransition("T2",S12,S13){
   public boolean trigger(ACLMessage msg,...) {
    if (msg instanceof E) return true;
    return false;
   }
   public void action(...) {
    VarFibonacci varFibonacci = (VarFibonacci)
                      sourceVariablesAndParameters;
    varFibonacci.x = varFibonacci.n - 1;
    DSCStarEvent call = new
           DSCStarEvent(DSCStarEvent.CALL,fibonacci);
    postMessage(call);
    CPEFibonacciEvent cpe = new
        CPEFibonacciEvent(DSCStarEvent.CPE,fibonacci);
    cpe.n = varFibonacci.x; postMessage(cpe);
   }}
  root.addTransition(t1);…;root.addTransition(t17);
  addBehaviour(root);
  CPEMainEvent cpe_main = new
                CPEMainEvent(DSCStarEvent.CPE,main);
  postMessage(cpe_main);}}
```

Fig. 16 A FibonacciAgent code excerpt

References

R. Alur, M. Benedikt, K. Etessami, P. Godefroid, T. W. Reps, and M. Yannakakis, “Analysis of recursive state machines,” *ACM Transactions on Programming Languages and Systems*, 27(4), 2005.

F. Bellifemine, A. Poggi, and G. Rimassa,, “Developing multi agent systems with a FIPA-compliant agent framework,”. *Software Practice And Experience* 31, 103-128, 2001.

L. Bettini and R. De Nicola, “Translating Strong Mobility into Weak Mobility,” *Proc. of 5th IEEE Conference on Mobile Agents*, G. Picco (ed), LNCS 2240, 2001, pp. 182-197.

S. Bouchenak and D. Hagimont, “Pickling threads state in the Java system,” *Proc. of Technology of Object-Oriented Languages and Systems Europe – Europe* (TOOLS Europe’2000), Mont Saint Michel / Saint Malo, France, Jun. 2000.

S. Bouchenak and D. Hagimont, “Zero Overhead Java Thread Migration,” *Technical Report N°0261*, INRIA, Montbonnet-St-Martin(France), May 2002.

FIPA Agent Management Specification, Management for agents on FIPA agent platforms, http://www.fipa.org/specs/fipa00023/SC00023K.html.

G. Fortino, F. Rango, W. Russo, “Statecharts-based JADE agents and tools for engineering Multi-Agent Systems”, *Proc of 14th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (KES2010), Cardiff, 2010.

G. Fortino, A. Garro, S. Mascillaro, W. Russo, “Using Event-driven Lightweight DSC-based Agents for MAS Modeling,” *International Journal on Agent Oriented Software Engineering*, 4(2), 2010.

G. Fortino, W. Russo, E. Zimeo, “A statecharts-based software development process for mobile agents,” *Information and Software Technology* 46(13), 907--921, 2004.

A. Fuggetta, G.P. Picco, and G. Vigna, “Understanding Code Mobility”, *IEEE Trans. on Software Engineering*, 24(5), pp. 342-361, 1998.

S. Funfrocken, “Transparent Migration of Java-based Mobile Agents: capturing and re-establishing the state of Java programs,” *Proc. of the 2nd Int’l Workshop on Mobile Agents*, K. Rothermel and F. Hohl (eds), LNCS 1477, pp. 26-37, Sept. 1998, pp. 26-37.

M. Hohlfeld and B. Yee, “How to migrate agents,” available at http://www.cs.ucsd.edu/~bsy, 1998.

Y. Ji, H. Jiang, and V. Chaudhary,. “Adaptation point analysis for computation migration/checkpointing,” *Proc. of the 2005 ACM Symposium on Applied Computing*, Santa Fe, NM, Mar 13 - 17, 2005.

R. Quitadamo, L. Leonardi, G. Cabri, “Leveraging strong agent mobility for Aglets with the Mobile JikesRVM framework”, *Scalable Computing: Practice and Experience*, 7 (4), December 2006.

T. Sakamoto, T. Sekiguchi, and A. Yonezawa, “Bytecode Transformation for Portable Thread Migration in Java,” *Proc. of the 4th Int’l Symp on Mobile Agents*, Zurich, Switzerland, Sept. 2000.

T. Sekiguchi, H. Masuhara, and A. Yonezawa, “A Simple Extension of Java Language for Controllable Transparent Migration and its Portable Implementation,” Coordination Languages and Models, LNCS 1594, Apr. 1999.

N. Suri, J.M. Bradshaw, M.R. Breedy, P.T. Groth, G.A. Hill, and R. Jeffers, “Strong mobility and fine-grained resource control in Nomads,” *Proc. of the 4th Int’l Symp on Mobile Agents*, pp. 79-92, Zurich, Sept. 2000.

M. Zhang and W. Li, “Persisting Autonoumous Workflow for Mobile Agents Using Mobile Thread Programing Model,” LNAI 1733, pp. 84-93, 2002.

C. Wicke, L.F. Bic, M.B. Dillencourt, and M. Fukuda, “Automatic State Capture of self-migrating computations in MESSENGERS,” Proc. of the 2nd Int’l Workshop on Mobile Agents, LNCS 1477, Springer-Verlag, pp. 68-79, 1998.

M. Wooldridge, N. Jennings, “Intelligent agents: theory and practice,” The Knowledge Engineering Review, 10(2), 115-152, 1995.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Reorganization in Massive Multiagent Systems

Henry Hexmoor
Department of Computer Science,
Southern Illinois University,
Carbondale, IL 62901, USA
Email: {hexmoor}@cs.siu.edu

***Abstract*—**We have explored principled mechanisms for converting a hierarchical organization to an edge type organization. Other than structural differences, organizations differ in information flow network and information sharing strategies. Beyond current effort, many other types of organizational adaptation are possible and require much further research that we anticipate to remain for future work. This article lays the foundation for automatic organizational adaptation.

I. Introduction

Hierarchical command and control (C2) structures are ineffective (Alberts and Hayes, 2003). *Power to the Edge* (PE) is an information and organization management philosophy that has superseded traditional organizational paradigms for C2. PE provides empowerment for edge members, as well as superior, interoperablem agile, shared awareness among all the members in the organization. PE provides adaptability in dynamic situations [6]. Man on the Loop (MOTL) complements PE with a set of computational tools for systematically altering organizational components for agile response to dynamics in large population multiagent systems [21, 25]. Reorganization is necessary to allow communities of agents or robots to reconfigure their organizational structure in agile response to changes in the environment.

We have explored principled mechanisms for converting a hierarchical organization to an edge type organization. Other than structural differences, organizations differ in information flow network and information sharing strategies. Many types of organizational adaptation are possible and require in-depth research that we anticipate to remain for our future work. This article lays the foundation for automatic organizational adaptation. We begin by outlining related work and background in section 2. In section 3 we present an approach to reorganization. Section 4 describes an implementation of a simulated testbed that will help us validate our salient concepts. Preliminary results and conclusions are in sections 5 and 6 respectively.

II. Backround

In highly dynamic operational environments, it is desirable for multi-agent systems (MAS) to be capable of self-directed structural reorganization. The organizational structure of MAS, whether based on a graph, hierarchy, federation, or other form, dictates the communication interactions between agents as well as the distribution of roles and authority throughout the system. When motivated to adapt, agents should do so without human intervention and in a manner that improves the overall performance of the system. Several motivating factors for reorganization will be discussed here as will current proposed methods for performing dynamic reorganization in MAS. Current research in the area of dynamic reorganization in multi-agent systems has yielded a few approaches in dealing with the problem of adaptation in uncertain and often hostile environments. Given the intelligent and autonomous nature of agents in MAS, individual agents must be capable of locally adapting their interconnection schemes with respect to other agents in the MAS. In addition, agents must be able to accept new roles and to comply with any restrictions or laws associated with these roles. Limitations of communication bandwidth, imposed by the environment or power considerations, or explicitly mandated due to security concerns, implies that agents may not, and probably will not, have a complete picture of the effectiveness and efficiency of the global system. With these limitations in mind, agents must adapt based on local perception of global performance [17].

The impetus directing reorganization varies widely. Common adaptation triggers are based on estimates of the overall performance of the MAS, timelines specifying that reorganization should take place at scheduled intervals, or structural requirements. Matson and DeLoach have described other adaptation triggers related to roles and contrast adaptation for timeline-based efficiency and for quality-based effectiveness [24]. They illustrate three scenarios resulting in a need for reorganization. The first of these relates to the situation in which the organizational objective demands a role that has not yet been assigned. In this situation, there may or may not be extra agents available to accept the required role. Second, a role that is currently assumed by an agent may be relinquished by that agent, resulting in incomplete role distribution as with the first scenario. Third, an agent may be forced to relinquish a role due to some internal fault or as a result of malicious activity. In this case, the system may not be informed of the need to reassign the lost role. similar triggers are described in [14]. These include allocation, reallocation, and exchange. In an allocation scenario, an agent has completed its task and is allocated a new task. In the reallocation scenario, an agent prematurely terminates its current task and is allocated a new one. In the final scenario, ex-

This work was supported by AFRL under contract FA8750-06-C-0138.

change, two agents swap tasks. Regardless of the adaptation trigger driving reorganization of a MAS, the desired outcome remains efficient completion of the system objective. A system of reorganization based on dynamic capability evaluation is presented in [26].

Matson and DeLoach's approach to reorganization of MAS first involves the evaluation of the system's ability to perform a desired task. Based on this evaluation, agents may decide to either proceed to satisfy the organizational goals, relax some goals, or abandon the process of reorganization and task acceptance altogether. The foundation of this approach is an organizational model consisting of goals, roles, agents, and capabilities. Based on this model, certain evaluative constraints are applied to the process. First, there must exist knowledge of which agents are available for inclusion in the system. Second, it must be determined what necessary capabilities exist in order to satisfy the demands of a role. Third, an assessment of the capabilities of all available agents must be made to determine their respective qualifications for acceptance of a given role. To perform this step, the authors have devised a capability taxonomy rooted at the abstract level. Leaf nodes of this taxonomy represent concrete functions and capabilities of an agent, such as the types of sensors (sonar, infrared, etc.) and motivators (wheels, tracks, etc.) the agent is equipped with. Finally, limitations applied to roles must be taken into consideration.

Considering these constraints, Matson and DeLoach formulated a six step evaluation process, which begins with the broad definition of system goals. Following this, the broad goals must be reduced into a simpler, structured format. Using this structured form of the system goals, the process determines all roles which will be required to complete the prescribed objectives. A *general purpose* method has been developed and discussed in [31]. This method is based on the organizational model for adaptive computational systems (OMACS) platform and has been shown to result in optimal network configurations. Another method for performing dynamic organization and reorganization is based on the principle of referral networks. [30] that describe such networks. In referral networks, agents make and sever connections with other agents in the system through the analysis of referrals provided by neighboring agents. An agent wishing to enter the network or alter its set of interconnections once within the network accepts referrals from surrounding agents. From these referrals, agents can form opinions regarding the quality of service provided by other agents and their respective trustworthiness. These are referred to as an agent's expertise and sociability [30].

Agents in referral networks use their knowledge of the trustworthiness and sociability of other agents in the system to decide which agents with which to sever communications or with which to add communication links. Agents that possess high trustworthiness and sociability rankings attract more agents. As the highly trusted and sociable agent gains connections with other agents, its degree increases and it therefore has a greater chance of being referred to other agents in the system. This, in turn, leads to a clustering of agents around the ones seen as the most fit. Fitness directly relates to trustworthiness and sociability. Clustering of this kind is linked closely to the concept of preferential attachment [5] and has been identified in many real-world networks, especially in the Internet. Referral networks are classical and useful, but other methods of reorganization exist. Some of these methods attempt to model biological and chemical organization methods. One such method, related to the concept of stigmergy, is referred to as the digital hormone model [27]. This model is based on the understanding that hormonal signals are used often in nature to form organizations of high complexity. In the digital hormone model (as it relates to agents or robots), agents emit activator or inhibitor signals, i.e. hormones, into their surroundings. Once diffused into neighboring agent regions, the agents in these regions combine the incoming hormone strengths with those already present in their area and adapt their behavior based on these recalculated hormone strengths. The actual reorganization process in the DHM requires four steps, which are repeated continuously, assumedly until some goal has been reached. These steps begin with agents assuming roles based on their abilities and associated rules which govern their behavior. Next, execution of roles takes place. This is followed by each agent transmitting and receiving digital hormones to and from their surrounding areas. The final step in this process involves updating each agent's view of the concentration of hormones in its surrounding area.

Some considerations to keep in mind when selecting an adaptation mechanism or creating a new one are the learning rate, stability, and global structure of the MAS formed by the mechanism [17]. Other considerations are given in [18]. In this, the authors present the question of which agents should be allowed to adapt in the event of a failure. Three possibilities are considered and include random agent sets, a single agent in the event of a team failure, and all neighboring agents in the event of a single node failure. Furthermore, the authors propose a candidate pool of available agents with which an adapting agent may establish a connection. These are limited to the set of all agents, ex-teammates of the adapting agent, or referred agents, as are used in referral networks. This is continues in [18] with this work by outlining a process by which agents adapt given the above noted constraints. The process begins with the construction of the candidate pool and proceeds with several filtering stages. Structural filtering and skill filtering are performed first, followed by degree filtering in which only candidates with the single highest degree are left to connect with the adapting agent.

A general overview of dynamic reorganization concepts and examines two metrics useful in examining MAS performance; society utility and agent utility is found in [16]. Society utility is further decomposed into the success of interactions, roles, and structures in the system. Agent utility is not clearly defined, as it differs from agent to agent in heterogeneous agent systems. In addition to these utility metrics, [15] classifies several types of reorganization "maneuvers." The first of these, pre-emptive reorganization, is a viable option in unpredictable environments where possible, or likely, events can be prepared for in order to take full advantage of them. Protective reorganization attempts not to take advantage of possible future events, but instead works to limit the negative effects of such events on the system. Exploitive reorganization takes place after the fact, and seeks to benefit from events that have already taken place. Finally, corrective reorganization attempts to lessen the damage caused by events which have previously occurred in order to maintain system usefulness. Specific methods for performing adaptation are not present in [16], but it provides many useful ideas for developing new methods or for elaborating on existing methods.

III. An Organizational Model

Next, we define salient attributes for a computational organization. We will begin by defining a set of parameters that characterize an organization. First, we will define capabilities.

1: A *capability* is basic agent ability with a degree in the range from 0.0 to 1.0. We will denote degree of a capability c with D(c).

We assume that there is no decay in capability and agents can only increase their capability. Furthermore, we assume capabilities are mutually exclusive. Let C denote a set of capabilities, which are required in the system for performing all tasks. I.e., C = $\{c_1,c_2,\ldots,c_n\}$. C is the set of all capabilities known by all agents. Each agent will possess each capability c_i to a different degree and may improve it by learning. This provides us with an n dimensional space of capabilities. Let's call this a C-space [21]. Next, we will define roles.

2: An action *role,* denoted with r, is a point in C-space that specifies a minimal capability profile to qualify an agent for the role.

For example, with two capabilities c1 and c2, <0.1, 0.5> is a role that an agent may adopt if it's capabilities c1 and c2 exceed 0.1 and 0,5 respectively [10].

To execute an action role, an agent senses its environment, picks the best rule to determine an action, and performs the action. Success or failure of actions performed are determined in the environment and not known a-priori. At best, an agent may determine a probability of success based on its role fitness to perform it. We'll call the rate of an agent's success its productivity.

Productivity is one of three components for an agent to determine its utility with respect to a goal. The second component is its synergy with others in that role. The third component is the level of fitness of an agent to the role. Fitness of an agent is the sum of ratios of its capability over required capabilities. We are now ready to define an agent's utility with respect to a role.

$$\text{Productivity} = \text{preference (A)} * \text{fitness (A, R)} \qquad (1)$$

3: The *utility* of an agent A, performing in a role R_i, denoted by $u(A, R_i)$, is a linear combination of its *productivity,* its *synergy* levels for that role, and its fitness, i.e.,

$$u(A, R) = P(A, R_i) + [(1/\text{sizeof}(R_i)) * \sum s(i, j)] \qquad (2)$$

Utility as defined here is intended to denote the relative satisfaction of an agent with a role. If this value is sufficiently high, an agent will be content with its current role. However, if this utility value is low, the agent will be inclined to change its role. As we will discuss norms later in this report, using individual utility as a basis for role change assumes a *selfish norm*. In general, norms are prescribed by a user, which in turn will affect agent decision making. So far, we've considered role changes motivated by utilities alone. However, a major intuitive motivator for role exchange is *opportunity*. In general, opportunity is determined by analyzing environmental attributes that suggest the degree to which adoption of a role by an agent will contribute to system or individual productivities. As an example, a midfielder may see the ball near the opponent's goal and determine that it has a good chance of scoring if it played a forward. Real world computations like this are very rapid and continuous, [19]. Opportunistic computing has also entered technical domains [28].

Agents need to continually, mentally quantify potential margins of system or individual productivity gains against all possible roles they could adopt. Whichever role candidate will yield the highest marginal gain will be the next role the agent will wish to take. The agent's choice of next role is a proposal that need to be presented to the organization and once processed; the agent may proceed with adoption of it.

For a vacant role, two temporal constraints of the role augment the notion of opportunity. The first is the immediacy of the need to occupy the role. Each open role will specify the urgency of the role. For instance, the team captain will assign a temporal urgency for each vacant role to be filled. Agents who vie for a vacant position, must meet the urgency constraint. Agents must dynamically compute their capability to transition into an open role. Naturally, this capability differs from an agent's innate abilities to perform action. It is not a personality trait. Also, it is not a universal agility or flexibility to take on roles in general. This capability differs with respect to each role and depends on the environmental circumstances. The second temporal constraint is the duration after which the role becomes obsolete and there will not be a need to occupy it. If the validity time window ends, we say such a role is expired. Agents must account for this constraint and should only consider a role if it is not yet expired.

In contrast to an action role, a role that is decision oriented is a manager role. An example of a manager role is captain of a football team.

4: *Rank* of a role r_i assigns a number to the role that reflects its relative importance in an institution. This is denoted by $\text{Rank}(r_i)$.

This is a highly simplified model of a role's valuation in an institution. Using this we introduce a notion of role order. The function "Rank" may return any natural number. The smaller the number the more preferred the role. Role r_j is the most preferred rank if $\text{Rank}(r_j) = 1$. Importance of a role is inversely proportional to its rank.

5: *Role Ordering* (RO) is an ordering of action roles. Each role is assigned a unique rank. I.e., <Rank (r_1) , Rank (r_2) , ….. , Rank (r_n) > specifies role ordering where $\text{Rank}(r_i)$ is the i^{th} position is the rank for i^{th} role.

If Rank(r_i)< Rank(r_j) then role corresponding r_i is preferred over the role r_j that has a smaller rank. RO sets up a trajectory in C-space.

6: A norm is a convention in the form of a rule shared by all individuals. It may govern role adoption with a set of rules. We denoted the set of norms by N.

For simplicity, we consider organizational norms to be mutually exclusive and non-overlapping. In this paper we will limit norms to rules that govern role change. An example of a norm that governs roles is that individuals incrementally improve their capabilities over time and are allowed to apply for a higher rank in the organization-- this is the *promotion norm*. Another norm that will govern roles is based on utilities. The *selfish* norm will only consider individual utilities whereas *beneficent* will only account for the society's benefits. We assume that norms are determined outside organizations. In our framework, Moe issues a norm.

7: A *department*, denoted by D_i, is a fixed number of agents who are performing the same role R_i. Each department will require a minimum number of individuals needed to occupy roles in that department at a given time. We assume that departments are static and will not change over time. If there is a significant change for a department such as the number of individuals required, a new organization is formed.

Since departments and roles have a one to one correspondence, ranking one prescribes an ordering on the other. We are now ready to define an organization.

8: An Organization is modeled as <C, R, D, N>, which is capabilities, roles, departments, and norms.

When an organization is initially populated by individuals, each agent adopts a role depending on its capability level. Each agent will aspire to occupy a role that is highest ranked in the organization.

After an organization is populated, it will experience reorganization where self-motivated individuals may change roles based on the promotion norm. This is shown in Figure 1.

We assume that individuals dynamically change roles and move to different departments. To capture a snaphot of organizational configuration we define a *state* that will be defined later.

9: At any given time, the configuration of individuals in departments of an organization is the *state* of that department.

States change as often as individuals change their roles. Therefore, there can be many state transitions over time. State transitions are most significantly affected by a norm change.

Thus far, our model of an organization lacks characteristic concepts of agents that will occupy it. In the next section we introduce these notions.

10: A *synergy network* among a group of agents is a graph among agents where the arcs represent a real value between -1.0 and 1.0 indicating negative or positive influence between pairs of agents. We will use *s(i,j)* as an directed function that returns the synergy value between agents i and j. Synergy is not reflexive and not symmetric.

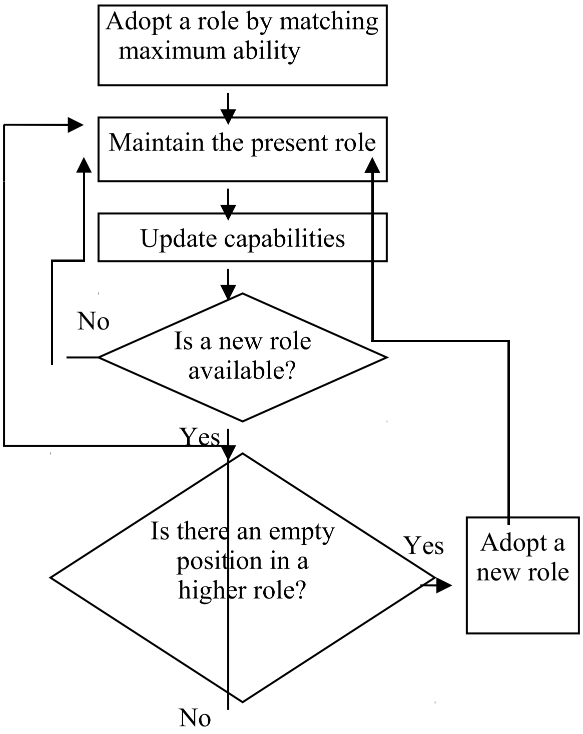

Adopt a role by matching maximum ability
Maintain the present role
Update capabilities
No
Is a new role available?
Yes
Is there an empty position in a higher role?
Yes
Adopt a new role
No

Figure 1. Flow chart for self-initiated promotion norm

Synergy between an agent agent A and another agent B is a degree of influence from B towards A. This can be positive or negative. We assume authority as a form of synergy. For example a manager role like a captain will have synergy towards players. Synergies change over time. One form of change is experience. Individuals in a department who interact with one another generally develop positive synergies towards one another that is proportional to the duration of time they remain the same department. However, this is not universally true and synergy change goes beyond departmental boundaries. What is important is how one agent's action enables another to perform its action.

Simply modeled, frequency of interactions between agents A and B that leads B to execute its role, increases synergy from A to B. If B is prohibited from executing its action due to A's action, then synergy decreases from A to B.

These changes in synergy cannot be simply determined from the organizational configuration alone. To recapitulate, Synergy changes are dynamic and dependent on interactions.

Beyond individual productiveness, an organization may produce something that is an emergent property and it is not attributed to a single individual. An example is scoring points that is a result of team work. Let's call this system productivity. Similar to an individual, we model a system utility that is a measure of the relative satisfaction of the organization.

11: *Organizational utility* is the sum of individual productivities plus balance of synergies among individuals in the organization.

12: A state of an organization is the combination of current, active norm and profile of roles occupying departments of the organization, i.e., composition of roles.

Since there are many possible role compositions and there are several possible norms, an organization can be in numerous states. A change in organizational role composition or a change in active norm will yield a state change. States of the system are completely observable and changes in states are under our control. Since state changes are not dependent on the sequences of past states, we can make the Markovian assumption. The nature of organizational state space fits a Markov decision process MDP. There are well known methods for solving MDPs such as the value iteration method [2].

IV. Implementation

In order to illustrate the implementation of reorganization, we have designed a model that will be implemented using Netlogo. Netlogo is a java based cross-platform multi-agent programmable modeling environment for simulating natural and social phenomenon. It is freely available online at http://ccl.northwestern.edu/netlogo/.

The Domain which we will use is a simplified version of the game of Soccer (i.e., see "http://en.wikipedia.org/wiki/Football_(soccer)".

The game in our simulation is designed in the following fashion.

Capabilities: In this simulation we assume each player has a set of capabilities associated with him, which may or may not be unique. The capabilities are variables that are subject to change with the progress of the game. In our game each player from the start of the game maintains his capabilities at the same level. The set of capabilities each player has are threefold: Speed, Accuracy with handling the ball, Ability to kick the ball (far/near).

Speed: This represents the speed with which a player moves across a field. His ability to chase the ball and move the ball along.

Accuracy: This sets the players accuracy in handling the ball in the field. His ability to tackle with the ball when the ball is within his reach is decided by this parameter. Accuracy of a player ranges between 20^0-100^0.

Kick: This ability of kicking a ball determines the distance a ball moves when a player kicks it.

The game is divided into a set of roles where each player can play only one role at a time. These are forward, Defender, Mid-fielder, and goal keeper. We omit details for brevity and space constraints.

Productivity: The productivity of each player is calculated according to the formula mention in the approach section the report although it had been adopted to each department to meet the requirement of the requirement of the department. The basis of calculation of productivity is mentioned in the above paragraphs describing the features of the department. The productivity of individual player is calculated and the productivity of the department is calculated basing on that. The team productivity is calculated considering the department productivities and the performance of the team as a whole. The productivities of the team are monitored in the graph in the user interface of the simulation.

Utility: the utility of each individual player performing in the designated role is calculated according to the earlier mentioned formula. The individual utility is how effective each person is performing in the designated role and the task at hand. We considered a randomly generated synergy while calculating the individual utilities. The department utility is calculated considering how effective those assigned individuals are at performing tasks and promoting the interests of the department. The utility of the team as whole has been calculated by considering the effectiveness of the departments performance in promoting the interests of the team , the synergies generated between the departments of the team and the overall performance of the team.

Norm: We implemented norm in this simulation. The norm in this game acts as the motivation or attitude with which the players play the game

We designed three norms, each of it differs from the others in unique way.

The norms implemented are Attack, Defend, Self-Gain. In the game the end user or MOTL gives the norm which governs the game.

Norms and reorganization:

Attack Norm: In this norm the players are expected to play with attack mode on their mind. The game is modifies such a way that there are more number of players in the forward department, four instead of three. With more number of players in the forward the game tends to be more aggressive at the cost of weakening the mid-field and defense departments. This is more offensive approach. The reorganization among the players is mainly motivated by the attack norm. The game starts normally and after a period of 500 clock cycles the system checks for any other player who has more capabilities to play in the forward. If it does not implicitly find one it calls for a reorganization and reassigns the roles to the players. The system does not look for best or worst performance it just does the reorganization until Moe is satisfied with it. Only Moe can instruct a system to stop reorganization. So Moe is the one who controls the extent to which she can permit reorganization.

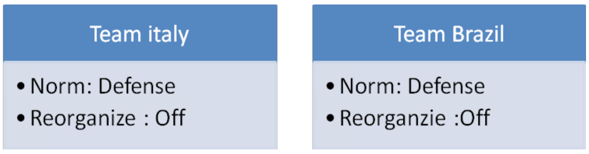
Defend Norm: In this norm the players are expected to play in a defensive mode. The defense department has four players instead of regular three. With more players in the defense the game tends to be more defensive at the cost of loosing aggression; i.e., by weakening the mid-field we also reduce the support to the forward department. After a period of 500 clock cycles, the system checks for players who can be more effective to play in the defense department, and calls for a reorganization. The system this way keeps reorganizing until it receives any instruction from Moe.

Selfish Norm: With this norm player's play in the regular department layout configurations. Instead of caring for the team, they play to maximize their productivity and improve their utility. When reorganization is initiated, the system checks if a player is suitable for the role she is playing. If the system considers the player can perform better in another role it calls for reorganization. This is performed by computing the average productivity of the department. If the player falls below the department average she is underperforming so she is replaced. The system maintains reorganizing until Moe aborts it.

Synergy: In this simulation, we take the average of the department's average productivity and compare it with the individual synergy. If player's productivity is more than the departments' productivity we assume she has positive synergy. If she is performing below his capability we assume that she has negative synergy. Each player in the game has a synergy with respective to their department.

V. Results

We consider two cases for exhibiting results of using our simulation
Case 1:



Team italy
- Norm: Defense
- Reorganize : Off

Team Brazil
- Norm: Defense
- Reorganzie :Off

Figure 4.

In the Case 1, we allow the simulation run for a time period of 11400 clock cycles.

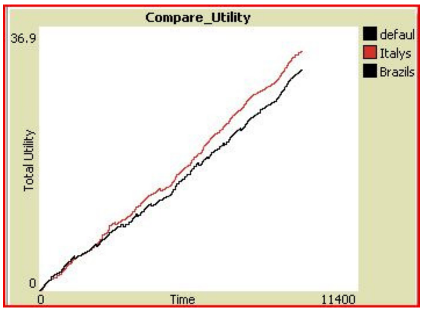
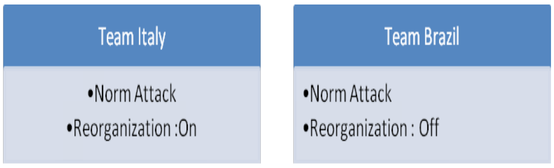


Figure 5.

The resultant graph in figure 5 compares utilities between the respective teams. Both teams norms were set to "Defense".

Case 2:



Team Italy
- Norm Attack
- Reorganization :On

Team Brazil
- Norm Attack
- Reorganization : Off

Figure 6.

In this case we set the norm for both teams in "attack" mode. Reorganization switch is turned on for Italy and it is turned off for Brazil. There are three players in Defense and Mid-Field department and there are four players playing on the forward department. The resultant graph of utilities for teams at a time cycle of 11400 clock cycles is shown in Figure 7.

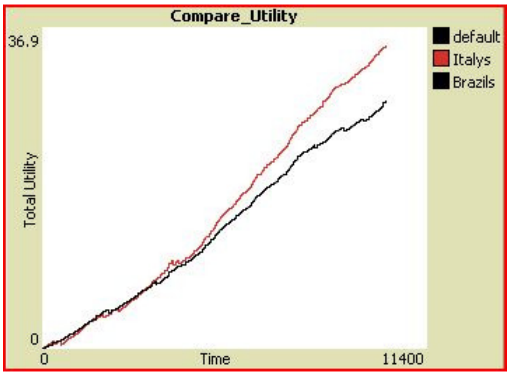


Figure 7.

At the time when the readings are noted the graph shows, Team Italy's utility 36.9 and Team Brazil's utility at 31.2. Due to the implementation of the reorganization algorithm the Team Italy has a considerable advantage over the opponents.

VI. Conclusions

We have made novel strides in managing massive agent organizations that contributes to development of our man on the loop paradigm. Our methodology allows for Moe to prescribe normative patterns of behavior, which in turn guide the reorganization process. Individuals most fit to play their current roles stay while others are directed to surroundings that augment their synergy. We have demonstrated with the popular game of simulated soccer but the results are generic and transfer to other domains.

References

[1] S. Abdullah and V,. Lesser, *"Multiagent reinforcement Learning and Self-Organization in Network Agent"* In Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems. 2007, pp. 172-179.

[2] M. Aicardi, F. Davoli, and R. Minciardi. *Decentralized optimal control of markov chains with a common past information set.* IEEE Transactions on Automatic Control, AC-32:1028–1031, 1987.

[3] D. S. Alberts, and R. Hayes, *"Understanding command and control"*, Command and Control Research Program (CCRP), available online at: www.dodccrp.org., 2006.

[4] M. Avvenuti, P. Corsini, P. Masci, and A. Vecchio, *"Opportunistic computing for wireless sensor networks"*, Proc. IEEE mobile ad hoc networking conference, 2007.

[5] A. Barabási, and R. Albert, *"Emergence of Scaling in Randiom Networks"*, Science 286: pp. 509–512, 1999.

[6] K. Chang, *"The Performance of Edge Organizations in a Collaborative Task"*, MS thesis, The Naval Postgraduate School, Monterey, CA., 2005

[7] H. Hexmoor and S. Pasupuletti, *"Institutional versus Interpersonal Influences on Role Adoption"*, In AAMAS workshop: Representation and approaches for time-critical resource/role/task allocation, J. Modi and T. Wagner (Eds), Melbourne, Australia, 2003.

[8] M. Gladwell, *"The Tipping Point: How Little Things Can Make a Big Difference",* Back Bay Books*, 2002.*

[9] E. Matson and S. DeLoach, *"Using Dynamic Capability Evaluation to Organize a Team of Cooperative, Autonomous Robots",* Proc. 2003 International Conference on Artificial Intelligence (IC-AI '03), 2003.

[10] A. Rahman, and H. Hexmoor, 2004. *"Negotiation to improve Role Adoption in Organizations",* In Proc. International Conference on Artificial Intelligence (IC-AI), pp. 476-480, CSREA Press, 2004.

[11] R. Albert, and A. Barabasi, *"Statistical Mechanics of Complex Networks."* In Review of Modern Physics (p. 74), 2002.

[12] K. Barber, and C. Martin, *"Dynamic Reorganization of decision making groups. Proceedings of 5th autonomous agents",* 2001.

[13] K. Carley, K., and L. Gasser, *"Computational organizational theory. Multiagent systems : A modern approach to distributed artificial intelligence",* pp. 299-300, 1999.

[14] L. Chaimowicz, M. Campos, and R. Kumar, *"Dynamic Role Assignment for Cooperative Robots",* In IEEE Conference on Robotics and Automation , 2002.

[15] V. Dignum, *"A Model for organizational interactions based on Agents, founded in Logic",* SIKS Dissertion Series 2004-1. 2004: Utrecht University , 2004.

[16] V. Dignum, V. Furtado, F. Dignum, and A. Melo, *"Towards a Simulation Tool for Evaluating Dynamic Reorganization of Agent Societies",* 2005.

[17] M. Gatson, M., and M. desJardins, *"Agent Organized Networks for Dynamic Team Formation",* In AAMAS, 2005.

[18] M. Gatson, J. Simmons, and M. desJardins, Adapting *"Network Structure for Efficient Team Formation",* In AAMAS Workshop on Learning and Evolution in Agent Based Systems , 2004.

[19] N. Glasser, and P. Marignot, *"The Reorganization of societies of autonomous agents",* In MAAMAW, pp. 98-111, 1997.

[20] H. Handley, and A. Levis, *"A Model to evaluate the effect of organizational adaption",* In H. Handley, and A. Levis., Computational and Mathematical Organization Theory 7 (pp. 5-44) Kluwer Academic. 2001.

[21] H. Hexmoor, B. McLaughlan, G. Tuli, *"Natural Human Role in Supervising Complex Control Systems",* In Journal of Experimental and Theoretical Artificial Intelligence, Taylor and Francis, pp. 59-77. 2008.

[22] K. Chang, P. Lehner, A. Levis, A. Zaidi, and Z. Zhao, *"On Causal Influence Logic".* Technical report,George Mason university Centre for Excellence, 1994.

[23] P. Kazakos, and A. Zaidi, *"An Algorithm for activation Timed Influence Nets",* In IEEE IRI 2008, July 13-15, 2008.

[24] E. Matson, and S. DeLoach, *"An Organizational Model for Designing Adaptive Multiagent Systems",* In AAAI Workshop: Agent Organization, 2004.

[25] B. McLaughlan, and H. Hexmoor, *"Influencing Massive Multi-agent Systems via Viral Trait Spreading",* In Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems, 2009.

[26] E. Matson, and S. DeLoach, *"Using Dynamic Capability Evaluation to Organize a Team of Cooperative Autonomous Robots",* Proc. the 2003 International Conference on AI , 2003.

[27] C. Shen, C. Choung, and P. Will, *"Simulating Self-Organization for Multi-Robor Systems",* In IEEE/RSJ International Conference on Intelligent Robots and Systems. 2002.

[28] G. Valetto, G. Kaise, and G. Kc. *"A moblie agent approach to process-based dynamic adaption of complex software systems",* In 8th European Workshop on Software process technology, pp. 102-116. 2001.

[29] P. Yolum. And M. Singh, *"Emergent Personalized Communities in Refferal Networks",* IJCAI workshop on Intelligent Techniques for Web Personalization , 2003.

[30] Zaidi, and P. Papatoni-Kazakas, *"Modelling with Influence Networks Using Influence Constants: A New Approach",* Proc. 2007 IEEE/SMC International Conference on Systems, Man, and Cybernetics, Montreal, Canada, 2007.

[31] A. Zhong and S. DeLoach, *"An Investigation of Reorganization Algorithms",* In International Conference on Artifiial Inteligence, 2006.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Effectiveness of Solving Traveling Salesman Problem Using Ant Colony Optimization on Distributed Multi-Agent Middleware

Sorin Ilie and Costin Bădică
University of Craiova, Software Engineering Department
Bvd.Decebal 107, Craiova, 200440, Romania
Email: sorin.ilie@software.ucv.ro,costin.badica@software.ucv.ro

Abstract—**Recently we have setup the goal of investigating new truly distributed forms of Ant Colony Optimization. We proposed a new distributed approach for Ant Colony Optimization (ACO) algorithms called Ant Colony Optimization on a Distributed Architecture (ACODA). ACODA was designed to allow efficient implementation of ACO algorithms on state-of-the art distributed multi-agent middleware. In this paper we present experimental results that support the feasibility of ACODA by considering a distributed version of the Ant Colony System (ACS). In particular we show the effectiveness of this approach for solving Traveling Salesperson Problem by comparing experimental results of ACODA versions of distributed ACS with distributed random searches on a high-speed cluster network.**

I. Introduction

As NATURAL phenomena are inherently distributed, we think that nature-inspired computing should allow a straightforward mapping onto existing distributed architectures. Therefore, to take advantage of the full potential of nature inspired computational approaches, we have setup the goal of investigating new distributed forms of Ant Colony Optimization (ACO hereafter) using state-of-the-art multi-agent technology.

In our recent work [1] we proposed a scalable multi-agent system architecture called ACODA (Ant Colony Optimization on a Distributed Architecture) that allows the implementation of ACO in a parallel, asynchronous and decentralized environment. The novelty of our approach is: (i) the problem environment is conceptualized and implemented as a distributed multi-agent system ([2], [3]) and (ii) ant management is reduced to messages exchanged asynchronously between the agents of the problem environment.

Existing computational approaches of ACO [4] are based on sequential algorithms, are highly synchronous and require use of global knowledge. While few parallel and distributed versions of ACO exist [5], they are mainly based on their sequential counterparts, thus hindering the potential gain through parallelization. For example, in [5] the authors propose a parallel, distributed, asynchronous and decentralized implementation of ACO. However, their approach requires maintenance of the globally best solution currently known using global update each time a better solution is found. Moreover, the authors' claim that "their implementation does not effect the accuracy, speed and reliability of the algorithm" is not supported by any experimental evidence.

The focus of this paper is to experimentally evaluate the feasibility of ACODA by considering a distributed version of ACO inspired by the Ant Colony System (ACS) [4]. In particular, we show the effectiveness of this approach at solving Traveling Salesperson Problem (TSP hereafter) by comparing experimental results of ACODA versions of distributed ACS with distributed random searches on a high-speed cluster network. The results clearly show that our distributed version of ACS based on ACODA preserves the nice heuristic properties of standard ACS, while also providing scalability by exploiting distributed computing architectures – multi-agent middleware in this case.

The paper is organized as follows. In Section II we present some background on ACO and distributed approach and we briefly review the ACS model that inspired our initial experiments. There are however notable differences between classic ACS and our distributed version based on ACODA (see Section V). In Section III we introduce ACODA architecture and underlying search algorithm. Section IV presents experimental results that support the effectiveness of our approach by comparing results obtained with running on ACODA our distributed version of ACS with other three distributed search methods. Section V presents related works, while Section VI presents our conclusions and points to future works.

II. Background

ACO is inspired by behavior of real ants. When ants are searching for food, they secrete pheromone on their way back to the anthill. Other colony members sense the pheromone and become attracted by marked paths; the more pheromone is deposited on a path, the more attractive that path becomes. The pheromone is volatile so it disappears over time. Evaporation erases pheromone on longer paths as well as on paths that are not of interest anymore. However, shorter paths are more quickly refreshed, thus having the chance of being more frequently explored. Intuitively, ants will converge towards the most efficient path, as that path gets the strongest concentration of pheromone. Artificial ants are programmed to mimic the behavior of real ants while searching for food. More details on the ACO metaphor can be found in [4].

In this paper we propose ACODA distributed approach for the implementation of ACO algorithms and show its effectiveness for solving TSP. The goal of TSP is to compute the shortest tour that visits each node of a complete weighted graph exactly once. The decision version of TSP is known to be NP-complete so it is very unlikely that a polynomial solution for solving TSP exists. So TSP is a good candidate for the application of heuristic approaches, including ACO.

The main and also new idea behind ACODA is to provide a multi-agent distributed architecture for modeling the problem environment. One or more anthills are located in this environment. Artificial ants originating from anthills will travel in the environment to find optimal solutions, following ACO rules. In order to approach TSP using ACO, the problem environment is modeled as a distributed set of interconnected graph nodes that are also anthills. Each graph node is modeled as a software agent that can host a population of ants. The ants travel between nodes until they complete a tour. Once they return to their originating anthill, they mark the solution with pheromone by retracing their path. The ants traveling is modeled as messages exchanged by the agents that represent the graph nodes.

Many ACO algorithms have been proposed in the literature. A good survey is [4]. While with ACODA we aim at proposing a general distributed framework based on agent middleware for different ACO algorithms, the ACO model considered in this paper is based on a particular version of ACO – the ACS system [4]. ACS is a sequential implementation of ACO that chooses to move ants in parallel instead of moving each ant until it finishes its tour. Our approach presents a few differences due to the ACODA requirements: (i) distributed architecture based on asynchronous message passing and (ii) avoid to use global knowledge.

ACO rules determine the amount of pheromone deposited on edges, the edge chosen by each ant on its way, and how fast the pheromone deposited on each edge evaporates. For this purpose we use the mathematical model of ACO that is used in ACS.

In ACS, ant k located at node i decides to move to node j using "pseudo random proportional rule" (1). Equation (1) chooses to directs the ant either to a completely random node or to a node of high desirability, and this decision is taken probabilistically.

$$j = \begin{cases} argmax_{l \in N_i}((\tau_{i,l})^{\alpha}(\eta_{i,l})^{\beta}), & \text{if } q \leq q_0 \\ J, & \text{otherwise} \end{cases} \tag{1}$$

where:

- α is a parameter to control the influence of $\tau_{i,j}$
- $\tau_{i,j}$ is the amount of pheromone deposited on edge (i, j)
- $\eta_{i,j}$ is the desirability of edge (i, j) computed as the inverse of the edge weight, i.e. $1/w_{i,j}$
- β is a parameter to control the influence of $\eta_{i,j}$
- q is a random variable uniformly distributed in $[0, 1]$
- q_0 such that $0 \leq q_0 \leq 1$ is a parameter that controls the selection between a random neighbor and most promising neighbor based on pheromone deposit and edge desirability
- J is a random node selected according to the probability distribution given by equation (2)
- N_i represents the set of neighbors of node i

An ant located in node i will randomly choose to move to node j with the probability $p_{i,j}$ computed as follows:

$$p_{i,j} = \frac{(\tau_{i,j})^{\alpha}(\eta_{i,j})^{\beta}}{\Sigma_j(\tau_{i,j})^{\alpha}(\eta_{i,j})^{\beta}} \tag{2}$$

where:

- α is a parameter to control the influence of $\tau_{i,j}$
- β is a parameter to control the influence of $\eta_{i,j}$
- j is a node reachable from node i that was not visited yet

Following equation (1), the ant makes the best possible move (as indicated by the learned pheromone trails and the heuristic information, i.e. the ant is exploiting the learned knowledge) with probability q_0, while it performs a biased exploration of the arcs with probability $(1 - q_0)$.

Better solutions need to be marked with more pheromone. So whenever an ant k determines a new tour V_k of cost L_k the ant will increase pheromone strength on each edge of the tour with a value that is inversely proportional to the cost of the tour.

$$\Delta\tau^k_{i,j} = \begin{cases} 1/L_k & \text{if edge } (i, j) \text{ belongs to found tour } V_k \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

When an ant travels along a given path, this traveling takes an amount of time that is proportional with the travel distance (assuming the ants move with constant speed). As pheromone is volatile, if a real ant travels more, pheromone will have more time to evaporate, thus favoring better solutions to be discovered in the future. We conclude that adding pheromone evaporation to our model can be useful, especially for solving a complex problem like TSP.

When an ant completes a tour it will retrace its steps marking the edges on the way with pheromone. The update will also take into account pheromone evaporation. Assuming an evaporation rate $0 \leq \rho < 1$, evaporation and pheromone update are implemented in ACS as follows:

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \rho\Delta\tau^k_{i,j} \tag{4}$$

Ants use equation (1) to probabilistically determine their next step. Therefore they will often choose the edge with the highest pheromone, while the exploration of less probable edges is low. This behavior can be compensated by decreasing the pheromone on edges chosen by ants using a local pheromone evaporation process. This has the effect of making them less desirable, increasing the exploration of the edges that have not been picked yet. Assuming that $0 \leq \xi < 1$ is the local evaporation rate and τ_0 is the initial amount of

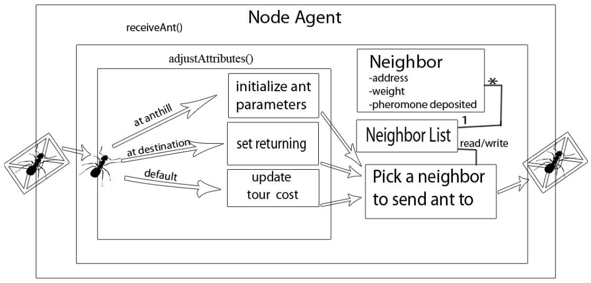


Fig. 1: Node structure in ACODA.

pheromone on each edge, whenever an ant traverses an edge it applies local evaporation by updating pheromone as follows:

$$\tau_{i,j} = (1-\xi)\tau_{i,j} + \xi\tau_0 \tag{5}$$

A good heuristics to initialize pheromone trails is to set them to a value slightly higher than the expected amount of pheromone deposited by the ants in one tour; a rough estimate of this value can be obtained by setting $\tau_0 = 1/(nC)$, where n is the number of nodes, and C is the tour cost generated by a reasonable tour approximation procedure [4]. For example we can set $C = nw_{avg}$ where w_{avg} is the average edge cost.

In order to observe the impact that evaporation has on the solutions, we have also considered a pheromone update scheme that does not include evaporation at all. In this case equations (5) and (4) are replaced with:

$$\tau_{i,j} = \tau_{i,j} + \Delta\tau^k_{i,j} \tag{6}$$

III. Architecture

In ACODA, the nodes of the graph are conceptualized and implemented as software agents [2]. For the purpose of this work, by software agent we understand a software entity that: (i) has its own thread of control and can decide autonomously if and when to perform a given action; (ii) communicates with other agents by asynchronous message passing. Each agent is referenced using its name, also known as agent ID.

The activity carried out by a given agent is represented as a set of behaviors. A behavior is defined as a sequence of primitive actions. Behaviors are executed in parallel using interleaving of actions on the agent's thread with the help of a non-preemptive scheduler, internal to the agent [3].

Node design (see Figure 1) must include a behavior for sending and receiving ants. Whenever an ant is received, the RECEIVE-ANT() behavior (see Table I) immediately prepares it and then sends it out to a neighbor node following ACO rules.

Ants are represented as objects with a set of attributes: cost of the currently search path (which becomes L_k when the ant completes a tour), pheromone strength (value of $\Delta\tau^k_{i,j}$), returning flag, best tour cost (value of the currently best tour that the ant knows, based on its search history) and a list of node IDs representing the path that the ant followed to reach its current location. The list is necessary for two reasons: i) the ant needs to retrace its steps in order to mark the tour with pheromone and ii) we need to avoid loops so only unvisited nodes are taken into account as possible next hops. Attributes are initialized when an ant is created and updated during the process of ant migration to reflect the current knowledge of the ant about the explored environment.

Nodes (represented as software agents) manage a list of neighbor nodes and best tour cost (the value of the currently best tour that the node knows, based on ants that traveled through this node). For each neighbor node we record the weight and the value of deposited pheromone of the corresponding edge. Note that each node and each ant maintain their own values of the best tours they encountered so far. So, whenever an ant is traveling through a node, the ant and the node are able to exchange and update their best tour information accordingly.

In our approach each node creates its own ant population thus becoming an anthill. Nodes calculate pheromone strength according to the tour cost (see equation (3)) and also update the ants' pheromone strength attribute. Additionally, nodes set the returning flag for returning ants, exchange ant information with other nodes, deposit pheromone when needed, and update the cost of the currently search path of an ant.

The structure of a node is presented in Figure 1. RECEIVE-ANT() behavior parses a received ant message, adjusts ant's attributes using ADJUST-ATTRIBUTES() method and sends it out to the address determined by BEST-NEIGHBOR() method. This happens whenever the agent's message queue isn't empty [3]. ADJUST-ATTRIBUTES() method sets returning flag and calculates pheromone strength using equation (3) whenever an ant has completed a tour. Ants that have returned to the anthill are re-initialized.

BEST-NEIGHBOR() uses the RANDOM-CHOICE() method determine the address of the node where to send the ant. When the ant returns to the anthill, the ant is sent to the first node from its list of visited nodes, popping it from the list, and the method DEPOSIT-PHEROMONE() is then called. LOCAL-EVAPORATE-PHEROMONE() together with DEPOSIT-PHEROMONE() implement pheromone update (deposits and evaporation). In order to implement different forms of ACO it is usually sufficient to modify the methods LOCAL-EVAPORATE-PHEROMONE(), DEPOSIT-PHEROMONE() and RANDOM-CHOICE() (see Section IV).

IV. Experiments and Discussions

In order to facilitate experimentation with ACODA, we created a distributed platform based on JADE framework [3] that can be configured to run on a computer network. The focus of the experiments was to show the effectiveness of ACODA to support distributed ACO-based algorithms for solving TSP. We configured ACODA platform on a high-speed cluster network of 7 computers and then we analyzed experimental results that we obtained by running (i) our distributed ACS-based algorithm and (ii) other distributed random search algorithms.

Setup. An experiment is structured as a fixed number of independent experimental rounds. A round consists of one

TABLE I: Algorithms for processing ant information.

```
RECEIVE-ANT()
  RECEIVE($ant$)
  ADJUST-ATTRIBUTES($ant$)
  SEND-TO($ant$,BEST-NEIGHBOR($ant$))

ADJUST-ATTRIBUTES($ant$)
  if AT-ANTHILL($ant$) then
    if RETURNING($ant$) then
      INITIALIZE($ant$)
    else
        SET-RETURN-FLAG($ant$)
        ▷ calculate ant pheromone using equation 3.
        CALCULATE-ANT-PHEROMONE-STRENGTH($ant$)
  UPDATE-BEST-TOUR()

BEST-NEIGHBOR($ant$)
  if RETURNING($ant$) then
    DEPOSIT-PHEROMONE()
    return LAST-VISITED-NODE($ant$)
  $bestNeighbor \leftarrow$ RANDOM-CHOICE()
  UPDATE-CURRENT-PATH-COST($bestNeighbor$)
  LOCAL-EVAPORATE-PHEROMONE()
  ADD-TO-VISITED-LIST($bestNeighbor$,$ant$)
  return $bestNeighbor$
```

execution of an algorithm on ACODA, for a given set of parameters. All parameters are initialized at the start of each round. Experimental data are collected during each round. At the end of the experiment (when the fixed number of rounds is reached) these data are used to calculate performance measures.

ACODA is a distributed platform. Setting-up and running it on several computers assumes two stages: (i) initialization; and (ii) execution.

During the initialization stage: JADE platform is started, a number of containers are created (typically one container for each available machine), and finally node agents are created and evenly distributed on available containers of the JADE platform.

The experiment is ran during the execution stage. Experiment execution is controlled using special control messages. Control messages are distinguished from ant exchange messages using their conversation ID. Command messages have their conversation ID set to "command", while ant exchange messages have their conversation ID set to "ant". Command messages are given higher priority than ant exchanging messages.

An experimental round starts when all node agents receive a "start" command, and ends when all node agents receive a "stop" command. A designated node agent – called *MasterNode* is responsible with issuing commands and calculating performance measures. *MasterNode* counts the total number of ants that it receives. When this number reaches a given maximum value M, *MasterNode* stops the current round (issuing a "stop" command) and starts a new round (issuing a "start" command). Note that *MasterNode* is needed only for evaluation and consequently it is not part of the distributed approach.

Duration T_i of a round i is recorded by *MasterNode* agent as time elapsed between issuing a "start" and "stop" command. While the distributed ACO algorithm is running, minimum tour costs are passed from node to node via ants. Whenever a node agent updates its current value of the minimum tour, the time elapsed since the node received the "start" command is also recorded. When an experimental round is finished, *MasterNode* agent saves the experimental data collected during the round.

We cannot assume that the best tour recorded by *MasterNode* is actually the best tour computed during an experimental round. So we must determine minimum of the best tours computed by all node agents together with the time when this tour was found.

Let us suppose that we have n node agents and k experimental rounds. For each node agent $j \in \{1, 2, \ldots, n\}$, let $t_{i,j}$ be the time of the last update of the best tour performed by node agent j in round $i \in \{1, 2, \ldots, k\}$, and let $v_{i,j}$ be the cost of the corresponding tour. *MasterNode* agent will collect values $v_{i,j}$ and $t_{i,j}$ and will determine the solution v_i and associated time t_i in round i as shown in first two rows of Table II.

The experimental data that were acquired during all the rounds of an experiment are post-processed to calculate performance measures as shown in last five rows of Table II.

Initial performance analysis. We experimented with ACODA on increasingly complex search versions (see Table III) in order to establish its effectiveness for implementing ACO-based algorithms. We considered our ACS-based version of ACO, together with other three distributed search methods – Random Choices, Cost Only, and Pheromone Search. The first two are not ACO-based, while the third is ACO-based.

For the Random Choices (RC) version we replaced equation (1) with randomly choosing the next hop of an ant from the unvisited nodes – i.e. the deposited pheromone and the edge weights are ignored.

For the Cost Only (CO) version we choose the next hop using equation (2), setting $\alpha = 0$ and $\beta = 1$ – i.e pheromone deposits are not taken into account.

Pheromone Search (PS) is an ACO-based search that uses equation (1) to determine the next hop. PS allows all ants to deposit pheromone regardless of tour cost. For this model, the ACO parameters were set to the values recommended by [4] for the ACS algorithm: $\tau_0 = 1/(n^2 w_{avg})$, $\rho = \xi = 0.1$, $\alpha = 1$, $\beta = 5$.

Finally, the ACS-based version of ACODA uses the same parameters as PS, but it allows only the ants that found the best tour so far to deposit pheromone. For all searches, the total number of ants is chosen equal to the number of nodes n (for each node the ant population consists of a single ant) as recommended in [4].

Note that, as we focus on finding the best solution, we chose $q_0 = 0$ in Equation (1), thus avoiding the convergence to a suboptimal solution (this fact is relevant for PS and ACS versions).

We ran several experiments using the following benchmark TSP maps selected from TSPLIB [6]: eil51, st70, kroA100,

TABLE II: Calculation of performance measures.

$v_i = \min_{j=1}^{n} v_{i,j}$	v_i is the solution found by round i.
$t_i = \min_{j=1}^{n}\{t_{i,j} \vert v_{i,j} = v_i\}$	t_i is the time in which solution was found by round i.
$v_{avg} = (\sum_{i=1}^{k} v_i)/k$	v_{avg} is the average value of solutions found in all rounds.
$t_{avg} = (\sum_{i=1}^{k} t_i)/k$	t_{avg} is the average time in which solutions were found in all rounds.
$v_{min} = \min_{i=1}^{k} v_i$	v_{min} is the best solution found after carrying out all rounds.
$t_{min} = \min_{i=1}^{k}\{t_i \vert v_i = v_{min}\}$	t_{min} is the time taken to find the best solution in all rounds for the first time.
$T_{avg} = (\sum_{i=1}^{k} T_i)/k$	T_{avg} is the average execution time of the rounds in an experiment.

TABLE III: Algorithms implemented on ACODA. "nop" means that the corresponding function has an empty body for that specific case.

	RC	CO	PS		ACS	
evaporation	no	no	no	yes	no	yes
LOCAL_EVAPORATE_PHEROMONE()	nop	nop	nop	eq (5)	nop	eq (5)
DEPOSIT_PHEROMONE()	nop	nop	eq (6)	eq (4)	eq (6)	eq (4)
Observations:	no pheromone update	no pheromone update	all ants deposit pheromone		only the best tour is marked	
RANDOM_CHOICE()	random unvisited neighbor	eq (2) $\alpha = 0$, $\beta = 1$	eq (1) $\alpha = 1$, $\beta = 5$		eq (1) $\alpha = 1$, $\beta = 5$	

ch150. Note that the number in the map name indicates the number of nodes of the map, so for example map st70 contains 70 nodes. These maps were chosen to experiment with different values of w_{avg} and $\Delta w = w_{max} - w_{min}$ where Δw is the difference between the maximum and the minimum edge weight and w_{avg} is the average edge weight.

Taking into account that we create an agent for each node, it follows that for st70 problem 70 agents were created. These agents are evenly distributed on 7 computers, so 10 agents must be created on each computer for solving st70 problem.

We define an ant move as the action of transferring an ant from a source node to destination node along a given edge. The number of ant moves was chosen according to the proposal from [4]. There, 1000 moves per ant were chosen for a 19 node map. Taking into account the sizes of our maps and scaling up proportionally, we determine a number $M = 10000$ of ant moves for our experiments.

We ran 10 rounds for each experiment on networks of 7 computers with dual core processors at 2.5 GHz and 1GB of RAM memory. These workstations were interconnected using a high-speed Myrinet interconnection network at 2Gb/s.

In the CO model the ants pick the smallest available weight (i.e the nearest unvisited neighbor) with a greater probability than the rest. This strategy guides the ants to worse solutions than the RC model. As expected, in the RC model, the difference $v_{avg} - v_{min}$ is larger as the tours variate in cost more.

In Table V we considered both evaporation scheme suggested by [4] for ACS (equation (4); see rows marked with $evap = 1$ on Table V) and absence of evaporation (equation (6); see rows marked with $evap = 0$ on Table V). Note that the PS model has no significant benefit from using the ACS evaporation scheme with the recommended parameter values in [4]. As a matter of fact in most cases evaporation decreases the quality of the solutions, see Table V for $evap = 0$ the values of v_{min} and v_{avg} are better than those for $evap = 1$. This is not the case for the ACS version of ACODA where evaporation improves the solutions. This suggests there is still room for improvement in the PS model. However, further study is needed in order to establish wether the evaporation parameters should be adjusted or a completely new approach should be developed.

Our experiments (see Table V) clearly show that the ACODA versions that take into account pheromone deposits are much more efficient at determining good solutions since the values of v_{min} and v_{avg} from Table V are better than those in Table IV. The fact that all these variations can be easily implemented using ACODA supports the flexibility of this architecture, while the fact that the best results are obtained when pheromone deposits are taken into account shows the effectiveness of ACODA in supporting distributed forms of ACO.

V. Related work

TSP is a classic benchmark problem for heuristic search algorithms. With the advent of distributed computing technologies, distributed versions of heuristic algorithms for TSP were also proposed. However, based on our literature review, there are very few works that propose ACO-based truly distributed TSP algorithms. Moreover, there are even fewer proposals that utilize recent advances of multi-agent systems middleware for ACO-based TSP [5]. Nevertheless, we could find references to multi-agent approaches to ACO algorithms for other combinatorial optimization problems ([7], [8], [9]).

A closely related approach to agent-based distributed ACO is presented in [5]. There, both graph nodes and ants are

TABLE IV: Experimental results for RC and CO.

	RC					CO				
map	v_{avg}	$t_{avg}[s]$	v_{min}	$t_{min}[s]$	$T_{avg}[s]$	v_{avg}	$t_{avg}[s]$	v_{min}	$t_{min}[s]$	$T_{avg}[s]$
eil51(426)	1390	85.6	1240	1.72	37.6	1263.8	22	1236	1	38.4
st70(675)	2504.6	53.6	2386	48.9	58.6	2612.6	18.3	2596	6.2	57.8
kroA100(21282)	38729.8	31.4	35276	29.6	99.3	78510.4	49.7	76387	90	145.4
ch150(6528)	46409	58.5	44815	148.1	209.4	31724.8	45.3	30766	186.8	201.2

TABLE V: Experimental results for ACS and PS.

		ACS					PS				
map	evap	v_{avg}	$t_{avg}[s]$	v_{min}	$t_{min}[s]$	$T_{avg}[s]$	v_{avg}	$t_{avg}[s]$	v_{min}	$t_{min}[s]$	$T_{avg}[s]$
eil51	1	448.6	14	440	27.2	40.1	482.2	15.8	471	2	38.1
(426)	0	453.2	19.3	444	7.2	39.7	449.4	12.8	442	10.9	39
st70	1	688.4	37.4	682	47.2	55.2	688	25.3	684	6.4	56.2
(675)	0	729.2	29.2	722	20.2	54.5	684.6	28.1	679	20.8	54.6
kroA100	1	23066	66.7	22380	60	91.9	22601.8	53.8	22286	31.6	92.3
(21282)	0	24887	53	24396	76.7	94.2	22997	29.1	22605	77.3	92
ch150	1	7263	86.6	7150	131.2	188.6	7729.4	17.4	7638	26	194.2
(6528)	0	7930.4	45.3	7806	6	194.7	6753.8	71.7	6692	57.5	192.8

implemented as JADE software agents. Ants centralize information about pheromone deposits and nodes' best tour cost through a single ACL message exchange per node [3]. This procedure adds up to $2n$ messages per tour per ant, where n is the number of nodes. Each ant has to notify the node about its next hop and the cost of its tour in order for the node to be able to update its pheromone levels. This generates other n messages. When an ant completes a tour, it compares the tour cost with the collected best tours from the nodes. A best tour synchronization is triggered for all the nodes if a better tour has been found. This brings an additional overhead of n messages. So, [5] approach requires at most $4n$ messages per tour per ant, while our approach requires at most $2n$ messages: n messages (ant moves) to complete a tour and n messages to deposit the pheromone. We avoid the additional overhead of sending to nodes all the information necessary to carry out their tasks, as in ACODA this information is already contained in ant messages exchanged between nodes.

ACODA implementation reported here is based on the sequential ACS algorithm presented in [4]. There are however three notable differences: i) We avoid the explicit iterations of the standard ACS. An iteration lasts until each ant has found a tour. With the distributed architecture it would be time consuming to provide the synchronization necessary for implementing the explicit iterations, because it would cancel the main benefits of an asynchronous, distributed architecture; ii) In ACS, best tours are compared after each iteration, thus allowing only the best ant to mark its tour. Again, this would require synchronization and centralized computation and we avoid it by allowing all ants to mark their tours; iii) In ACS ants move synchronously, taking one step at a time, while in our approach ants move asynchronously.

Papers [8] and [9] propose JABAT – a JADE-based middleware for agent teams. JABAT supports distributed implementation and collaboration of population-based optimization algorithms. In particular, JABAT was applied to TSP. There is however an important difference between JABAT and ACODA. JABAT agents represent improvement algorithms, which basically means that improvement algorithms are sequential and they cooperate for solving a problem in a distributed way. ACODA agents provide a natural distributed model of the problem environment that is suitable for distributed ACO algorithms. Moreover, we could not find scalability studies referring to JABAT.

Paper [7] proposes a JADE-based multi-agent environment for dynamic manufacturing scheduling that combines intelligent techniques of ACO and multi-agent coordination. However, the focus in [7] is to evaluate the impact of ACO intelligence on multi-agent coordination, rather than to utilize multi-agent middleware for improving ACO. ACO intelligence is embedded into job and machine agents with the role of ants, which is different from ACODA where ants are passive objects exchanged by software agents that provide a distributed model of the problem environment.

In [10] authors compare a distributed form of ACS with flooding algorithm applied on resource discovery problem. They experiment with both algorithms using *ns-2* network simulation tool [11]. The down side of this is that no real execution time measure can be made using their approach. They showed that ACS is the better approach in terms of: best success rate, least number of hops and least traffic. The detailed algorithm and ACO parameters are not presented in order to duplicate their approach using our ACODA framework, for a realistic comparison. The differences between their approach and ACODA are: (i) resource queries are handled centrally at a single anthill, thus introducing single point of failure, (ii) they do not take edge weights into account as they are trying to solve the resource discovery problem, not the TSP problem, and (iii) ants are implemented as *ns-2* mobile agents, while in ACODA we are using JADE. Moreover, in practice there is no need for code mobility as every ant is governed by the same behavioral rules. So, in our approach we use "nodes as agents" to control ants' movement using messages, rather than using code (i.e. JADE agent) mobility.

Paper [12] proposes purely theoretical frameworks for multi-agent systems. No experiments or implementations are mentioned. The authors present a distributed form of ACO based on so called "smart messages" approach to multi-agent systems where agent mobility is used to implement complex communication over dynamic networks. They use delegate multi-agent systems to manage these smart messages in order to design a multi-agent approach for ACO. They do not present ways of representing the environment, determining convergence or stopping condition of ACO experiments.

In [13] authors thoroughly analyze multi-colony ACO algorithms applied on TSP. The main difference between their and our approaches is that they run a separate colony on each available processor. At predetermined points in time, the solutions are centrally collected and local search is performed on the best solution provided by each of the colonies. This is not a fully decentralized approach to ACO, as it requires synchronization and centralized sequential local search after each solution was centrally collected. Our approach is completely decentralized, parallel and asynchronous, and thus it could lend itself to heavy parallelization.

VI. Conclusions

In this paper we presented experimental results with our new multi-agent framework for truly distributed ACO. An initial implementation of the framework using JADE multi-agent platform was outlined. This implementation followed the ACO approach initially proposed for the ACS system. However, three important differences between ACODA and original ACS system were highlighted. This approach was initially evaluated on sample benchmark TSP problems. Experimental results were encouraging, as they clearly support the effectiveness of our proposal for distributed ACO, thus confirming the feasibility of our ACODA framework. We already identified suitable directions for future works: i) strengthen the scalability results by experimenting with this approach on larger TSP problems and larger computer networks; ii) support the generality of our framework by considering other forms of ACO rather than only ACS; iii) improve the ACODA architecture to enable experimentation with larger TSP problems than the current version of ACODA is able to support.

Acknowledgment.

This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund—Investing in People, within the Sectoral Operational Programme Human Resources Development 2007–2013.

References

[1] S. Ilie and C. Bădică, "Distributed multi-agent system for solving traveling salesman problem using ant colony optimization," in *Intelligent Distributed Computing IV, Proc.4th International Symposium of Intelligent Distributed Computing, IDC'2010*, ser. Studies in Computational Intelligence. Springer, 2010, vol. 315, pp. 119–130.

[2] M. Wooldridge, *An Introduction to MultiAgent Systems*. John Wiley & Sons Ltd, 2002.

[3] F. L. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE*. John Wiley & Sons Ltd, 2007.

[4] M. Dorigo and T. Stutzle, *Ant Colony Optimization*. MIT Press, 2004.

[5] E. Ridge, D. Kudenko, and D. Kazakov, "Parallel, asynchronous and decentralised ant colony system," in *In Proc.of the First International Symposium on Nature-Inspired Systems for Parallel, Asynchronous and Decentralised Environments (NISPADE)*, 2006.

[6] G. Reinelt, "Tsplib - a traveling salesman library," *ORSA Journal on Computing*, 1991, http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/.

[7] W. Xiang and H. P. Lee, "Ant colony intelligence in multi-agent dynamic manufacturing scheduling," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 1, pp. 73–85, 2008.

[8] D. Barbucha, I. Czarnowski, P. Jedrzejowicz, E. Ratajczak, and I. Wierzbowska, "Jade-based a-team as a tool for implementing population-based algorithms," in *Proc.6th International Conference on Intelligent Systems Design and Applications: ISDA'2006*. IEEE Computer Society, 2006, pp. 144–149.

[9] I. Czarnowski, P. Jedrzejowicz, and I. Wierzbowska, "A-team middleware on a cluster," in *Proc.3rd KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications: KES-AMSTA'2009*, ser. Lecture Notes in Computer Science. Springer-Verlag, 2009, vol. 5559, pp. 764–772.

[10] S. M. Fattahi and N. M. Charkari, "Ant distributed acs algorithm for resource discovery in grid," *Special Issue of the International Journal of the Computer, the Internet and Management*, vol. 17, no. SP1, 2009.

[11] The ns-2 project, a network simulation tool. Http://nsnam.isi.edu/nsnam/index.php/.

[12] T. Holvoet, D. Weyns, and P. Valckenaers, "Patterns of delegate mas." Los Alamitos, CA, USA: IEEE Computer Society, 2009, pp. 1–9.

[13] C. Twomey, T. Stützle, M. Dorigo, M. Manfrin, and M. Birattari, "An analysis of communication policies for homogeneous multi-colony aco algorithms," *Inf. Sci.*, vol. 180, no. 12, pp. 2390–2404, 2010.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Selected Security Aspects of Agent-based Computing

Piotr Szpryngier, Mariusz Matuszek
Faculty of Electronics Telecommunications and Informatics, Gdansk University of Technology
Email:{piotrs,mrm}@eti.pg.gda.pl

***Abstract*—The paper presents selected security aspects related to confidentiality, privacy, trust and authenticity issues of a distributed, agent-based computing. Particular attention has been paid to authenticity of and trust in migrating mobile agents executables, agent's trust in runtime environment, inter agent communication and security of agent's payload. Selected attack vectors against agent-based computing were described and risk mitigation methods and strategies were proposed and discussed based on presented cryptography measures. In summary expected effectiveness of proposed countermeasures was described.**

I. Introduction

AGENT-BASED computing based on conceptual software entities named *agents* (often *intelligent agents*) plays an important role in integration of various distributed systems and services. Agent paradigm describes agents as closed entities, existing within agent runtime environments (e.g. [1]), capable of percepting and effecting their environment and aware of their *mission*. Agents typically have a large dose of autonomy in pursuing their goals.

Systems incorporating agent paradigm rarely utilise single agents. More often whole *societies* of agents are employed, interacting socially to reach common goals. Typical examples of such interactions are negotiations, contracts, cooperation and coordination of common tasks. All such interactions are based on communication, which in most cases follows established patterns and norms [2]. Runtime environments for agents also try to follow a set of established rules and guidelines [3].

In heterogeneous, open agent environments, where both agents and agent environments originate from and are hosted by different organisations special attention must be paid to security issues—both agent's security and payload (data) security.

Risks may originate from social attacks (e.g. attempts at fraud, cheating, deception) and technical attacks (e.g. agent environment modifying agent's code or data; agents attacking their environment and creating a denial of service by overloading of selected resources or crashing the container etc.). Many authors addressed selected aspects of agent-based computing in recent years. Extensive research of threats and countermeasures was presented by NIST [4]. In a followup to this paper, Jensen [5] described a taxonomy of protection techniques for agents and agent platforms. The taxonomy was divided into two main groups:

- detective measures—methods to detect modifications of agent's code, agent's payload and agent environment;
- preventive measures—obfuscating of code, data, agent's route etc.).

A major role in those techniques is played by cryptography—digital signatures, encrypted code, one time passwords, session keys. A review of agent protection measures reflecting the social element of agent architectures (agent's context, autonomy, communication with other agents, mobility) was presented by Borselius in [6]. Moreover, author proposes the following ways of addressing threats coming from malicious servers:

- contractual agreements—where server platform operators enter into contract with customers, declaring safety of the environment and administrative practices crafted to avoid violation of customers' privacy or integrity of their data, agent code and computations;
- trusted hardware;
- trusted network nodes;
- fault tolerant computing techniques—i.e. cooperation of many agents, partitioning and duplication of data and redundancy in agents functionality;
- execution tracking—gathering and offering logs of agents activities;
- data encryption (however it should be noted that, once decrypted, data will be easily available to interested parties and also that communication is required to obtain decryption key);
- hiding of code and data in blackboxes [7];
- undetachable signatures carrying encoded constraints on permitted agents' code use—if those constraints are not met, messages sent by agents will not be signed by a host;
- sandboxes and code signing when agents' code meets requirements of security policy requirements (proof carrying code).

Computations carried out by use of encrypted functions were deemed prospect less.

Another attempt at protection methods taxonomy was made by Leszczyna [8]. In his paper he proposed to classify protective measures into following subgroups:

- runtime data protection,
- ensuring of mobile code immutability and security,
- event logging.

In addition, he proposed an extension of JADE agent platform protecting agents against tracking. His proposed solution [8] assumes use of trusted parties, encrypting agents route and possibly other data it carries en route between hosts, thus making it impossible for an attacker to determine where a given agent is coming from and its route, which in turn makes it difficult to deduct on agents' goals and findings. A drawback of this approach is a necessary requirement for agents to always return to a recent forwarding trusted party. In addition, nodes forming trusted parties must securely store encryption keys and correctly match them to returning agents.

Yet another JADE extension was proposed in [9], where Zwierko and Kotulski proposed an original method of ensuring mobile agent's integrity, combining protection of code, data and computational state. Their method was based on secret sharing scheme and zero-knowledge protocol.

A secure agent platform, based on Public Key Infrastructure (PKI), was described by Ismail [10]. Author points at the importance of protection of objects containing data from other agents or server data, because such objects may be referenced by a mobile agent. Java environment (sandboxes, memory protection) provides only a partial protection of data against destruction by a malicious mobile agent, because it lacks mechanisms which would allow definition of various access rights based on authentication or permit runtime evolution and delegation of such access rights during execution and communication as well as carrying such rights during agents migration. A proposed solution [10] is to introduce a mechanism of dynamic permissions exchange between cooperating agents. When migrating between hosts an agent needs to carry its permissions and export them to other agents already working on target host. To allow it agents must have means to mutually authenticate themselves. It was proposed that such mechanisms are to be provided by hosts using an external, trusted PKI. A trusted CA generates a key pair for each host (platform) and signs the public keys. Initially agents provide only minimal rights to other cooperating agents. As cooperation progresses those rights may be gradually extended. A *right* in this context means a particular method able to act on a particular object. In her deliberations author did not address agents' mobile nature, methods of rights protection, ease of implementation or expected gains. In her paper she defined requirements and analysed various authentication scenarios:

- of a client agent created by a local server;
- of a foreign (visiting) agent—a problem in this case is the necessary communication between a local and a foreign agent because of its impact on processing performance.

Access control lists (ACLs) were proposed for rights management and secure naming services and servers were proposed to allow communication where symbolic names are used as addresses. Each server uses a key-store mechanism to store its private key. Every agent has three relations (sender, owner, writer) with other entities (servers, users): Each of them signs agents' code, data and rights.

A protection against hostile hosts hosting agent platforms was proposed by Hohl [7]. Author stresses that protection of mobile agents against malicious servers and protection of agent environments against malicious agents are of equal importance. He describes attacks which can be carried by servers against agents' data, code and execution. Two main attack classes are distinguished:

- passive attacks—where agents' data, code and communication are spied upon / intercepted;
- active attacks—where agents' data, code, execution patterns and communications are subject to modifications, alterations and impairment.

A main concept addressed in [7] is a blackbox mechanism—blocking access to code and data. Author states, that it is not possible to construct an agent with a generic blackbox, but only with a time limited one. The stated reason is threat of dictionary attacks. He proposes to construct such a time limited blackbox with the use of transformations using random parameters (so called mess-up algorithms). However, this method ca not be applied to every kind of code and data. In addition, blackboxes may be vulnerable to various attacks, including sabotage, testing of limitations (e.g. finding agents' upper limit on price it is willing to pay) which lets an attacker uncover partially agents' data. Another problem is finding the right time limit of a blackbox. Too short limit may impede achievement of agent's goals and reveal its data and mission before it is accomplished. Setting the limit too long increases the chance of a successful attack against the blackbox.

II. Security issues in typical agent environments

Each type of computing has security considerations. Agent based computing is no different in this respect. Typical problems include: *authenticity* (who is the sender/author of a message), *confidentiality* (of sensitive/valuable data) and *privacy* (when personal data is carried/processed). Sometimes non-repudiation is also important. Cryptographic protocols are used to address these issues, mostly digital signatures and data encryption. Agents may sign messages to try to ensure its authenticity. As discussed later in this paper, this approach may be insufficient. All the mentioned security aspects of computations influence users trust in results—also known as results authenticity. Agent-based computing security mainly consists of:

- ensuring agent authenticity (is it really the agent sending data and not a malicious phantom (fake));
- trusted computing environment (including trusted agent platform — i.e. not introducing changes to agent's code or data; not interfering with agent actions unless it is necessary to protect itself against a malicious agent);
- authenticity of results—obtained results can be trusted to be original, not falsified;
- maintaining confidentiality of results during their transfer to recipient.

Typical agent environments [11][12][13] assume full trust in hosts and provide only a limited set of security tools — user authentication and cryptographic data tunnels (TLS/SSL). Moreover agent containers may be authenticated using digital signatures. Typically combinations of DSA/SHA1 and RSA/MD5 are used. MD5 algorithm is considered weak [14] as collision possibilities were detected. To start data signing it is necessary to provide a password deciphering a private key. It implies, that during agent activity either the password or the deciphered private key is accessible and visible, otherwise user engagement would be necessary on each signing attempt. Public key certificates are self-signed by corresponding private keys and must be distributed in advance to agent containers with the assumption, that each agent container uses identical combination of hash and signing functions. It implies full control over agent and hosting environment by users deploying agent-based applications. In addition, this environment should be isolated from threats coming from the Internet to ensure trust in results.

III. Attacks on agent environment and security assessment

Typical attacks on agent environments and agent-based computing lead to falsifications of computing results or disruption of agent activities. They include:

- undetected injection of hostile agent(s) posing as legitimate ones and deforming the results of agent interactions;
- snooping or substitution of messages;
- posing as a legitimate agent to damage or destroy data stored on machines hosting agent environment.

Unfortunately, neither JADE [11][12] nor other environments provide security to agents or authenticity of results because of several reasons: agents acting within a remote agent container are in an alien environment, thus are susceptible to remote interference (eavesdropping, intrusion, decompiling, interception of results, etc.). Agents in this situation are unable to securely store any sensitive data (e.g. signing keys, passwords, etc.) because at any time such data may be read by a host. Even SSL session keys are not protected because of these issues.

Current solutions in agent-based computing do not address any of the security postulates [10][4][5][8], i.e. neither really confidential communication is provided nor it is authentic. Proposed remedies are selective in their nature (i.e. nothing is gained by SSL encryption when session keys are readily available). Any solution in this area requires cooperation between agent and a trusted host to be effective [8].

IV. Improving security of agent-based computing

To solve the problem of authentic and confidential delivery of agent-based computing results to the originating party several assumptions should be made:

- hosts taking part in agent-based computing should be methodically designed, tested and reviewed, i.e. they fulfil the requirements of ISO Evaluation Assurance Level 4 [15]. Such requirements are in use when designing digital signature environments that are considered valid by authorities (government, legal system). Similarly, system projects should follow generally recognised software engineering safe practices, i.e. automatic architecture management and testing to ensure system resistance to most potential attacks.
- Public Key Infrastructure with PK certificates issued by recognised Certification Authorities (CA) should be designed and implemented.
- Agent environments should be extended to provide communication interfaces to host based services like signing, time stamping, session key generation, etc.

The solution proposed below is similar to the one described in [10]. However, it is based on the assumption of trusted hosts availability. We postulate, that without the support of trusted hosts the security of agent-based computing can not be attained. With these limitations in mind we envision a secure agent-based computing in a following way:

1) User (computation originating party) signs the agent's code and data, assuring the receiving host of agent's credibility.
2) Agent executes inside a monitored container, trusted not to modify agent's code or results.
3) For communication with other agents or originating host agent calls local trusted host provided services like: digital signing or time stamping. After obtaining either a time stamp certificate or a signature agent combines it with a message and sends it to a recipient. To provide in-transit data security either SSL tunnels may be utilised or, in their absence, messages can be encrypted with a session key enciphered by a recipient's public key. Encrypted session key is attached to a cipher text.
4) A receiving party, using trusted PK certificates of computers involved in computations, verifies the signature of a message, therefore gaining trust in its contents and its origin.

V. Summary

We analyse the security of a proposed approach:

1) A signed agent should convince each hosting computer about agent's code authenticity. Public key certificate of the originating party should be commonly available to allow any party to validate the signature and therefore identify agent's origin.
2) Data and results gathered or produced by an agent will be trustworthy if the environment in which they were obtained is sufficiently secure. Access to a public key certificate issued by a commonly recognised CA attests to the hosts credibility. The procedures of obtaining such a PK certificate ensure, that the host and the organisation running it is sufficiently trustworthy.
3) Similar arguments can be used to indicate, that trust can be extended to a hosting computer and its signature. Host security requirements create a similar level of confidence in its private (signing) key. Agents can not store any signing keys—even in an encrypted form — because at signing it would have to be uncovered and in a such case accessible to all observers.

Implementation of our proposals should significantly improve trustworthiness of agent based computing. They are not perfect, and additionally require a very high level of security for all computers and a PK infrastructure. In our approach it is also mandatory to develop new mechanisms of cooperation between an agent and a hosting computer, because agents can not hide any secret within themselves and such secrets are necessary to use cryptographic protocols. Currently we are working on implementation of our proposals as extensions to Jade environment. We expect to have some results sometime around half of next year and after verification of their efficacy and efficiency we plan to publish our findings.

References

[1] "JADE (Java Agent DEvelopemnt Framework) Online Documentation," Telecom Italia Lab. [Online]. Available: http://jade.tilab.com /doc/index.html

[2] "FIPA — IEEE foundation for intelligent physical agents." [Online]. Available: http://www.fipa.org

[3] "MASIF." [Online]. Available: http://www.omg.org/cgi-bin/doc?orbos /97-10-05

[4] W. Jansen and T. Karygiannis, *Mobile Agent Security*. Gaithersburg, MD 20899: NIST, 1999, vol. NIST Special Publication 800-19.

[5] W. Jansen, *Countermeasures for mobile agent security*. Elsevier, 2000.

[6] N. Borselius, "Mobile agent security," *Electronics & Communication Engineering*, vol. 14, no. 5, pp. 211–218, 2002.

[7] F. Hohl, *Time limited blackbox security: Protecting mobile agents from malicious hosts*, ser. LNCS. Berlin: Springer-Verlag, 2006, vol. 1419, pp. 92–113.

[8] R. Leszczyna, "Architecture supporting security of agent systems," Ph.D. dissertation, Gdańsk University of Technology, Faculty of Electronics Telecommunications and Informatics, 2006.

[9] A. Zwierko and Z. Kotulski, "Integrity of mobile agents: A new approach," *Intl. Journal of Network Security*, vol. 4, no. 2, pp. 201–211, 2007.

[10] L. Ismail, "A secure mobile agents platform," *Journal of Communications*, vol. 3, no. 2, 2008.

[11] "Jade Board: Jade Security Guide," Telecom Italia Lab S.p.A., 2004. [Online]. Available: http://jade.tilab.com/doc/index.html

[12] G. Vitaglione, *Mutual-authenticated SSL IMTP connections*. Telecom Italia Lab, 2008.

[13] V. Gunupudi and S. Tate, "SAgent: a security framework for JADE," in *Proceedings of the 5th intl. joint conference on Autonomous agents and multiagent systems*. AAMAS, 2006, pp. 1116–1118.

[14] B. Schneier, *Applied Cryptography*, 2nd ed. New York: John Wiley & Sons, 1996.

[15] "ISO15408 Common Criteria." [Online]. Available: http://standards.iso.org/ittf/PubliclyAvailableStandards/

ISBN 978-83-60810-27-9
ISSN 1896-7094

Agent-Oriented Modelling for Simulation of Complex Environments

Inna Shvartsman,
Kuldar Taveter
Tallinn University of Technology,
Department of Informatics, Raja
15, 12618 Tallinn, Estonia, Emails:
innashvartsman@hot.ee,
kuldar.taveter@ttu.ee

Merle Parmak
Estonian National Defence College
Riia 12, 51013 Tartu, Estonia,
Email: merle.parmak@mil.ee

Merik Meriste
Tallinn University of Technology,
Lab for Proactive Technologies
and University of Tartu, Institute of
Technology, Nooruse 1 50411
Tartu, Estonia, Email:
merik.meriste@ut.ee

***Abstract*—This article addresses the application of agent-oriented modelling to composing scenarios for simulating problem domains consisting of heterogeneous entities that include humans, physical subsystems, and software components whose behaviours depend on the situation at hand. The article presents an overview of agent-oriented modelling and addresses the application of agent-oriented modelling and simulation for context-aware crisis management and military urban operations. We develop an approach for constructing vignettes of situation-aware behaviour to be further simulated by means of software agents and describe the creation of practical context-aware training scenarios. Finally, the article explores a platform currently in use for possible execution of agent-based simulations. Our approach is applicable in practice for testing typical behavioural vignettes in specific scenarios using the platform. Unique benefit of the proposed approach is its usability to observe how real subjects form their decisions to behave in certain situations.**

I. Introduction

THIS article addresses the problem of composing practical computer-based training scenarios for context-aware crisis management and military operations in urban terrain. To achieve context-awareness, we utilize both agent-based and context-aware computing. The purpose of the latter is to let systems react to users based on their (simulated) environments. In other words, context-aware computing leverages context information to improve the interactions among users and their environments. The kinds of problem domains where our agent-oriented method can be applied consist of heterogeneous autonomous entities that include humans, physical subsystems, and software components whose behaviours 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. In this article, we describe the creation of context-aware scenarios for training purposes. In developing the scenarios, we use the top-down approach of *agent-oriented modelling* [1] where a problem domain is first conceptualized in terms of the goals to be achieved by a *socio-technical system*, the roles required for achieving them, and the domain entities embodying the required knowledge.

Conceptually, we consider models as abstractions reducing the complexity of a system for better understanding of its particular aspects and their impact on the system's behaviour. Naturally, application of a model ought to be "based upon the aspect of the agent's behaviour under investigation as well as the level of aggregation of individual agents and their effects" [15].

Running models of agents involved – simulations – show the effects of the behaviours of individual agents as well as provide information on the complex feedback dynamics required for the understanding of emergent behaviour by the system as a whole. As interactions between the agents involved are highly complex, performing simulations is the only way to predict their outcome. Appropriate simulations can help to understand the expected behaviour of an individual agent or an entire system over time.

A problem domain can be simulated by either developing a new simulation environment for it or by utilising an existing simulation environment. The first approach has been used by the third author in performing simulations of business-to-business electronic commerce [2], distributed manufacturing [3], cooperation between different stakeholders at airports [4], and aircraft turnaround processes at airports [5]. For all the problem domains mentioned, simulation environments were developed from scratch by using the JADE agent platform [6]. A new problem domain currently addressed by us is military operations in an urban environment. Differently from the other projects mentioned [2, 3, 4, 5], in that domain we intend to utilise an existing simulation environment, which is already used for training purposes by the Estonian Defence Forces, rather than developing one from scratch. In both cases, a problem domain at hand should be diligently analysed to enable realistic simulations. This article aims to generalise based on the projects mentioned and the ongoing project and offer an approach for constructing situations to be simulated by means of agent-oriented modelling. The rest of this article is structured as follows. Section II describes the related work in modelling and simulation. Section III gives an overview of agent-oriented modelling. Section IV describes the problem

978-83-60810-27-9/09/$25.00 © 2010 IEEE

domain of military operations in an urban environment. Section V designs by means of agent-oriented modelling the scenario to be simulated based on a given case study. Section VI illustrates platform-specific design of the case study scenario on the VBS2 simulation environment. Finally, Section VII draws conclusions.

II. Related Work

Context-aware computing is only gaining momentum. It has been pointed out at [16] that "the concepts of *situation, context, event, goal, intention, action, activity, behavior* need further studies from a number of different points of view, including the views of situation in linguistics, cognitive science, human factors, computer science and artificial intelligence, as well as in both industrial and military applications". The paper [14] emphasizes an extremely challenging nature of context-aware crisis management: "Uncertainty arises in these environments in several ways: (i) information can be incomplete, (ii) information can have varying degrees of confidence, (iii) information can be inconsistent or contradictory, (iv) information can be numeric but yet interpreted in terms of common sense approximations, and (v) information can evolve over time with respect to (i)–(iv)." The aspect of proper models in this context is difficult to underestimate, despite of its challenging nature. This environment, more than any other factor, strongly influences combat identification (e.g., cognitive processes, situational awareness, and visual discrimination), movement, and the capabilities of (the systems of) the contending parties [9].

III. Agent-oriented Modelling

Agent-oriented modelling proposed in [1] is a holistic approach for analysing and designing socio-technical systems consisting of humans and technical components. In the context of this article, a socio-technical system is a simulation system with "human-in-the-loop" simulation capabilities.

Agent-oriented modelling proposes a set of canonical models. The models for analysing a problem domain describe the functional goals of a socio-technical system to be designed, the quality goals describing how the functional goals should be achieved, the roles required for achieving the goals, and the domain entities capturing the knowledge to be represented within the system. Design models of agent-oriented modelling describe what human and man-made (e.g., software) agents are required for achieving the goals, what private and shared information these agents possess and process, and how they interact and behave. Analysis models and design models are complemented by platform-specific design models that describe the implementation of the socio-technical system on a particular software platform. The types of models proposed by agent-oriented modelling are represented in Table I. In addition to representing for each model the abstraction layer (analysis, design, or platform-specific design), Table I maps each model to the vertical viewpoint aspect of interaction, information, or behaviour. Each cell in the table represents a specific viewpoint. We will next give an overview of agent-oriented models proceeding by viewpoints.

From the viewpoint of *behaviour analysis*, a *goal model* can be considered as a container of three components: goals, quality goals, and roles [1]. A *goal* is a representation of a functional requirement of the socio-technical 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 subgoals and subquality 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. Roles are modelled in detail in the viewpoint of interaction analysis. The notation for representing goals and roles is shown in Table II. This notation is used in Section V for presenting agent-oriented models in the example case study. Goal models go hand in hand with *motivational scenarios* that describe in an informal and loose narrative manner how goals are to be achieved by agents enacting the corresponding roles [1].

From the viewpoint of *interaction analysis*, the properties of roles are expressed by role models. A *role model* describes the role in terms of the responsibilities and constraints pertaining to the agent(s) playing the role. *Organisation model* is a model that represents the relationships between the roles of the socio-technical system, forming an organization [1].

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 socio-technical system [1].

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. *Interaction models* represent interaction patterns between agents of the given types. They are based on responsibilities defined for the corresponding roles [1].

In this article, we represent interaction models by means of action events and non-action events [1]. An *action event* is an event that is caused by the action of an agent, like sending a message or starting a machine. An action event can thus be viewed as a coin with two sides: an action for the performing agent and an event for the perceiving agent. A message is a special type of action event—*communicative action event*—that is caused by the sending agent and perceived by the receiving agent. On the other hand, there are *non-action events* that are not caused by actions—for example, the fall of a particular stock value below a certain threshold, the sinking of a ship in a storm, or a timeout in an action. The notation for modelling both kinds of events is represented in Figure 1. Non-action events also include exogenous events. An *exogenous event* is a kind of event whose creating agent we are not interested in. As has been pointed out in [4],

Table I.
The model types of agent-oriented modelling

	Viewpoint aspect		
Abstraction layer	Interaction	Information	Behaviour
Analysis	Role models and organization model	Domain model	Goal models and motivational scenarios
Design	Agent models and interaction models	Knowledge models	Scenarios and behaviour models
Platform-specific design	Platform-specific design models		

Table II.
Notation for modelling goals and roles

Symbol	**Meaning**
▱	Goal
☁	Quality goal
🯅	Role
――――――	Relationship between goals
– – – – – –	Relationship between goals and quality goals

exogeneous events need to be generated by the given simulation system. For example, the appearance of strangers in the scenario to be described in Section V can be modelled as an exogeneous event that is generated by the simulation system.

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 [1]. Knowledge model is particularly important when designing a simulation system from scratch. Since in the case study described in this article we rely on an existing simulation environment instead, we have chosen to represent knowledge only by the domain model.

Finally, from the viewpoint of *behaviour design*, we model how agents make decisions and perform activities. There are two kinds of models under this viewpoint. A *scenario* is a behaviour model that describes how the goals set for the system can be achieved by agents of the system. *Behaviour models* describe the behaviours of individual agents [1].

IV. Urban Operations

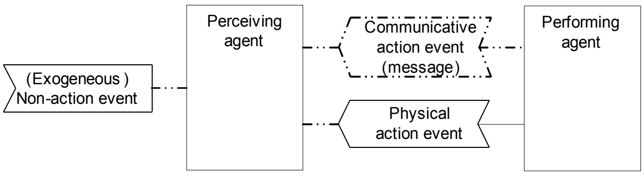

Perceiving agent
Communicative action event (message)
Performing agent
(Exogeneous) Non-action event
Physical action event

Figure 1. The notation for modelling events.

We use agent-oriented modelling in the context of urban operations. Compared to conventional wars between nation-states, multidimensional operations of modern warfare are asymmetric in several aspects. Operational environment in a future battlefield is irregular, characterised by high rate and rapid changes, but also considerable constraints. An example of this kind of operational environment is urban operations to be addressed in the next paragraph. Dimensions, 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 [11]. This multi-dimensionality makes the modelling and simulation of the environment of unconventional warfare complicated but for training purposes highly relevant task.

In the setting described, the nature of recent conflicts, where the population is targeted on the political, social, economical, and physical front, and the current rate of urbanisation, has shifted the attention to urban operations, maybe more than in any other time in history. It is also fair to assume that the battlefield of tomorrow is dominated by the urban environment. An urban area is a terrain where man-made construction and the presence of non-combatants are the dominant features. Urban operations are defined as all operations planned and conducted across the range of military operations on, or against objectives within, an urban area [7]. Urban operations have the following three facets:

- Urban area (terrain, a system of streets and buildings);
- Civilians (all non-combatants in operation area);
- Military operations (*interactions* of combatants of both sides in order to attain an object).

The combination of physical terrain, civilian population, and urban systems fundamentally distinguishes urban operations from other types of operations. To guarantee operational success, it is crucial to consider psychological readiness and especially limits of human capabilities of own forces. Creating working simulations of urban operations for the best possible training presupposes the ability to predict emergent behaviour of the agents in situations. Models must be able to deal with the dynamicity of human behaviour while taking account that it is not always rational. Behaviour is based on agents' sets of beliefs and a multitude of variables that potentially influence those beliefs: *individual*, *social*, and *information* background factors [12]. For military use, this aspect of modelling has been recently highlighted and referred as individual, organisational, and societal (IOS) models [10].

Because of the dynamic nature of urban environment, designing realistic scenarios for simulations of urban operations is not a trivial task. Neither is trivial the evaluation of such scenarios. Appropriate methods are required for coming up with realistic scenarios. One of such methods could be *agent-oriented modelling* that was described in Section III. It is noteworthy that the interaction, information, and behaviour viewpoint aspects of agent-oriented modelling that were explained in Section III rather well correspond to the respective social, information, and individual background factors for agents' behaviours mentioned above that originate in [12].

V. "Agentification" of Simulations

The dynamic nature of crisis management and urban military operations in particular is driving the need for new types of computational models that focus on human behaviour, especially on human behaviour in social units, such as organisations and societies ([10], p. 23). As was pointed out in Section IV, urban environment is characterized by uncertainty and intense *interactions* between various kinds of agents and between agents and their environments. In other words, one needs to introduce the context of interactions leading to situational awareness of the agents involved. A reasonable way to achieve that is by proper "agentification" of simulation scenarios based on agent–oriented modelling. By "agentification" we mean employing agent-related abstractions for modelling and simulation. "Agentification" is thus broader than just using software agents in simulations. Most military simulation environments available today view simulations in terms of objects to be manipulated rather than interacting agents. However, these environments can be used in an agent-oriented fashion but a different mindset is required for doing this. According to this mindset, models must deal with inherent uncertainty and dynamic adaptation that characterise human behaviour and should be capable for modelling both rational and non-rational behaviour ([10], p. 6). This kind of mindset can be achieved by using agent-oriented modelling for analysing the problem domain at hand and composing a simulation scenario for it. We will next illustrate this claim by the case study that is based on a game scenario for the field simulation to be used in training. The scenario has been worked out with the help of adventure games′ specialists [8]. The scenario is one of the scenarios that have been used, assessed, and elaborated in numerous psychological experiments [8]. We will next turn the scenario into the simulation scenario by using the kinds of models suggested by agent-oriented modelling and overviewed in Section III. We prove our point by using just a subset of agent-oriented models.

The first model to be created is the goal model that determines the overall purpose of the simulation and its subgoals. This model serves to discuss the purpose of the simulation with all the stakeholders involved: military commanders and experts, trainers, trainees, adventure games′ specialists, etc. As is reflected by Figure 2, the overall purpose of the simulation is to evacuate the building. Achieving the purpose can be divided into the following subgoals, each of which represents a particular aspect of the evacuation: penetrate into the building, help the injured, ensure safety inside, ensure safety outside, and collect and pass information. Each subgoal can, in turn, be divided into third-level subgoals. Figure 2 represents the refined subgoals for the "Help the injured" subgoal. For clarity, the other subgoals are elaborated in separate figures which we do not present here because of space constraints. Achieving a goal may be characterized by a quality goal which in the given context represents the criteria for evaluating the extent to which the goal in the simulation has been achieved. The goal model also shows the roles that are required for achieving the goals of the simulation scenario. The roles are separately modelled further on in this section.

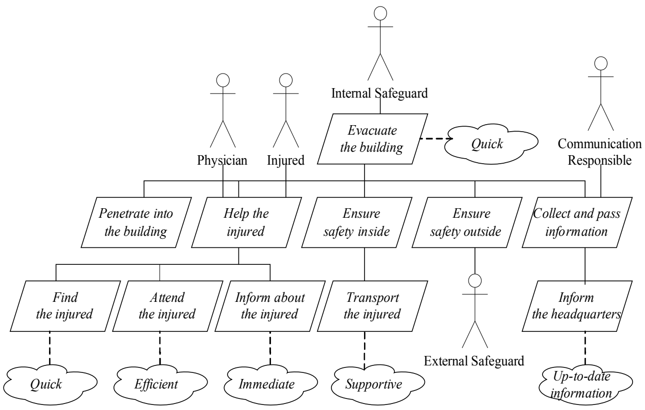


Figure 2. The goal model for the urban operation.

Table III presents a motivational scenario for the case study. The motivational scenario describes in a loose narrative manner how the goals represented in the goal model are to be achieved by agents playing the roles included by the goal model.

The roles put forward by the goal model are modelled in Tables IV - VIII. As usual, the roles are described in terms of the responsibilities and constraints applying to the agents that will perform the roles.

After having defined the goals and roles for the simulation scenario, we will characterize the relations between the roles involved. This can be done by the *organization model*. In our example, the organization model depicts relationships between the roles Injured, Physician, Internal Safeguard, External Safeguard, and Communication Responsible. It is also shown in Figure 3 that an agent performing the role Communication Responsible reports to an agent performing the role Headquarter. The role Headquarter is not a part of the simulation scenario because it is played by an external agent that collects information from an agent performing the role Communication Responsible.

We also need to describe the resources used by agents playing the roles to achieve the goals of the simulation scenario. In the simulation scenario, each *resource* is some unit of information used by agents. The resource types of the scenario are listed in Table IX. The second column of the table shows the roles related to the resources. Each resource is briefly characterized in the "Description" column.

Combining the organization model with resources yields us the domain model represented in Figure 4. As was described in Section III, *domain model* is a kind of conceptual model of a system which describes the entities embodying knowledge in the system and the relationships between them. According to the domain model shown in Figure 4, the resource type Injuries is associated with the

Table III.
The motivational scenario for the simulation

Scenario name	An urban rescue operation
Scenario description	The building that is located in the enemy`s territory and shielded our warriors was hit by a bomb. The rescue team has to perform the following tasks: • Penetrate into the building; • Find the warriors killed; • Find and evacuate the warriors injured; • Find and detonate possible explosives. During evacuation, the following events occur: • Civilians appear outside of the building; • Small cave-in occurs in the building.
Quality description	The building is in ruins, low, and dark. There are bodies and many obstacles in the building. Because of the danger of cave-in, the tasks have to be accomplished as soon as possible and definitely within 30 minutes. All the members of the rescue team are equipped with radio transmitters. The members of the rescue team have to provide other team members and the headquarter constantly with up-to-date information.

Table IV.
The role model for External Safeguard

Role name	External Safeguard
Description	The role of the external safeguard of the building during the operation
Responsibilities	Ensure safety outside the building Inform the Communication Responsible about any potential threats Receive the injured from the Internal Safeguard along with the instructions Inform the Communication Responsible about the injured received and the instructions
Constraints	Quick, efficient, informed, and helpful behaviour

Table V.
The role model for Physician

Role name	Physician
Description	The role of the physician during the operation
Responsibilities	Penetrate into the building Find the bodies in the building Tell the injured apart from the dead Inform the Communication Responsible about the injured and dead found Attend the injured Pass the injured to the Internal Safeguard along with the instructions
Constraints	Quick, efficient, informed, and helpful behaviour

roles Injured, Physician, Internal Safeguard, and External Safeguard. More precisely, information about injuries flows from an agent performing the role Injured to the agents playing the roles Physician and Internal and External Safeguard. The resource type Cave-In is associated with the roles Internal Safeguard and Communication Responsible, i.e., an agent performing the role Internal Safeguard informs an agent playing the role Communication Responsible about the cave-in. Also, an agent performing the role External Safeguard informs an agent playing the role Communication Responsible about the injured who has been evacuated. For this purpose we have introduced the resource type Evacuated which is associated with the roles External Safeguard and Communication Responsible. The last but not the least resource is that of the type Situation. This resource presents the information collected about the evolving situation in the urban operation. An agent performing the role Communication Responsible passes this information to an agent performing the role Headquarter.

Table VI.
The role model for Internal safeguard

Role name	Internal Safeguard
Description	The role of the internal safeguard of the building during the operation
Responsibilities	Penetrate into the building Ensure safety inside the building Find and detonate possible explosives Inform the Communication Responsible about any potential threats Support the Physician in attending the injured Pass the injured to the External Safeguard along with the instructions by the Physician
Constraints	Quick, efficient, informed, and helpful behaviour

Table VII.
The role model for Communication Responsible

Role name	Communication Responsible
Description	The role of the communication responsible in the operation
Responsibilities	Collect and pass information to the headquarter
Constraints	Quick, efficient, informed, and helpful behaviour

Table VIII. The role model for Injured

Role name	Injured
Description	The role of the injured in the operation
Responsibilities	Tell the physician about the injuries
Constraints	Precise information

As the final step of problem domain analysis, we represent the environments to be simulated and how they are related to the roles of the scenario. By an *environment* we mean a set of surrounding conditions for agents that mediates the interactions among agents and their access to resources [1]. We represent the environments by the environment model depicted in Figure 5. The simulation scenario involves two environments: City and Building. The roles Physician, Internal Safeguard, and Injured are enacted in the Building environment, while the roles External Safeguard and Communication Responsible are enacted in the City environment. The Injured role is enacted in both environments. The city consists of several buildings and the building contains several bodies and obstacles. All this may

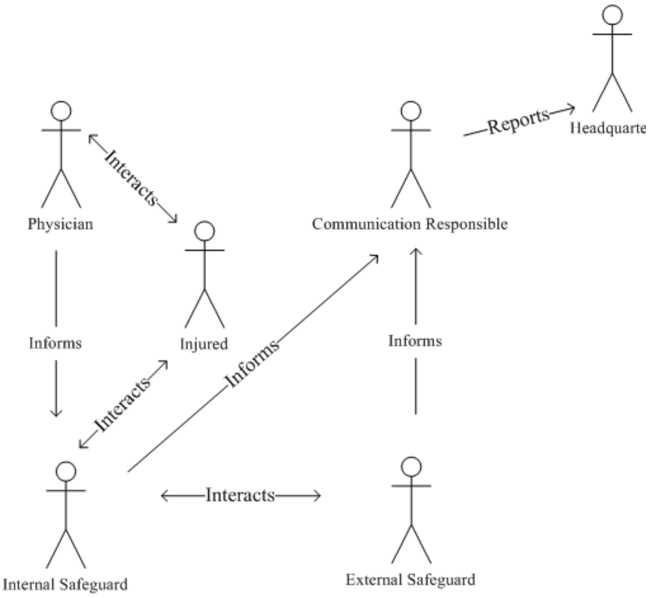


Figure 3. The organization model for the simulation scenario.

TABLE IX.
THE RESOURCES OF THE SIMULATION SCENARIO

Resource	Roles	Description
Injuries	Injured, Physician, Internal Safeguard, External Safeguard	Information about the injuries
Cave-In	Internal Safeguard, Communication Responsible	Information about the cave-in
Evacuated	External Safeguard, Communication Responsible	Information about the injured who has been evacuated
Situation	Communication Responsible, Headquarter	Information collected from agents performing the roles Internal Safeguard and External Safeguard

sound trivial but is definitely required for creating the simulation scenario.

Having defined the goals for the scenario to be simulated and the roles comprised by the scenario, as well as the information resources involved by the scenario, their relationships to roles, and the environments in which the scenario occurs, we have created a minimal set of analysis models. Our next task is to design simulations in such a way that any role in the simulation system could be performed by either a human agent or a software agent. This enables to perform training simulations in teams of any size and evaluate the performance of individual human agents. We illustrate platform-independent design by presenting in Figure 6 an interaction model for the scenario. The interaction model depicted in Figure 6 includes the roles of three purposeful agents – External Safeguard, Internal Safeguard, and Communication Responsible – whose goals comply with the goals set for the simulation scenario by the goal and role models. In addition, the interactions involve the role Physician that is not represented in this figure. Corresponding to the notation represented in Figure 1 and according to the explanations provided in Section III, the interaction model represents the interactions between agents performing the above-mentioned roles as action events. In addition, the interaction model includes two non-action events representing the cave-in and appearance of strangers. Distinguishing between action events and non-action events is crucial in the simulation of military operations. We have decided to model the non-action events as exogeneous events because both of them are generated by the simulation environment. How these events can be defined for a particular simulation environment is exemplified in Section VI. Please note that the notation used in Figure 6 does not prescribe any order for the occurrence of events.

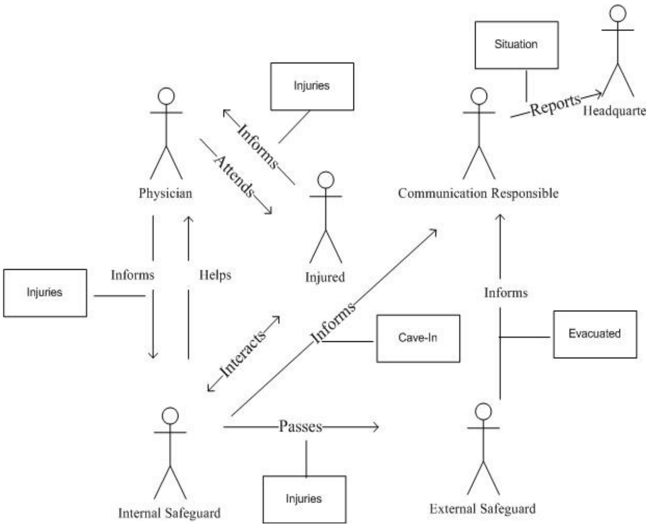


Figure 4. The domain model for the simulation scenario.

What does explicit capturing of interactions buy us in simulations? First of all, the simulation environment can generate exogeneous non-action events at different times within certain intervals. Second, the simulation environment can randomly vary interaction latencies between agents playing the roles. These aspects give us a freedom to

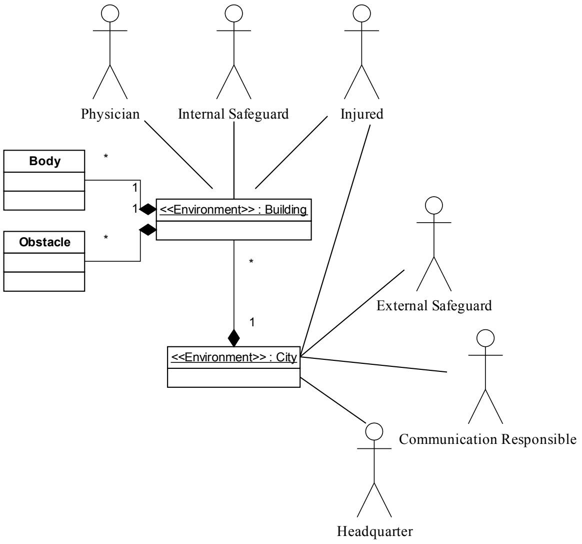


Figure 5. The environment model for the simulation scenario.

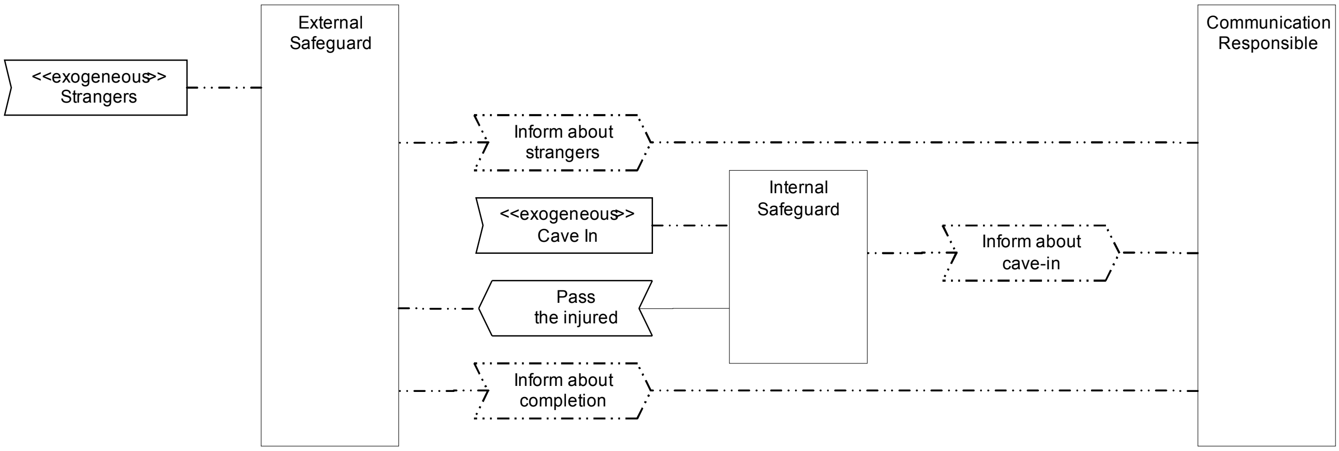


Figure 6. Interaction model for the scenario to be simulated.

manipulate with unpredictable environmental variables which are characteristic to real military operations. Third, we are not dependent of a group size because we can have one or more roles played by humans and the rest of the roles played by software agents. In such a way we can combine "real" interaction latencies with simulated ones. Since simulated latencies follow a random pattern, as a result we will have an *emergent behaviour* of a kind.

Mimicking an operational reality with this kind of design, we can find out how a human would react to different events occurring in the environment or caused by other agents. In addition, we can also experiment with various parameters that define the behaviours of individual agents. As the behavioural aspect is a complex one and therefore deserves further attention and research efforts by our multidisciplinary research team, we will address it as an important part of our future work. In this article we will confine the treatment of individual agent behaviours to a small example to be presented in Section VI.

VI. Platform-specific Design

We have chosen Virtual Battlespace 2 (VBS2, http://www.vbs2.com/) [13] as the platform for agent-oriented simulations. The rationale for this choice is twofold. First, VBS2 is one of the simulation platforms used by the Estonian Defence Forces. Second, VBS2 serves as a good example for simulation platforms that have been designed without agents on mind but that nevertheless can be used in an agent-oriented fashion. As implementing agent-oriented simulations is still work in progress, we illustrate platform-specific design for VBS2 by the snapshot shown in Figure 7. The snapshot reflects the experiments we have performed with VBS2 until now. The experiments have revealed that the agent-oriented analysis and design models presented and explained in Section V provide a sufficient backbone for performing agent-oriented simulations within the VBS2 environment. The details of such simulations are being elaborated and represented as a set of platform-specific models but the main structure of the simulations is in place. The snapshot shown in Figure 7 illustrates how the exogeneous event Strangers modelled in Figure 6 can be represented by means of VBS2.

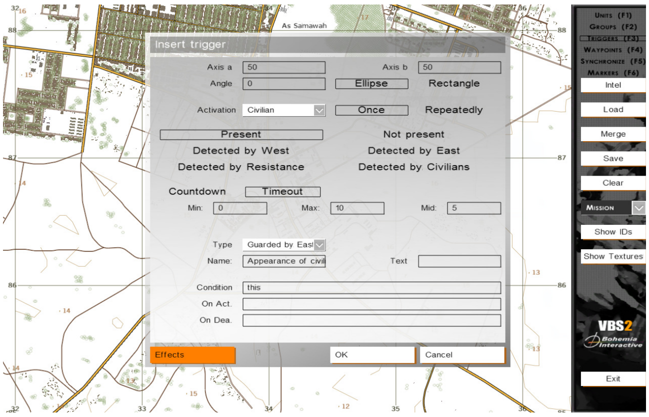
VII. Conclusions

We have addressed composing computer-based training scenarios for context-aware crisis management. In this article, we described the creation of context-aware scenarios for training staff for military operations. The method we have applied for developing the scenarios is agent-oriented modelling [1]. We described how the problem can be modelled in an agent-oriented fashion by creating the goal model, role models, and domain model for the scenario and turning them into the environment model, interaction models, and platform-specific models. The contributions of this article are as follows:

- Employing agent-oriented modelling for structuring a networked domain so that it would lend itself to agent-oriented simulation;
- Offering a conceptual top-down approach for performing training simulations with VBS2 in teams of any size and evaluating the performance of individual human players;
- Enabling the manipulation with unpredictable environmental variables which are characteristic to real military operations.

Realistic training is a major concern of today's militaries. It is very difficult to train a person to deal with something that has not been defined. For asymmetric operations, and hence for urban operations, asymmetry is an inherent threat. Consequently, we need to develop our training scenarios in the direction of modelling human behaviour. It is especially important to create computational models of interactive human behaviours in a particular context. We intend to support this by agent-oriented behaviour models.

In a practical sense, our approach is applicable for testing typical behavioural vignettes in specific scenarios. Each role played by a software agent can be replaced by a real player in our approach. The unique benefit of the proposed approach is its usability to observe how real subjects form their decisions to behave in certain situations. Using agent-oriented modelling, we can explore the behaviours of humans forming parts of complex socio-technical systems. Through these observations we can eventually reach the level of flexibility required for simulating any complex socio-



Insert trigger

Axis a: 50

Axis b: 50

Angle: 0

Activation: Civilian

Min: 0

Max: 10

Mid: 5

Type: Guarded by East

Name: Appearance of civil

Text:

Condition: this

On Act.:

On Dea.:

Figure 7. Defining the exogeneous event of the appearance of strangers.

technical system as a whole. We also plan to work out a visual environment for agent-oriented modelling that could be linked to several simulation platforms, both "conventional" and agent-oriented.

Acknowledgement

This research was supported by European Social Fund's Doctoral Studies and Internationalisation Programme DoRa.

References

[1] Sterling, L. & Taveter, K. (2009). *The Art of Agent-Oriented Modeling*. Cambridge, MA, and London, England: MIT Press.

[2] Taveter, K. (2005). Business process automation with representing and reasoning on trust. In: *Proceedings of the 3rd International IEEE Conference on Industrial Informatics* (INDIN'05). 10 - 12 August 2005, Perth, Western Australia. Washington, DC: IEEE Computer Society.

[3] Taveter, K. & Wagner, G. (2006). Agent-oriented modelling and simulation of distributed manufacturing. In: Rennard, J.-P. (ed.), *Handbook of Research on Nature Inspired Computing for Economy and Management (527-540)*. Hershey, PA: Idea Group.

[4] Sterling, L. & Taveter, K. (2009). Event-based optimization of air-to-air business processes. In N. Stojanovic, A. Abecker, O. Etzion, & A. Paschke (Eds.), *Proceedings of the Intelligent Event Processing – AAAI Spring Symposium 2009*. Menlo Park, CA: AAAI Press

[5] Miller, T., Pedell, S., Sterling, L & Lu, B. (2010). Engaging stakeholders with agent-oriented requirements modelling. *Agent-oriented Software Engineering Workshop at AAMAS (accepted, in press)*.

[6] Bellifemine, F., Caire, G. & Greenwood, D. (2005). *Developing Multiagent Systems with JADE.* Chichester, UK: John Wiley and Sons.

[7] TNO. (2008). *An Introduction to Urban Operations.* TNO report. The Hague, The Netherlands: Organisation for Applied Scientific Research (TNO).

[8] Parmak, M., Mylle, J. J. C. & Euwema, M. C. (2010). Personality and the perception of situational structuredness in a military environment: Seeking and enjoying sensation versus structure as a soldier. *Journal of Applied Social Psychology (under review)*.

[9] Andrews, D. H., Herz, R. P. & Wolf, M. B. (Eds.). (2010). *Human Factors Issues in Combat Identification*. Surrey, UK: Ashgate Publishing.

[10] Zacharias, G. L., MacMillan, J. & Van Hemel, S. B. (Eds.). (2008). *Behavioural Modelling and Simulation: From Individual to Societies.* Washington, DC: The National Academies Press.

[11] Gross, M. (2010). *Moral Dilemmas of Modern War: Torture, Assassination, and Blackmail in an Age of Asymmetric Conflict*. New York, NY: Cambridge University Press.

[12] Fishbein, M. & Ajzen, I. (2010). *Predicting and Changing Behaviour: The Reasoned Action Approach.* New York, Hove: Psychology Press.

[13] Bohemia Interactive Australia. (2010). *Virtual Battlespace 2 (VBS2)*. Release Version 1.5, 24 January.

[14] Lewis, L., Buford, J. & Jakobson, G. (2009). Inferring threats in urban environment with uncertain and approximate data: An agent-based approach. *Appl Intell 30*: 220–232.

[15] NATO MSG-062. (2010). *Guide to Modelling & Simulation (M&S) for NATO Network-Enabled Capability* ("M&S for NNEC"). NATO RTO Technical Report TR-MSG-062, February.

[16] IEEE CogSIMA 2011. (2010). IEEE Conference on Cognitive Methods in Situation Awareness and Decision Support, Call for Papers. Retrieved August 28, 2010, from http://www.cogsima2011.org/cfp.html.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Improving Fault-Tolerance of Distributed Multi-Agent Systems with Mobile Network-Management Agents

Dejan Mitrović
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Serbia
Email: dejan@dmi.uns.ac.rs

Zoran Budimac, Mirjana Ivanović
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Serbia
Email: {zjb, mira}@dmi.uns.ac.rs

Milan Vidaković
Faculty of Technical Sciences, University of Novi Sad, Serbia
Email: minja@uns.ac.rs

***Abstract*—Large-scale agent-based software solutions need to be able to assure constant delivery of services to end-users, regardless of the underlying software or hardware failures. Fault-tolerance of multi-agent systems is, therefore, an important issue. We present an easy and flexible way of introducing fault-tolerance to existing agent frameworks. The approach is based on two new types of mobile agents that manage efficient construction and maintenance of fault-tolerant multi-agent system networks, and implement a robust agent tracking technique.**

I. Introduction

AGENT technology represents one of the most consistent approaches to distributed systems development. *Software agents* can be defined as executable software entities with varying degrees of intelligence, autonomy, and the ability to communicate to each other in order to solve common problems. An important feature of some software agents is *mobility*, which enables them to move from one node in a network to another. Migration path is often chosen autonomously by the agent, and in general case it cannot be predicted.

Agents need an environment in which they can execute their tasks. A *multi-agent system* (MAS), or *agent framework*, represents a programming environment that controls agent's life-cycle and provides it with all the necessary mechanisms for task execution.

Multi-agent systems are applicable to a wide range of problems, such as information retrieval [11], mobile telecommunication networks [7], and power supply management [20], [21]. These solutions often employ physically distributed, interlinked agent frameworks, in combination with migrating agents. When compared to the standard client-server model, agent-based approach yields in improved performance and flexibility, reduced bandwidth requirements and, ultimately, cost.

Large-scale agent-based solutions are, as any software system, exposed to failures. Communication interruptions in systems comprising a large number of frameworks can occur due to traffic overload or hardware failures; agents carrying important gathered data might become lost because of a software glitch. These faults, however, must not affect the functioning of the system as a whole – power supply must not be interrupted because of a failed MAS node in the network. Fault-tolerance mechanisms should, therefore, be an important part of any MAS implementation.

Fault-tolerance techniques used in traditional distributed architectures are often static, require special infrastructural support, and restrict the functionality of agent frameworks [14]. Instead, multi-agent systems should employ dynamic properties of their agents in order to provide efficient, domain-independent fault-tolerant solutions.

EXtensible Java-based Agent Framework (XJAF) [9] is our own FIPA-compliant [5] MAS. It consists of a set of loosely-coupled modules called *managers*. Each module is responsible for handling a distinct part of the overall agent management process. Managers can be added dynamically and are recognized solely by their interfaces, allowing custom implementations. Multiple distributed instances of XJAF can be connected to each other, forming a tree-like network structure, with the primary instance representing the root of the tree.

XJAF has a built-in support for agent mobility. While pursuing their goals, agents can move freely among the connected instances of the environment. The process of agent location tracking is achieved through *forwarding pointers* [18]: when an agent moves from one XJAF instance to another, the originating instance keeps a migration pointer to the target instance. An agent can then easily be found by following a path through the tree of instances.

When it came to fault-tolerance, the original implementation of XJAF had two major drawbacks. First of all, the tree-like structure of interconnected instances was very fragile, as the breakdown of any one node would divide the network into two sets of mutually unreachable systems. There was also no easy way of reintroducing the repaired MAS to the network. Secondly, as the forwarding pointers technique kept no track of alternative routes, a migrating agent would have become lost if any intermediary MAS in its path broke. The implementation of location tracking was also not robust enough, as it, for example, did not include cyclic path management.

Our main motivation behind the presented work was to alleviate these problems and, by doing so, to improve fault-tolerance of our system. We propose a solution based on the introduction of two types of specialized mobile agents. We have replaced the tree-like structure of interconnected XJAF

978-83-60810-27-9/09/$25.00 © 2010 IEEE

instances with a fully-connected graph, and one of the main tasks of the introduced agents is to allow efficient construction and maintenance of the new organization. Additionally, the newly introduced agents have the job of building a fault-tolerant location tracking system, one that will allow for robust agent tracking mechanism. And although our initial goal was to improve the fault-tolerance of XJAF, our solution based on light-weight mobile agents can easily be applied to any MAS, as it is framework-independent, requiring only few changes to the underlying MAS implementation.

The rest of the paper is organized as follows. In Section II we provide an overview of the related work. Section III describes our approach to building and maintaining a network of multi-agent systems in greater detail. Section IV presents our solution to the robust agent-tracking problem. Finally, in Section V we draw a conclusion of the presented work and propose some future enhancements.

II. Related work

Agents are often used as low-level tools for assuring conditions and providing support for higher-level processes. For example, *middle agents* [6], [17] enable efficient flow of information, by acting as yellow page servers, data integrity protectors, and intermediaries between information requesters and information providers. FUSION@ [3] employs deliberate agents that perform intelligent load balancing, manage all communication to and from the MAS, supervise the integrity of other agents, etc. The main advantage of these approaches is flexibility, as existing agents (that is, functionalities) can be modified, and new agents can be added, all without disturbing the remaining parts of the system.

The benefits of using mobile agents in network management have been presented in [1], [12], and [16]. It has been shown that mobile agents enable flexible, dynamic network building, provide for efficient discovery of network elements that violate normal behavior, and offer remote maintenance features. As opposed to centralized administration systems, distributed agent architectures that host and dispatch light-weight mobile agents increase network management efficiency by reducing the traffic, especially in unstable network environments.

Significant part of the research effort regarding MAS fault-tolerance is oriented towards increasing robustness and failure resistance of agents themselves. This problem can usually be handled with *replication* techniques, such as those described in [2], [13], and [19]. The simplified idea behind these techniques is to keep multiple copies of an agent, distributed across a number of frameworks. In case the original agent fails, one of the copies automatically takes over its task execution process. Although agent fault-tolerance is beyond the scope of this paper, we apply the basic concepts behind the replication techniques to achieve robust agent tracking.

Fault-tolerance of a multi-agent system based on middle agents can be improved with a *swarming controller* [4]. The purpose of the controller is to reduce the chance of an agent becoming isolated. For this task, it relies on two components: (1) a *population manager*, which maintains an optimum number and distribution of mobile middle agents, so that there is always at least one alternative path for two agents to communicate; and, (2) an *information propagator*, which handles knowledge distribution, assuring that if any agent does become isolated, it can still access enough information to continue uninterrupted execution.

DimaX [10], [19] is an agent framework with a advanced, built-in fault-tolerance techniques. It uses interdependence graphs for evaluating the significance of every agent in the system, and maintaining an optimum number of replicas accordingly (process known as *adaptive replication*). The system also employs *host monitors*, low-level agents that try to perform early detection of failures and inform all interested parties about the problem in a timely manner. This would, for example, allow an agent and its replicas to leave the troubling environment, or to move important data from it. At the basis of the monitoring is a *heartbeat* technique, which agents in the system use to indicate their valid operational status to each other.

When it comes to fault-tolerance, certain aspects of the DimaX system are superior to ours. However, DimaX was designed and built around the concept of fault-tolerance. Our goal was to instead propose light-weight solutions that could be easily integrated into any existing MAS.

The problem of failed intermediary nodes in a forwarding pointers-based agent tracking system was addressed in [8]. The proposed directory service requires that every agent remembers up to *n* previously visited hosts, while each framework keeps track of up to *n* different positions of the agent. As the agent moves, these data sets are used to propagate information about the new host, providing a way of building alternative paths to the agent.

This solution to the problem of forwarding pointers is similar to ours. However, our solution offers increased flexibility. First of all, we assign handling of all transition-related information to a new type of agent, releasing the frameworks and actual mobile agents from keeping track of the migration process. Secondly, in the solution proposed in [8], *n* has to be chosen at design time, usually by analyzing the environment in which the system executes. We argue that it is difficult to choose a good value for *n*, mostly because component failure rate is not constant over time. Instead, we adapt the number of alternative paths to the agent's behavior – the more an agent moves across nodes, the more mobility is important to it, and we increase the number of alternative paths to it accordingly. However, for unstable environments, we also allow for the increment rate of number of paths to be risen and changed at run-time.

The Adaptive Agent Architecture (AAA) [14], [15] is a multi-agent system in which middle agents called *brokers* forward requests to and responses from registered agents. Since an operational broker is crucial for the functioning of the entire system, AAA comprises multiple brokers forming a team. Behavior of all team members is guided by a *teamwork theory*, which drives them to perform tasks that are beneficiary to the entire team. For example, if a registered agent becomes disconnected because its broker had failed, all remaining team members will have a joint commitment of reconnecting to the agent.

We have based the behavior of agents in our solution on the teamwork theory proposed in AAA – an agent that detects a failure of a MAS will have a commitment of informing other agents of the failure. However, in AAA, all brokers are located within the same system, and a broker can easily instantiate and add another broker. Our agents solve a more complex problem of creating and maintaining a network of physically distributed frameworks. In addition, we have included some optimizations of the teamwork theory. If an error occurs, not all team member will try to solve it. Instead, the order of problem solvers is organized by a simple procedure.

III. Building a reliable network

Distributed, interconnected multi-agent systems comprising mobile agents can provide efficient, flexible solutions for a range of problems, such as information retrieval, power supply management, telecommunication networks, etc.

XJAF has a built-in support for agent migration and a mechanism for distributed problem solving. Previously, multiple instances of the MAS were organized into a tree-like, hierarchical structure, as depicted in Fig 1. During start-up, each instance (except for the top-level, primary) would inspect the address of its parent node and then register itself with it. But, although easy to built, the tree structure was characterized by a single point of failure, that is a breakdown of any one node in the tree would make all of its sub-nodes unreachable by the rest of the system, and vice versa. In addition, the system did not allow for an easy reintroduction of a repaired MAS.

In order to overcome this problem, we have redesigned the inter-MAS connection system so that it organizes XJAF instances into a graph-like structure. The produced graph is fully connected. This approach is optimal as it introduces no significant overhead and it makes the process of finding a path to the target MAS a trivial issue.

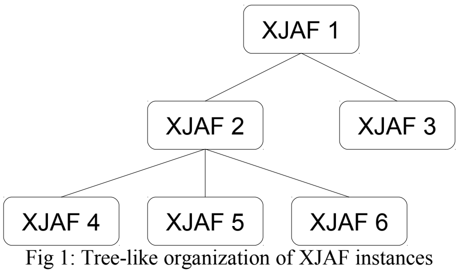


Fig 1: Tree-like organization of XJAF instances

Each MAS contains a list of all other systems in the network. A record in this list is a triplet (*Address, State, Timestamp*). *Address* is the physical address of the neighboring MAS, whereas *State* is its active state, as perceived by the current MAS (e.g. *running, unresponsive*, etc.). *Timestamp*s are an efficient way of selecting between mutually exclusive messages [8]. If an agent receives two messages about a MAS, one informing it that the MAS was started, other that it was shut down, the agent can use the timestamp of each message to determine whether the MAS is currently active (i.e. it was down, now it's up) or whether it is actually unreachable (i.e. it was up, now it's down).

The connection graph is maintained by a special type of mobile agent named *ConnectionAgent*. Each MAS in the network has a single instance of this agent residing in it. *ConnectionAgent* performs the tasks of adding a new MAS to an existing graph, as well as detecting a broken MAS and informing the remaining instances of the failure. As discussed earlier, the benefits of using a mobile agent in network management are lower demands on network traffic and system resources, and improved, distributed control, which becomes especially important in unstable environments.

A. Registering a new MAS

In the previous implementation, a newly created instance of XJAF, *X1*, would have had a single parent MAS to register with. In that configuration, however, a problem might arise if the parent MAS is not available (e.g. due to a failure) during the start-up of *X1*. So instead of having a single parent to register with, *X1* will now scan a predefined range of addresses until it finds the first available MAS. In order to optimize the scan, an administrator can narrow the range as much as possible, but at the same time keeping a sufficient number of possibilities.

After finding an available MAS, *X1* will create an instance of a *ConnectionAgent*, which then takes over the registration process. *ConnectionAgent* is a light-weight mobile agent that follows a simple registration algorithm. It migrates from one node in the network to another, spreading information about the newly created system.

ConnectionAgent keeps a list of all multi-agent systems it has detected so far. Upon arriving to a node in the network, the agent synchronizes its own list with that of its current host. As a result of the synchronization process, the host receives information about the agent's originating MAS, while the agent receives information about all systems the host is connected to. During this process, timestamps are used to select between mutually exclusive messages related to the same MAS, as described earlier. If the synchronization has resulted in any changes (i.e. the agent has learned about some new nodes in the network), *ConnectionAgent* will update its own list and send those changes to the originating MAS.

The agent then chooses next unvisited MAS from its list, and moves there. If there are no more unvisited systems, the process of adding the new MAS to the network has completed successfully, and the agent returns home. There, it continues to execute steps related to detecting a broken MAS, as described in the next sub-section.

Pseudo-code for the *ConnectionAgent*'s registration algorithm is as follows:

```
when (arrived_to_MASx) →
  mark_MASx_as_visited;
  new_MAS_list = synchronize_with_MASx;
  if (not empty(new_MAS_list))
  {
    update_my_list(new_MAS_list);
    ok = send_new_MAS_list_home;
  }
  else
    ok = ping_home;
   if (not ok)
    terminate_and_reverse_reg_process;
  next = get_next_unvisited_MAS;
  if (exists(next))
    move_to(next);
  else
    move_to(home);
```

After the synchronization process, the agent tries to communicate with its originating host. It will try to send it the list of any newly discovered multi-agent systems, or simply "ping" it, if there are no new systems in the result of the synchronization. In any case, the return value of this communication is used to indicate whether the originating MAS is still up and running or whether it had broken in the meantime. If the agent detects a failure of its originating MAS, it will terminate and reverse the registration process, as described in the next sub-section.

Fig 2 (a) depicts a scenario in which a new XJAF instance, *X1*, needs to be added to an existing network comprised of three frameworks: *A*, *B*, and *C*.

When the *ConnectionAgent* is created, only *C* will be in its list. After migrating to *C* and performing the synchronization with it, the *ConnectionAgent* will receive information about the two remaining instances, *A* and *B*. At the same time, *X1* will become registered with *C*. The agent will also send information about newly discovered systems to its originating MAS. This situation is shown in Fig 2 (b). The agent will then repeat the process two more times, visiting *A* and *B* and synchronizing information with them. Finally, it will have no more unvisited instances, meaning that the network is fully built, as shown in Fig 2 (c).

B. Detecting a broken MAS

Upon the successful completion of the registration process, the agent will return to its originating host, where it will continue to execute steps related to maintaining information about the state of the network. However, two problematic situations might occur during the registration: the originating MAS might fail, or the agent might become lost, e.g. due to failure of its current host.

In order to overcome these issues, *ConnectionAgent* needs to keep a heartbeat connection with its originating host. That is, it will communicate with the host at regular intervals, in, among other things, trying to detect its failure. The failure is handled by reversing the registration process – going back through the list of previously visited network nodes and informing them of the broken MAS. Similarly, if the originating MAS doesn't receive any communication request from the agent within a certain time threshold, it will initialize and dispatch a new instance of *ConnectionAgent*, restarting the registration procedure.

In a built network of multi-agent systems, *ConnectionAgents* maintain a heartbeat connection to each other, in order to detect failures. Being guided by the teamwork theory, an agent that discovers a problem will inform other agents about it.

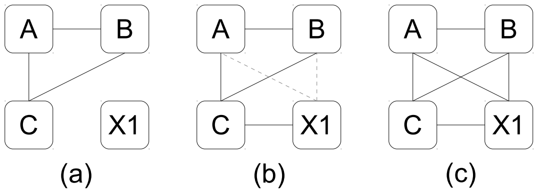


Fig 2: Adding multi-agent system X1 to an existing network

Agents execute asynchronously and independently, so many instances can detect the same failure at the same time and start informing each other, resulting in an unnecessarily increased usage of bandwidth and system resources. To prevent this from happening, we introduce an algorithm for controlling the heartbeat connection.

Multi-agent systems in the network are ordered by their respective timestamps (i.e. start-up times). We will say that *A* is *above B*, if it has earlier timestamp than *B*. Similarly, *A* is *below B* if it has later timestamp than *B*. Each *ConnectionAgent* keeps a heartbeat connection only with agents directly above and directly below it. Therefore, failure of a MAS will trigger a reaction in two agents. An agent that is below the broken MAS will forward this information only to agents below itself. At the same time, it will establish a new heartbeat connection with the *ConnectionAgent* residing in the first available MAS that is above the broken one. Similarly, an agent that is above the broken MAS forwards the information to all agents above itself, and establishes a new heartbeat connection with the *ConnectionAgent* residing in the first available MAS below the broken one. By following this simple set of rules, *ConnectionAgents* can easily detect and handle even multiple failures of a series of nodes.

While informing other agents of a broken MAS, *ConnectionAgent* again relies on timestamps. If, for example, the broken MAS is rebooted before the agent has finished informing other participants of the problem, there will be two agents operating in the network: one telling that the MAS is down, and the other telling that the MAS is up. But, because the second agent is carrying a later timestamp, the correct state can easily be selected.

IV. Fault-tolerant agent tracking

The forwarding pointers approach is an efficient and easy way of tracking agent's whereabouts. As shown in Fig 3, if an agent moves from XJAF1 to XJAF2, and later to XJAF6, there will be a pointer from the first to the second, as well as from the second to the third host in the path. The pointers are used to forward any messages directed to the migrating agent.

However, if XJAF2 (or, in general, any node in the agent's path) breaks, the agent will be lost. As there are no backup paths, the originating framework is unable to determine the current host of the agent. The same holds if only the communication link between any two nodes in the path fails.

Combination of the newly integrated graph-like organization of XJAF instances and the concepts behind replication techniques described earlier can be efficiently used to develop a robust agent tracking system. The algorithm is, again, implemented in a form of an agent, named *RemnantAgent*, making the solution suitable for any MAS.

Each mobile agent in the system has one or more instances of a light-weight *RemnantAgent* assigned to it, with a single instance residing in each MAS node in the path. A *RemnantAgent* can be considered to be a light replica of the agent. It responds to two events triggered by the agent's migration. The first event is signaled when the agent moves from one node to another. The event is dispatched to all but the agent's target MAS. The handler of this event can be described by the following pseudo-code:

```
when (agent_moved_from_X_to_Y) →
  agent_here = false;
  locations_after_this.push_back(Y);
  all_nodes_in_path.remember(X);
  all_nodes_in_path.remember(Y);
  // propagate migration backwards
  foreach (N in all_nodes_in_path)
    if (not N in locations_after_this)
      inform(N, agent_moved, X, Y);
```

RemnantAgent holds the two sets of information: (1) an unordered list of all nodes in the agent's path, and (2) a list of locations the agent has visited after leaving the *RemnantAgent*'s node. The second list is used for iterative reconstruction of alternative paths, while the first list is used to propagate elements of the second list. That is, whenever an agent moves from one node to another, this information is propagated backwards to all nodes in the path. Back nodes are identified by belonging to the list of all nodes in the path, but not to the list of future locations. All *RemnantAgents* that handle this event will propagate the information to all back nodes, which might seem as unnecessary and traffic-heavy. However, this approach is required as any node, or a link between any two nodes, might break at any time.

The second event triggered by the migrating agent is sent only to the agent's target MAS, signifying the agent's arrival. The handler for this event is as follows:

```
when (agent_arrived_from_X) →
  agent_here = true;
  locations_after_this.clear();
  all_nodes_in_path.remember(X);
```

Consider a scenario in which there are 4 multi-agent systems, *A*, *B*, *C*, and *D*, comprising a network, and there is an agent (marked as ☺) originating in *A*. In the pursuit of its goal, the agent moves from *A* to *B*, as shown in Fig 4 (a). The migration triggers a creation of a *RemnantAgent* in *A*, which remembers that agent went to *B* (indicated by → *B*). Each host builds the unordered list of all frameworks interactively, by being part of the migration, or by propagating migration between other hosts. So after the first step, both *A* and *B* have the same list (*A*, *B*): *A* has built its list by sending the agent to *B*, while *B* has built its list by receiving the agent from *A*.

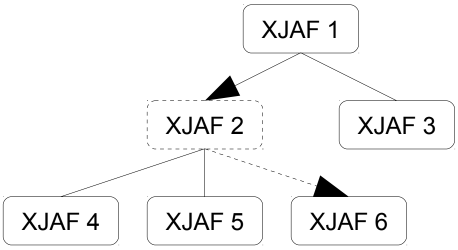


Fig 3: If XJAF2 fails, any agents that have through it are lost

Next, the agent moves from *B* to *C*, as shown in Fig 4 (b). *RemnantAgent* in *B* remembers the new position (→ *C*) and, after comparing this list with that of all nodes in the path (*A*, *B, C*), propagates the migration back to *A*, informing it of the change. *RemnantAgent* in *A* updates the position of the agent, keeping the previous value (→ *B*, *C*). If the agent continues to move, this time from *C* to *D*, the migration process will create another instance of *RemnantAgent* in *C* and trigger the location update in all previous instances. This situation is depicted in Fig 4 (c).

Now, suppose *A* needs to deliver a message to the migrating agent. *RemnantAgent* in *A* has the following list of locations: *B*, *C*, *D*. The last known location is *D* so it forwards the message directly there. But, let's say that there is a problem with the communication link between *A* and *D*, because of which the message cannot be delivered directly. To solve this issue, *RemnantAgent* in *A* inspects the previous host of the migrating agent and tries to deliver the message through it. So it forwards the message to *RemnantAgent* in *C*, which then, by looking at its own list of locations, delivers it to *D*. Similarly, if even *C* is down, the message can still be delivered, by going through *B*. In total, there are 4 alternative paths available:

1. *A* → *D*
2. *A* → *C* → *D*
3. *A* → *B* → *D*
4. *A* → *B* → *C* → *D*.

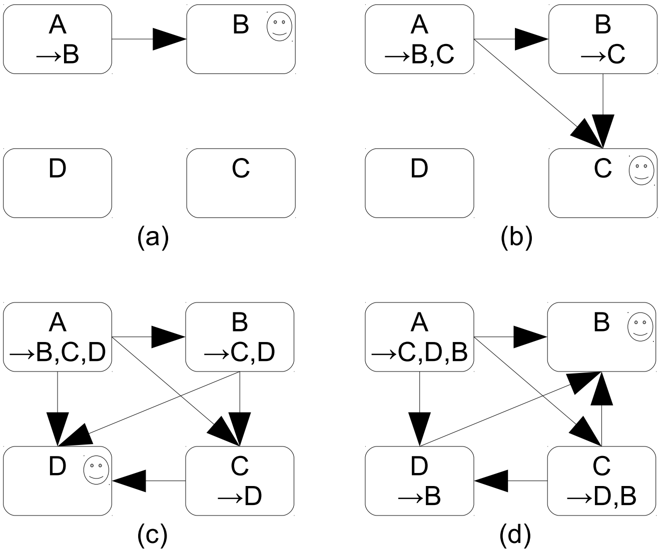


Fig 4: Robust agent tracking system

By going from the last known location backwards, each *RemnantAgent* will try to deliver the message in the shortest path possible.

Cyclic paths can also be easily handled. If, for example, the agent chooses to move from *D* back to *B*, the path tracking algorithm is similar, only this time *B*, as the current host, clears its list of locations, while all other nodes set *B* as the last known location of the agent. In addition, *A* will remove the previous instance of *B* from its list of locations. The final situation is depicted in Fig 4 (d). Again, there are 4 possible ways of reaching the agent from *A*:

1. $A \rightarrow B$
2. $A \rightarrow D \rightarrow B$
3. $A \rightarrow C \rightarrow B$
4. $A \rightarrow C \rightarrow D \rightarrow B$.

To improve the fault-tolerance of the tracking system even further, migration from each host can trigger the creation of *n* additional instances of *RemnantAgent*. These instances would then be distributed across different MAS nodes. Each new node can serve as an additional intermediary step in a path to the agent. The parameter *n* is configurable, and can be changed at run-time to reflect the overall system stability.

V. Conclusion and future work

Fault-tolerance of agent frameworks is an important issue. Large-scale agent-based systems need to be able to function in an uninterrupted fashion, assuring constant delivery of services to end-users. The underlying agent technology must, therefore, be capable of overcoming problems that emerge due to software or hardware failures.

The goal of the work presented in this paper was to offer a light-weight, yet efficient solution for improving fault-tolerance of existing agent systems. This has been achieved by the introduction of two types of mobile agents: (1) *ConnectionAgent* for building and maintaining reliable networks of distributed MAS instances, and (2) *RemnantAgent*, in charge of providing efficient agent tracking mechanisms by introducing robustness to the original forwarding pointers technique. All presented functionalities can be easily integrated in any existing MAS, with minimal changes to the underlying implementation.

Our future work on the subject of MAS fault-tolerance will be concentrated on providing flexible and robust agent replication and timely fault-detection techniques.

References

[1] A. Bieszczad, B. Pagurek, T. White, “Mobile agents for network management”, In *IEEE Communications Surveys*, 1998.

[2] A. Fedoruk, R. Deter, “Improving fault-tolerance by replicating agents”, In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2*, pp. 737 – 744, 2002.

[3] D. I. Tapia, J. Bajo, J. M. Corchado, “Distributing functionalities in a SOA-based multi-agent architecture”, In *7th international conference on Practical applications of agents and multi-agent systems*, PAAMS 2009, pp. 20 - 29, 2009.

[4] D. Šišlak, M. Pechouček, M. Rehak, J. Tožička, P. Benda, “Solving inaccessibility in multi-agent systems by mobile middle-agents”, In *Multiagent and Grid Systems - An International Journal*, pp. 73 - 87, 2005.

[5] FIPA Homepage, http://www.fipa.org

[6] K. Decker, K. Sycara, M. Williamson, “Middle-agents for the Internet”, In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, 1997.

[7] K. Jurasovic, M. Kusek, G. Jezic, “Multi-agent service deployment in telecommunication networks”, In *Agent and multi-agent systems: technologies and applications*, LCNS Springer Berlin / Heidelberg, pp. 560 – 569, 2009.

[8] L. Moreau, “A fault-tolerant directory service for mobile agents based on forwarding pointers”, In *Proceedings of the 2002 ACM symposium on Applied Computing*, pp. 93 – 100, 2002.

[9] M. Vidaković, B. Milosavljević, Z. Konjović, G. Sladić, “EXtensible Java EE-based agent framework and its application on distributed library catalogues”, In *Computer science and information systems, ComSIS*, pp. 1 – 16, Vol. 6, No. 2, 2009.

[10] N. Faci, Z. Guessoum, O. Marin, “DimaX: a fault-tolerant multi-agent platform”, In *Proceedings of the 2006 international workshop on Software engineering for large-scale multi-agent systems*, pp. 13 – 20, 2006.

[11] R. Punithavathi, K. Duraiswamy, “A fault tolerant mobile agent information retrieval system”, In *Journal of computer science*, Vol. 6, pp. 553 - 556, 2010.

[12] R. Stephan, P. Ray, N. Paramesh, “Network management platform based on mobile agents”, In *International journal of network management*, Vol. 14, No. 1, pp. 59 - 73, 2004.

[13] S. Geraci, L. Giacalone, C. Leone, S. Mangano, G. Pitarresi, A. Scaglione, S. Sorce, A. Genco, “Fault tolerance”, In *Mobile agents: principles of operation and applications*, edited by A. Genco, pp. 139 – 179, WIT Press, USA, 2008.

[14] S. Kumar, P. R. Cohen, “Towards a fault-tolerant multi-agent system architecture”, In *Proceedings of the fourth international conference on Autonomous agents*, pp. 459 – 466, 2000.

[15] S. Kumar, P. R. Cohen, H. J. Levesque, “The adaptive agent architecture: achieving fault-tolerance using persistent broker teams”, *Technical report: CSE-99-016-CHCC*, 1999.

[16] T. C. Du, E. Y. Li, A.-P. Chang, “Mobile agents in distributed network management”, In *Communications of the ACM*, Vol. 46, No. 7, pp. 127 - 132, 2003.

[17] T. R. Payne, M. Paolucci, R. Singh, K. Sycara, “Facilitating message exchange through middle agents”, In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2*, pp. 561 – 562, 2002.

[18] T. Y. Yeg, T. I. Wang, “A ratio-based update scheme for mobile agent location management”, In *Agent and multi-agent systems: technologies and applications*, Springer-Verlag Berlin / Heidelberg, pp. 100 – 109, 2009.

[19] Z. Guessoum, N. Faci, J. P. Briot, “Adaptive replication of large-scale multi-agent systems: towards a fault-tolerant multi-agent platform”, In *Proceedings of the fourth international workshop on Software engineering for large-scale multi-agent systems*, pp. 1 – 6, 2005.

[20] Z. Yang, C. Ma, J. Q. Feng, Q. H. Wu, S. Mann, J. Fitch, “A multi-agent framework for power system automation”, In *International journal of innovations in energy systems and power*, Vol. 1, No. 1, 2006.

[21] Z. Zhang, J. D. McCalley, V. Vishwanathan, V. Honavar, “Multiagent system solutions for distributed computing, communications, and data integration needs in the power industry”, In *Proceedings of the General Meeting of the IEEE Power Engineering Society*, IEEE Press, pp. 45 - 49, 2004.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Argumentative agents

(Invited Paper)

Francesca Toni
Department of Computing
Imperial College London, UK
Email: ft@imperial.ac.uk

Abstract—**Argumentation, initially studied in philosophy and law, has been researched extensively in computing in the last decade, especially for inference, decision making and decision support, dialogue, and negotiation.**

This paper focuses on the use of argumentation to support intelligent agents in multi-agent systems, in general and in the ARGUGRID project[1] and Agreement Technology action[2]. In particular, the paper reviews how argumentation can help agents take decisions, either in isolation (by evaluating pros and cons of conflicting decisions) or in an open and dynamic environment (by assessing the validity of information they become aware of). It also illustrates how argumentation can support negotiation and conflict resolution amongst agents (by allowing them to exchange information and fill in gaps in their incomplete beliefs). Finally, the paper discusses how arguments can improve the assessment of the trustworthiness of agents in contract-regulated interactions (by supporting predictions on these agents' future behaviours).

I. Introduction

ARGUMENTATION, initially studied in philosophy and law, has been researched extensively in computing in the last decade, especially for inference, decision making and decision support, dialogue, and negotiation [1], [2], [3].

Simply stated, argumentation focuses on interactions where parties plead for and against some conclusion. In its most abstract form [4], an argumentation framework consists simply of a set of abstract *arguments* and a binary relation representing the *attacks* between the arguments. By instantiating the notion of arguments and the attack relation, different argument systems can be constructed, predominantly based upon logic. One such system is assumption-based argumentation (ABA) [5], [6]. Here, arguments are computed from a given set of *rules* and are supported by rules and *assumptions*. Also, an argument attacks another argument if the former supports a claim conflicting with some assumptions in the latter, where conflicts are given in terms of an underlying notion of *contrary* of assumptions. Rules, assumptions and contraries are defined in terms of an underlying logic *language*. Different choices for this language give different instances of ABA.

Argumentation provides a powerful mechanism for dealing with incomplete, possibly inconsistent information. It is also fundamental for the resolution of conflicts and differences of opinion amongst different parties. Further, it is useful for "explaining" outcomes generated automatically. As a consequence, argumentation is a useful mechanism to support several aspects of agents in multi-agent systems. Indeed, agents are goal-driven, self-contained entities with partial information on the environments in which they are situated (including other agents inhabiting these environments), with conflicting goals, but often in need to cooperate in order to achieve these goals (e.g. because resources controlled by other agents are needed to achieve these goals). Cooperation is supported by communication in multi-agent systems, and opinions as well as explanations are often exchanged amongst communicating agents.

The potential of *argumentative agents* has been/is being explored in two European activities: the EC-funded ARGUGRID project and the COST-funded Agreement Technologies action.

The *ARGUGRID project* developed a grid-based platform populated by rational decision-making agents associated with service requestors/providers and users [7]. Within agents, argumentation as envisaged in ABA is used to support decision making, taking into account (and despite) the often conflicting information that these agents have, as well as the preferences of users, service requestors and providers [8], [9], [10]. In the ARGUGRID platform, argumentation is also used to support the negotiation between agents [10], [11] on behalf of service requestors/providers/users. This negotiation takes place within dynamically formed virtual organisations [12]. The agreed combination of services, amongst the argumentative agents, can be seen as a complex service within a service-oriented architecture [13]. The ARGUGRID approach has been validated by way of industrial application scenarios in e-procurement and earth observation [7], [8], [9].

The *Agreement Technologies action* aims at developing computer systems in which autonomous agents negotiate with one another, typically on behalf of humans, in order to come to mutually acceptable agreements [14], [15]. Agreement Technologies include argumentation, negotiation, and trust computing, as well as combinations of these.

In this paper we review some of the achievement to date of these activities, in providing argumentative agents and validating them against other approaches and in applications.

The paper is structured as follows. In section II we give some background on abstract argumentation and ABA. In section III we illustrate ways in which argumentative agents can take decisions. In section IV and V we describe the use of ABA for conflict resolution and negotiation, respectively, amongst argumentative agents. In section VI we review a

[1] www.argugrid.eu

[2] www.agreement-technologies.eu

possible integration of arguments with statistical information in trust computing. In section VII we conclude.

II. Argumentation

This section gives essential background on logic-based argumentation, focusing on abstract argumentation [4] and assumption-based argumentation [5], [6].

Abstract argumentation frameworks [4] are pairs (Arg, att) where Arg is a set of arguments and $att \subseteq Arg \times Arg$ is a binary relation representing attacks between arguments.

The main purpose of argumentation theory is to identify which arguments in an argumentation framework are rationally "acceptable". Several notions of acceptability have been proposed in the literature on argumentation, some providing an "intrinsic" measure of argument strength (e.g. [16]), whereby the acceptability of an argument depends on its internal logical structure, others giving a "dialectical" measure (e.g. [4], [17], [18], [19]), depending exclusively on attacking arguments.

An example of dialectical measure for abstract argumentation frameworks is given by *conflict-free extensions*, namely sets X of arguments such that there is no argument in X which attacks another argument in X. Another example is given by *admissible extensions* [4], namely sets X of arguments that are conflict-free and capable of defending themselves against every attacking argument (namely for every argument Y that attacks X, there is some argument in X that attacks Y). A further example is *preferred extensions* [4], namely (subset) maximal admissible extensions. These examples of dialectical measures are all "qualitative", based predominantly on the capability of arguments to defend themselves. "Quantitative" dialectical measures have been proposed too (e.g. [19]).

Assumption-based argumentation (ABA) frameworks [5], [6] can be defined for any logic specified by means of (inference) *rules*, by identifying sentences in the underlying logic *language* that can be treated as *assumptions*. Intuitively, *arguments* are deductions (in the chosen logic language) of a conclusion (or claim) supported by a set of assumptions. Then, *attacks* against arguments are always directed at the assumptions supporting the arguments. More precisely, an attack by one argument against another is a deduction by the first argument of the *contrary* of an assumption supporting the second argument.

The inference rules may be domain-specific or domain-independent, and may represent, for example, causal information, or laws and regulations. Assumptions are sentences in the language that are open to challenge, for example uncertain beliefs ("it will rain"), unsupported beliefs ("I believe that some service provider is reliable"), or decisions ("I will purchase a specific service"). Typically, assumptions can occur as premises of inference rules, but not as conclusions. In general, the contrary of an assumption is a sentence representing a challenge against the assumption. For example, the contrary of the assumption "it will rain" might be "the sky is clear". The contrary of the assumption 'I will purchase a specific service" might be "I will purchase a different service" (where I only need one service). The contrary of the assumption "I believe that some service provider is reliable" might be "there is evidence against that service provider being reliable".

Given arguments and attacks, several qualitative dialectical measures of acceptability have been deployed within ABA [5], [17], [6], including conflict-free, admissible and preferred extensions. Query answering with respect to these dialectical measures is implemented, for any ABA framework given as input, in the CaSAPI system[3] [20], [21], [22]. Here, queries represent claims whose dialectical validity with respect to a chosen notion of extension is under scrutiny.

III. Argumentation for decision-making

Qualitative decision theory [23] has been advocated for quite some time as a viable and useful alternative to classical quantitative decision theory [24], when a decision problem cannot be easily formulated in standard decision-theoretic terms using decision tables, utility functions and probability distributions. A number of qualitative approaches to decision making, e.g. [25], [26], [27], [28], have been put forward, with argumentation-based decision making amongst them (e.g. [27]). In decision-theoretic terms, argumentation can be used as a model to compute a utility function which is too complex to be given a simple analytical expression in closed form. Argumentation has been proposed for decision making under certainty (where the outcomes of decisions are known to the decision maker) [8], [9], strict uncertainty (where the outcomes of decisions are uncertain and no probabilistic information is available) [29], [30], [31], [32], [10], and also for decision under risk (where some probabilistic information is known) [16], [33], [34]. Argumentation has also been used to support practical reasoning [35], [36], and decision support systems [37], [38], [39]. Further, arguments can be seen as supporting "values", as in value-based argumentation for decision-making [40]. Finally, argumentation can be used for computing decision tables, utility functions and probability distributions in classical quantitative decision theory [41].

Here we summarise two different uses of ABA (see section II) to support decision making under certainty [8], [9] and under strict uncertainty [10].

We consider the following decision problems:

- Let $\mathcal{D}$ be a (non-empty) set of alternative decisions.
- The outcomes of decisions are individual states $s \in \mathcal{S}$ (if under certainty) or sets of states $S \subseteq \mathcal{S}$ (if under uncertainty).
- States can be seen as sets of "goals" of (or benefits for) the decision maker, which can be represented as sentences in a given set $\mathcal{G}$.
- Preferences over goals may be optionally specified, e.g. in the form of weights (positive integers). These can be expressed by a mapping $w : \mathcal{G} \rightarrow \mathbb{N}$. The case where all weights are the same is equivalent to the case where no weights are specified.

[3] http://www.doc.ic.ac.uk/~dg00/casapi.html

- Rather than being known a-priori, outcomes of decisions are determined from a belief base $\mathcal{B}$. The beliefs "entailed" by this base correspond to states.

Decisions may correspond to products, including e-procurement products [8], earth observation products [9], commodities [10] etc. For these decision problems, argumentation can be used to determine the relative value of different decisions, to single out decisions with "top-most" value and for explaining decisions.

Under certainty, and assuming all goals have equal weight, decisions with top-most value can be defined as "dominant" decisions, where

- a decision $d \in \mathcal{D}$ is *dominant* if and only if the outcome $s \in \mathcal{S}$ of d is such that, for any alternative decision $d' \in \mathcal{D}$, if $s' \in \mathcal{S}$ is the outcome of d', then $s \supseteq s'$.

This is the approach taken in [9].

Alternatively, decisions with top-most value are decisions resulting in states that are upper bounds of partial orders over states. Under certainty, a partial order $\sqsupseteq$ over states can be given as follows:

- a state $s \in \mathcal{S}$ is *strictly preferred to* a state $s' \in \mathcal{S}$ (denoted $s \sqsupset s'$) if and only if
  1) there exists a goal $g \in \mathcal{G}$ such that $g \in s$ but $g \notin s'$, and
  2) for each goal $g' \in \mathcal{G}$, if $w(g') \geq w(g)$ and $g' \in s'$ then $g' \in s$;
- a state $s \in \mathcal{S}$ is *preferred to* a state $s' \in \mathcal{S}$ (denoted $s \sqsupseteq s'$) if and only if $s \sqsupset s'$ or $s = s'$.

This partial order is used in [10] to define a partial order over decisions under strict uncertainty, as we will see below.

Note that, when all weights are the same, $s \sqsupseteq s'$ is equivalent to $s \supseteq s'$.

In [8], [9], the belief base maps features of products to goals of the decision maker. For example, a hotel with rooms costing less than 50£ may be believed to be cheap (where the price is a feature and being cheap is a goal) [9]. Further, in the case of e-procurement for an e-ordering system [8], an e-ordering system with a 3-year flat cost may be deemed to decrease costs. Here, features determine univocally (with certainty) goals. The belief base is represented as an ABA framework from which the following arguments can be built:

(i) "choose decision d because d allows to achieve goal g"
(ii) "do not choose d because some other decision allows to achieve goal g and I am not sure d does"

(see [8], [9] for formal details). Arguments of type (ii) attack arguments of type (i) and vice versa. Then, dominant decisions, as given above, correspond to admissible sets of arguments for the given ABA frameworks. ABA thus allows to compute dominant decisions and explain these decisions (by presenting the arguments). Moreover, in [9] we also propose a different argumentation semantics based on counting (and resulting in a numerical, rather than Boolean value for arguments) and links this semantics to dominant decisions when weights are given (see [9] for details).

In [10], the belief base encodes again a mapping between features and goals, but using information that may be incomplete (e.g. that a good school is located in the vicinity of a real estate property) and that may lead to conflicts/inconsistencies (e.g. that a real estate property is in a safe area and is not in a safe area).[4] As a consequence, decisions correspond to sets of states, where different states correspond to different assumptions (completing the information) and different resolutions of the conflicts. These resolutions (states) are preferred extensions, in the argumentation sense (see section II), and they can be compared using the standard minimax criterion from decision theory, using the following notations:

- for a given decision $d \in \mathcal{D}$, let $cred\_pref(d)$ be the set of all $s \in \mathcal{S}$ such that s is satisfied in some preferred extension of the belief base extended by d;
- for any set of states $S \subseteq \mathcal{S}$, let $min(S)$ be a state such that for each goal $g \in \mathcal{G}$, g is satisfied in $min(S)$ if and only if g is satisfied in every state in S.

Then

- a decision $d \in \mathcal{D}$ is *minimax-preferred to* a decision $d' \in \mathcal{D}$ if and only if
  - $min(cred\_pref(d)) \sqsupseteq min(cred\_pref(d'))$

  where $\sqsupseteq$ is as defined earlier.

This notion of minimax-preference is equivalent to a purely argumentative preference notion between sets of states, defined using the following notion:

- for a given decision $d \in \mathcal{D}$, let $scept\_pref\_state(d)$ be the state $s \in \mathcal{S}$ consisting of all goals holding in all preferred extensions of the belief base extended by d.

Then,

- a decision $d \in \mathcal{D}$ is *sceptically-preferred to* a decision $d' \in \mathcal{D}$ if and only if
  - $scept\_pref(d) \sqsupseteq scept\_pref(d')$.

The fully argumentative notion of sceptically-preferred and the partially argumentative, partially decision-theoretic notion of minimax-preferred are equivalent [10]. Top-elements of the partial orders given by either notions are decisions with top-most values.

Overall, the two approaches considered use argumentation for different purposes: on one side, [8], [9] encodes a fully decision-theoretic notion of dominance into an ABA framework, and uses argumentation to "explain" dominant decisions under certainty; on the other side, [10] uses argumentation to deal with conflicts and incomplete information for decisions under strict uncertainty, and a sceptical semantics to mirror a minimax decision-theoretic criterion.

IV. ARGUMENTATION FOR CONFLICT RESOLUTION

Complex multi-agent systems are composed of heterogeneous agents with different, possibly incomplete beliefs and

[4] Note that, in ABA, assumptions are used as premises of rules to represent incomplete information that can be "completed" by making suitable assumptions. Also, assumptions are used to render rules defeasible, thus paving the way to resolving inconsistencies.

different, possibly conflicting desires. Conflicts may thus arise amongst agents for (at least) two reasons. Firstly, agents may make different assumptions to fill gaps in their beliefs, where some of these assumptions may be incorrect, and decide on incompatible, conflicting actions due to misinformation. Secondly, even if agents share the same information, they may still disagree if they have conflicting desires.

Due to ABA's suitability in dealing with incomplete and conflicting information, agents' beliefs and desires can be represented in ABA [42], [43]. Following an existing trend of work in argumentation for conflict resolution (e.g. see [44]), in [45] we use ABA to resolve conflicts between *two agents*. These conflicts arise when the agents have different goals, g_1 and g_2, and different decisions, d_1 and d_2, having those goals as respective outcomes, according to their respective individual belief bases. These bases are assumed to be represented as ABA frameworks. Here, rules are used to represent beliefs about the achievement of desires as well as factual information.

For both agents, the goal belongs to a conflict-free extension of the agent's ABA framework, extended with the agent's decision. The agents' objective is to resolve the conflict, by agreeing on a common goal g and a common decision d such that g is a variant of both g_1 and g_2. For example, in a service-oriented architecture, if the two agents represent two service requestors from the same organisation, with different requirements but with the shared goal of obtaining some service of a certain type, then

- g_1 may correspond to obtaining a service s_1 of that type, by purchasing it (decision d_1),
- g_2 may correspond to obtaining another service s_2 of the same type, by purchasing it (decision d_2), and
- g may correspond to obtaining a service s of that type, by purchasing it (decision d), where s may be one of s_1 or s_2 or a new service.

Here, the original choice of s_1/s_2 by the agents may be dictated by their lack of knowledge of the other agents requirements or of the availability and characteristics of services. For example, the first agent may know that s_1 fulfils some requirement of the second agent while the latter may be unaware of this. By passing information to the second agent, the first agent may be able to persuade the second to purchase s_1 (in this case s would be s_1). Thus, conflict resolution amounts to identifying a goal that is the outcome of a decision in conflict-free extensions of either belief bases (ABA frameworks) after the agents have shared factual information (e.g. that s_1 fulfils some requirement).

Alternatively, this conflict resolution can be achieved by "merging" the two ABA frameworks. The merge eliminates misunderstanding between agents, allows to revise the agents' incorrect assumptions and takes into account desires from both agents. To satisfy desires from both agents, the mechanism of concatenation is used to merge rules. Upon a successful concatenation merge, both agents' desires may be satisfied (if they can be satisfied). Details of this approach can be found in [45]. Here, a dialogical counterpart to the merge is also sketched as a more realistic approach to conflict resolution.

V. Argumentation for negotiation

The need for negotiation arises when autonomous agents have conflicting interests/desires but may benefit from cooperation in order to achieve them. In particular, this cooperation may amount to a change of goals (as in conflict-resolution, see section IV) and/or to the introduction of new goals (e.g. for an agent to provide a certain resource to another, even though it may not have originally planned to do so). Typically negotiation involves (fair) compromise.

Argumentation-based negotiation is a particular class of negotiation, whereby agents can provide arguments and justifications as part of the negotiation process [46]. It is widely believed that the use of argumentation during negotiation increases the likelihood and/or speed of agreements being reached [47]. Argumentation can be used to support the decision-making taking place prior to or during negotiation. Moreover, argumentation can be used to conduct negotiation, by supporting the resolution of conflicts giving rise to the need of negotiation and by filling in information gaps and rectifying misinformed beliefs.

In [10] we propose the use of ABA to support decision making under strict uncertainty (as described in section III) prior to the agents engaging in negotiation. The negotiation takes place between a buyer and a seller (e.g. of services) and results in (specific forms of) contracts, taking into account contractual properties and preferences that buyer and seller have. The negotiation is guided by a "minimal concession" strategy that is proven to be in symmetric Nash equilibrium. Adopting this strategy, agents may concede and adopt a less-preferred goal to the one they currently hold (namely a goal with a smaller weight, according to the presentation in section III) for the sake of reaching agreement. Thus, agreement amounts to compromise. This approach has been extended in [48] to incorporate rewards during negotiation. These rewards in turn can be seen as arguments in favour of agreement.

In [11] we study the use of a form of ABA, given in [49], for improved effectiveness of the negotiation process, in particular concerning the number of dialogues and dialogue moves that need to be performed during negotiation without affecting the quality of solutions reached. The focus here is on negotiation of resources in resource reallocation settings. This work complements studies on protocols for argumentation-based negotiation (e.g. [50]) and argumentation-based decision making during negotiation (e.g. [10]) by integrating argumentation-based decision making with the exchange of arguments to benefit the outcome of negotiation. Agents engage in dialogues with other agents in order to obtain resources they need but do not have. Dialogues are regulated by simple communication policies that allow agents to provide reasons (arguments) for their refusals to give away resources; agents use ABA in order to deploy these policies. We assess the benefits of providing these reasons both informally and experimentally: by providing reasons, agents are more effective in identifying a reallocation of resources if one exists and failing if none exists.

VI. Argumentation for trust computing

Computing trust is a problem of reasoning under uncertainty, requiring the prediction and anticipation by an agent (the evaluator) of the future behaviour of another agent (the target). Despite the acknowledged ability of argumentation to support reasoning under uncertainty (e.g. see [16]), only Prade [51], Dondio & Barret [52] and Parsons et al [53] have considered the use of arguments for computing trust in a local trust rating setting. Dondio & Barret [52] propose a set of trust schemes, in the spirit of Walton's argument schemes [54], and assume a dialectical process between the evaluator and the target whereby the evaluator poses critical questions against arguments by the target concerning its trustworthiness. Prade [51] proposes an argumentation-based approach for trust evaluation that is bipolar (separating arguments for trust and for distrust) and qualitative (as arguments can support various degrees of trust/distrust). Parsons et al [53] define an argumentation logic where arguments support measures of trust, e.g. qualitative measures such as "very reliable" or "somewhat unreliable".

There are several non-argumentation based methods to model the trust of the evaluator in the target. Sabater and Sierra [55] classify approaches to trust as either "cognitive", based on underlying beliefs, or "game-theoretical", where trust values correspond to subjective probabilities and can be modelled by uncertainty values, Bayesian probabilities, fuzzy sets, or Dempster-Shafer belief functions. The latter approach is predominant nowadays for trust computing. However, Castelfranchi and Falcone [56] argue against a purely game-theoretic approach to trust and in favour of a cognitive approach based upon a mental model of the evaluator, including goals and beliefs. Moreover, some works (e.g. [57]) advocate the need for and benefits of hybrid trust models, combining both the cognitive and game-theoretical approach.

In recent work [58], we propose a hybrid approach for constructing Dempster-Shafer belief functions modeling the trust of the evaluator in the target by combining statistical information concerning the past behaviour of the target and arguments concerning the target's expected behaviour. These arguments are built from current and past contracts between evaluator and target, and are integrated with statistical information proportionally to their validity. Concretely, in a service-oriented setting, the statistics are drawn from past behaviour of the target in the delivery of agreed services and, following [59], a classification of this behaviour as "good" (the service was delivered as agreed), "bad" (the service was not delivered as agreed) or "inappreciable" (the evaluator cannot judge the delivery of the service). Clearly, the more "good" behaviour the target has shown in the past the more likely the evaluator will be to trust it. The arguments are drawn from contracts regulating the delivery of services, as follows:

- a forecast argument supporting the claim of not trusting the target (as far as delivering a service is concerned) if there is no guarantee on the quality of that service in the form of a written contract clause;
- a forecast argument supporting the claim of trusting the target if there exists a guarantee in the form of a contract clause;
- an argument attacking the forecast argument for trusting the target if the target has in the past "most often" violated existing contract clauses.

The applicable arguments and attacks form an abstract argumentation framework (see section II). They are combined with the statistics in accordance to their strength, measured using the method of [19].

This method of measuring trust extends a standard method for trust [59] that relies upon the statistical information only. The two methods have identical predictive performance when the evaluator is highly "cautious", but the hybrid method gives a significant increase when the evaluator is not or is only moderately "cautious". Moreover, with the hybrid method, target agents are more motivated to honour contracts than when trust is computed on a purely statistical basis. The comparison between the two methods is performed within a simulated setting (see [58] for details).

VII. Conclusion

Argumentation, initially studied in philosophy and law, has been researched extensively in computing in the last decade, especially for inference, decision making and decision support, dialogue, and negotiation.

This paper has summarised some of the uses of argumentation

(i) to help agents to make decision, either in isolation (by evaluating pros and cons of conflicting decisions) or in an open and dynamic environment (by assessing the validity of information they become aware of)
(ii) to support negotiation and
(iii) conflict resolution amongst agents, and
(iv) to improve the assessment of the trustworthiness of agents in contract-regulated interactions.

The paper has focused on contributions to (i)–(iv) developed within two European initiatives: the EC-funded ARGUGRID project and the COST-funded Agreement Technologies action.

Acknowledgment

The author would like to thank the Agreement Technology COST action IC 0801 for partially sponsoring this work.

References

[1] C. I. Chesñevar, A. G. Maguitman, and R. P. Loui, "Logical models of argument," *ACM Computing Surveys*, vol. 32, no. 4, pp. 337–383, 2000.

[2] T. Bench-Capon and P. E. Dunne, "Argumentation in artificial intelligence," *Artificial Intelligence*, no. 171, pp. 619–641, 2007.

[3] I. Rahwan and P. McBurney, "Guest editors' introduction: Argumentation technology," *IEEE Intelligent Systems*, vol. 22, no. 6, pp. 21–23, 2007.

[4] P. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games," *Artificial Intelligence*, vol. 77, no. 2, pp. 321–257, 1995.

[5] A. Bondarenko, P. M. Dung, R. A. Kowalski, and F. Toni, "An abstract, argumentation-theoretic approach to default reasoning," *Artificial Intelligence*, vol. 93, no. 1–2, pp. 63–101, 1997.

[6] P. Dung, R. Kowalski, and F. Toni, "Assumption-based argumentation," in *Argumentation in AI*, I. Rahwan and G. Simari, Eds. Springer-Verlag, 2009, pp. 199–218.

[7] F. Toni, M. Grammatikou, S. Kafetzoglou, L. Lymberopoulos, S. Papavassileiou, D. Gaertner, M. Morge, S. Bromuri, J. McGinnis, K. Stathis, V. Curcin, M. Ghanem, and L. Guo, "The ArguGRID platform: An overview," in *Proceedings of Grid Economics and Business Models, 5th International Workshop (GECON 2008)*, ser. Lecture Notes in Computer Science, J. Altmann, D. Neumann, and T. Fahringer, Eds., vol. 5206. Springer, August 2008, pp. 217–225.

[8] P.-A. Matt, F. Toni, T. Stournaras, and D. Dimitrelos, "Argumentation-based agents for eprocurement," in *Proceedings of the 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)- Industry and Applications Track*, M. Berger, B. Burg, and S. Nishiyama, Eds., 2008, pp. 71–74.

[9] P.-A. Matt, F. Toni, and J. Vaccari, "Dominant decisions by argumentation agents," in *Proceedings of the Sixth International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2009), affiliated to AAMAS 2009*, ser. Lecture Notes in Computer Science, P. McBurney, I. Rahwan, S. Parsons, and N. Maudet, Eds., vol. 6057. Springer, 2010, pp. 42–59.

[10] P. M. Dung, P. M. Thang, and F. Toni, "Towards argumentation-based contract negotiation," in *Proceedings of the 2nd International Conference on Computational Models of Argument (COMMA'08)*, ser. Frontiers in Artificial Intelligence and Applications, P. Besnard, S. Doutre, and A. Hunter, Eds., vol. 172. IOS Press, 2008, pp. 134–146.

[11] A. Hussain and F. Toni, "On the benefits of argumentation for negotiation - preliminary version," in *Proceedings of 6th European Workshop on Multi-Agent Systems (EUMAS-2008)*, 2008.

[12] J. McGinnis, K. Stathis, and F. Toni, "A formal framework of virtual organisations as agent societies," *EPTCS*, vol. 16, p. 1, 2010.

[13] F. Toni, "Argumentative KGP agents for service composition," in *Proc. AITA08, Architectures for Intelligent Theory-Based Agents, AAAI Spring Symposium*, M. Balduccini and C. Baral, Eds. Stanford University, 2008.

[14] S. Ossowski, "Coordination and agreement in multi-agent systems," in *Proceedings of Cooperative Information Agents XII, 12th International Workshop (CIA 2008)*, ser. Lecture Notes in Computer Science, M. Klusch, M. Pechoucek, and A. Polleres, Eds., vol. 5180. Springer, 2008, pp. 16–23.

[15] ——, "Coordination in multi-agent systems: Towards a technology of agreement," in *Proceedings of Multiagent System Technologies, 6th German Conference (MATES 2008)*, ser. Lecture Notes in Computer Science, R. Bergmann, G. Lindemann, S. Kirn, and M. Pechoucek, Eds., vol. 5244. Springer, 2008, pp. 2–12.

[16] P. Krause, S. Ambler, M. Elvang-Gøransson, and J. Fox, "A logic of argumentation for reasoning under uncertainty," *Computational Intelligence*, vol. 11, pp. 113–131, 1995.

[17] P. Dung, P. Mancarella, and F. Toni, "Computing ideal sceptical argumentation," *Artificial Intelligence*, vol. 171, no. 10–15, pp. 642–674, 2007.

[18] M. Caminada, "Semi-stable semantics," in *Proceedings of 1st International Conference on Computational Models of Argument (COMMA'06)*, ser. Frontiers in Artificial Intelligence and Applications, P. E. Dunne and T. J. M. Bench-Capon, Eds., vol. 144. IOS Press, 2006, pp. 121–130.

[19] P.-A. Matt and F. Toni, "A game-theoretic measure of argument strength for abstract argumentation," in *Proceedings of 11th European Conference on Logics in Artificial Intelligence (JELIA 2008)*, ser. Lecture Notes in Computer Science, S. Hölldobler, C. Lutz, and H. Wansing, Eds., vol. 5293, 2008, pp. 285–297.

[20] D. Gaertner and F. Toni, "CaSAPI - a system for credulous and sceptical argumentation," in *Proceedings of the International Workshop on Argumentation and Non-Monotonic Reasoning (ArgNMR 2007), affiliated to LPNMR 2007*, G. Simari and P. Torroni, Eds., 2007.

[21] ——, "On computing arguments and attacks in assumption-based argumentation," *IEEE Intelligent Systems, Special Issue on Argumentation Technology*, vol. 22, no. 6, pp. 24–33, November/December 2007.

[22] ——, "Hybrid argumentation and its properties," in *Proceedings of the Second International Conference on Computational Models of Argument (COMMA'08)*, ser. Frontiers in Artificial Intelligence and Applications, P. Besnard, S. Doutre, and A. Hunter, Eds., vol. 172. IOS Press, 2008, pp. 183–195.

[23] J. Doyle and R. H. Thomason, "Background to qualitative decision theory," *AI Magazine*, vol. 20, no. 2, pp. 55–68, 1999.

[24] S. French, *Decision theory: an introduction to the mathematics of rationality*. Ellis Horwood, 1987.

[25] J. Pearl, "From conditional oughts to qualitative decision theory," in *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, D. Heckerman and E. H. Mamdani, Eds. Morgan Kaufmann, 1993, pp. 12–22.

[26] D. Poole, "Probabilistic Horn abduction and Bayesian networks," *Artificial Intelligence*, vol. 64, no. 1, pp. 81–129, 1993.

[27] B. Bonet and H. Geffner, "Arguing for decisions: A qualitative model of decision making," in *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI-96)*, E. Horvitz and F. V. Jensen, Eds. Morgan Kaufmann, 1996, pp. 98–105.

[28] D. Dubois, H. Fargier, and P. Perny, "Qualitative decision theory with preference relations and comparative uncertainty: An axiomatic approach," *Artificial Intelligence*, vol. 148, pp. 219–260, 2003.

[29] J. Fox, P. Krause, and M. Elvang-Gøransson, "Argumentation as a general framework for uncertain reasoning," in *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, D. Heckerman and E. H. Mamdani, Eds. Morgan Kaufmann, 1993, pp. 428–434.

[30] J. Fox and S. Parsons, "On using arguments for reasoning about actions and values," in *Working Papers of the AAAI Spring Symposium on Qualitative Preferences in Deliberation and Practical Reasoning*, J. Doyle and R. H. Thomason, Eds., 1997, pp. 55–63.

[31] L. Amgoud, "A unified setting for inference and decision: An argumentation-based approach," in *proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI'05)*. AUAI Press, 2005, pp. 26–33.

[32] L. Amgoud and H. Prade, "Making decisions from weighted arguments," in *Decision theory and multi-agent planning*, G. D. Riccia, D. Dubois, R. Kruse, and H.-J. Lenz, Eds. Springer, 2006, pp. 1–14.

[33] S. Parsons, "Normative argumentation and qualitative probability," in *Proceedings of the First International Joint Conference on Qualitative and Quantitative Practical Reasoning (ECSQARU-FAPR'97)*, ser. Lecture Notes in AI, D. M. Gabbay, R. Kruse, A. Nonnengart, and H. J. Ohlbach, Eds., vol. 1244. Springer, jun 9–12 1997, pp. 466–480.

[34] L. Amgoud and H. Prade, "Using arguments for making decisions: A possibilistic logic approach," in *Proceedings of the 20th Conference of Uncertainty in Artificial Intelligence (UAI'04)*, D. M. Chickering and J. Y. Halpern, Eds. AUAI Press, 2004, pp. 10–17.

[35] H. Prakken, "Combining sceptical epistemic reasoning with credulous practical reasoning," in *Proceedings of 1st International Conference on Computational Models of Argument (COMMA'06)*, ser. Frontiers in Artificial Intelligence and Applications, P. Dunne and T. J. M. Bench-Capon, Eds., vol. 144. IOS Press, 2006, pp. 311–322.

[36] I. Rahwan and L. Amgoud, "An argumentation-based approach for practical reasoning," in *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'06)*, H. Nakashima, M. P. Wellman, G. Weiss, and P. Stone, Eds. ACM, 2006, pp. 347–354.

[37] J. Fox, N. Johns, C. Lyons, A. Rahmanzadeh, R. Thomson, and P. Wilson, "PROforma: a general technology for clinical decision support systems," *Computer Methods and Programs in Biomedicine*, vol. 54, no. 10–15, pp. 59–67, 1997.

[38] S. Modgil and P. Hammond, "Decision support tools for clinical trial design," *Artificial Intelligence in Medicine*, vol. 27, no. 2, pp. 181–200, 2003.

[39] M. Morge and P. Mancarella, "The hedgehog and the fox. An argumentation-based decision support system," in *Proceedings of the 4th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS'07), affiliated to AAMAS*, ser. Lecture Notes in Computer Science, I. Rahwan, S. Parsons, and C. Reed, Eds., vol. 4946. Springer, 2008, pp. 114–131.

[40] F. S. Nawwab, T. J. M. Bench-Capon, and P. E. Dunne, "A methodology for action-selection using value-based argumentation," in *Proceedings of the Second International Conference on Computational Models of Argument (COMMA'08)*, ser. Frontiers in Artificial Intelligence and

Applications, P. Besnard, S. Doutre, and A. Hunter, Eds., vol. 172. IOS Press, 2008, pp. 264–275.

[41] P.-A. Matt, "Argumentation as a practical foundation for decision theory," Ph.D. dissertation, Department of Computing, Imperial College London, 2010.

[42] F. Toni, "Assumption-based argumentation for selection and composition of services," in *Proceedings of the 8th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA VIII)*, ser. Lecture Notes in AI, F. Sadri and K. Satoh, Eds., vol. 5056. Springer, 2008, pp. 231–247.

[43] ——, "Assumption-based argumentation for closed and consistent defeasible reasoning," in *Proceedings of the First International Workshop on Juris-informatics (JURISIN 2007), in association with The 21th Annual Conference of The Japanese Society for Artificial Intelligence (JSAI2007)*, ser. Lecture Notes in Computer Science, K. Satoh, A. Inokuchi, K. Nagao, and T. Kawamura, Eds., vol. 4914. Springer, 2008, pp. 390–402.

[44] L. Amgoud and S. Kaci, "An argumentation framework for merging conflicting knowledge bases," *International Journal on Approximate Reasoning*, vol. 45, no. 2, pp. 321–340, 2007.

[45] X. Fan, F. Toni, and A. Hussain, "Two-agent conflict resolution with assumption-based argumentation," in *Proceedings of the Third International Conference on Computational Models of Argument (COMMA'10)*, P. Baroni, F. Cerutti, M. Giacomin, and G. Simari, Eds. IOS Press, 2010, pp. 231–242.

[46] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: Prospects, methods and challenges," *Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001.

[47] I. Rahwan, S. D. Ramchurn, N. R. Jennings, P. McBurney, S. Parsons, and L. Sonenberg, "Argumentation-based negotiation," *The Knowledge Engineering Review*, vol. 18, no. 4, pp. 343–375, 2004.

[48] P. M. Dung, P. M. Thang, and N. D. Hung, "Argument-based decision making and negotiation in e-business: Contracting a land lease for a computer assembly plant," in *Proceedings of the 9th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA IX)*, ser. Lecture Notes in Computer Science, M. Fisher, F. Sadri, and M. Thielscher, Eds., vol. 5405. Springer, 2009, pp. 154–172.

[49] A. Hussain and F. Toni, "Assumption-based argumentation for communicating agents," in *Proceedings of "The Uses of Computational Argumentation", AAAI Fall Symposium*, T. Bench-Capon, S. Parsons, and H. Prakken, Eds. Stanford University, 2009.

[50] J. van Veenen and H. Prakken, "A protocol for arguing about rejections in negotiation," in *Proceedings of the 2nd International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2005), affiliated to AAMAS 2005*, ser. Lecture Notes in Computer Science, S. Parsons, N. Maudet, P. Moraitis, and I. Rahwan, Eds., vol. 4049. Springer, 2006, pp. 138–153.

[51] H. Prade, "A qualitative bipolar argumentative view of trust," in *Proceedings of the 1st International Conference on Scalable Uncertainty Management (SUM 2007)*, ser. Lecture Notes in Computer Science, H. Prade and V. S. Subrahmanian, Eds., vol. 4772. Springer, 2007, pp. 268–276.

[52] P. Dondio and S. Barrett, "Presumptive selection of trust evidences," in *Proceedings of the 6th International Conference on Autonomous Agent and Multi-Agent Systems (AAMAS'07)*, E. H. Durfee, M. Yokoo, M. N. Huhns, and O. Shehory, Eds., 2007, p. 166.

[53] S. Parsons, P. McBurney, and E. Sklar, "Reasoning about trust using argumentation: A position paper," in *Proceedings of the Seventh International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2010), affiliated to AAMAS 2010*, 2010.

[54] D. Walton, C. Reed, and F. Macagno, *Argumentation Schemes*. Cambridge University Press, 2008.

[55] J. Sabater and C. Sierra, "Review on computational trust and reputation models," *Artifical Intelligence Review*, vol. 24, no. 1, pp. 33–60, 2005.

[56] C. Castelfranchi and R. Falcone, "Trust is much more than subjective probability: Mental components and sources of trust," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS-33)*, 2000.

[57] E. Staab and T. Engel, "Combining cognitive with computational trust reasoning," in *Proceedings of the 11th International Workshop on Trust in Agent Societies (AAMAS-TRUST'08)*, ser. Lecture Notes in Computer Science, R. Falcone, K. S. Barber, J. Sabater-Mir, and M. P. Singh, Eds., vol. 5396. Springer, 2008, pp. 99–111.

[58] P.-A. Matt, M. Morge, and F. Toni, "Combining statistics and arguments to compute trust," in *Proceedings of the 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, W. van der Hoek and G. A. Kaminka, Eds., 2010.

[59] B. Yu and M. P. Singh, "Distributed reputation management for electronic commerce," *Computational Intelligence*, vol. 18, no. 4, pp. 535–549, 2002.

ISBN 978-83-60810-27-9
ISSN 1896-7094

An agent based planner for including network QoS in scientific workflows

Zhiming Zhao, Paola Grosso, Ralph Koning, Jeroen van der Ham, Cees de Laat
System and Network Engineering research group
Informatics Institute, University of Amsterdam
Science Park 904, 1098XH, Amsterdam, the Netherlands
{Z.Zhao, P.Grosso, R.Koning, vdHam, C.T.A.M.Delaat}@uva.nl

***Abstract*—Advanced network infrastructure plays an important role in the e-Science environment to provide high quality connections between largely distributed data sensors, and computing and storage elements. However, the quality of the network services has so far rarely been considered in composing and executing scientific workflows. Currently, scientific applications tune the execution quality of workflows neglecting network resources, and by selecting only optimal software services and computing resources. One reason is that IP-based networks provide few possibilities for workflow systems to manage the service quality, and limit or prevent bandwidth reservation or network paths selection. We see nonetheless a strong need from scientific applications, and network operators, to include the network quality management in the workflow systems.**

Novel network infrastructures open up new possibilities in network tuning at the application level. In this position paper, we discuss our vision on this issue and propose an agent based solution to include network resources in the loop of workflow composition, scheduling and execution when advanced network services are available. We present the first prototype of our approach in the context of the CineGrid project.

***Index Terms*—Quality of Service, Semantic web, Advanced Network, Scientific Workflow, e-Science**

I. Introduction

ADVANCED network infrastructure plays an important role in the e-Science environment to provide high quality connections between largely distributed data sensors, and computing and storage elements; by enabling large scale data movement between different devices and nodes, applications which require extremely large scale data movement can be enabled. However, The quality of the network connections and services has rarely been taken into account by current scientific workflow management systems, because 1) traditional IP-based networks provide limited reservation capability for workflow engines; 2) the existing e-Science applications assume available network connections as non-changeable services, and seek customized solutions at software level to optimise computing processes and data storage; and 3) the existing applications mainly consider the functionality of the e-Science services, and limited support has been provided for including (network) quality requirements for the services in the composition, enactment and execution of workflows.

Still advanced network services are essential for many scientific applications, where data transmission delays become a bottleneck of the global performance of the application [1], [2] . Without the quality guarantee at the network level, the global workflow quality requirements can not be assured. Several strategies have been tried to improve the workflow performance in these cases, such as caching data in a closer location to the computing element [3], [4], or reducing the load of computing tasks by reusing the previous computed results [5]. However, for applications that require data streams from remote data sensors, such as in [6], those solutions will not be sufficient, and advanced network services are crucial to accelerate large data transfer between the distributed application components.

The recent emergence of advanced network infrastructures for e-Science enables tuning of network performance at the application level. For instance, the hybrid network architecture [7] can provide connections on different layers based on the same physical fibre; this allows applications to reserve dedicated circuits for transferring very large quantity data. By including network resources in the scheduling loop, the high level application gets an extra opportunity to optimize execution and improve performance.

These new opportunities come with some challenges. Not all network infrastructures provide the network services for reserving specific connections or allocating network bandwidth; the service invocation in different network domains is often proprietary and not easily extensible, and makes request for network service provisioning across sites difficult [8]; scheduling network resources requires knowledge on the current state of the network, which implies the existence of a sophisticated monitoring system.

In this paper, we discuss the challenges in including network resources in scientific workflows, and propose an agent based architecture to accomplish this goal and realize application level quality tuning of network resources. We apply this in the context of an ongoing project, CineGrid [9]. The paper is organized as follows; first, we will review the state of the art in this field, and then present our solution and the first prototype of our system.

II. Resource QoS in scientific workflow

The quality of a service is often expressed as set of quality attributes. Sabata presented a QoS taxonomy in which QoS attributes are grouped into *metrics* and *polices* [10]. The metrics include quantifiable attributes for measuring application performance (namely performance metrics), security (security level) and importance, while the polices refer to the qualitative attributes for specifying service level and management polices.

A QoS attribute is typically represented as a triple (attribute name, value, the range of the value), and timeliness and accuracy are the most often cited performance metrics. A description language such as QML [11] aims at a generic model for specifying different categories of quality attributes.

Since the development of services are highly application dependent, and the quality description models are often proposed by different society, semantic web based technologies have been widely used for describing QoS attributes [12]–[14], and to integrate different QoS models [15]. While composing a workflow, services at different abstraction level will naturally have different model on the quality; a semantic model for matching these quality attributes and mapping them will be essential.

From the perspective of a scientific workflow life cycle QoS is relevant to four aspects: composition, selection, execution control and provenance. In the following sections we briefly review the existing work on each issue.

A. QoS aware workflow composition

A workflow composition process that is QoS-aware must: 1) compose a service of the highest quality and 2) determine the quality of the composition process itself. The first goal is achieved by computing the global quality starting from the QoS attributes of constituting services [16]. Graph reduction is a widely used approach [17]; a pre-defined set of logic patterns define certain reduction rules which can be used to simplify the logical dependencies among constituting services. From the reduction rules, the quality parameters are computed; for instance the computing time of two sequentially connected services is computed as the sum of the quality of each of them, the computing time of a two parallel services is computed as the maximal one from them. The second goal requires modelling the quality attributes of the semantic links between services, the composition quality of the workflow can then be evaluated by the semantic fit and the reliability of the selected service in the workflow [16].

B. QoS aware service selection

Searching for suitable services from available resources is a basic procedure in composing a workflow. QoS aware service selection implies two steps: properly formulating the requirements and selecting resources that meet these requirements. Rosenberg proposed a QoS enabled description language, the Vienna composition language (VCL) [18], to specify an abstract flow for workflow composition. The VCL defines an abstract workflow as four parts: feature definition, feature constraints, global constraints and the business protocol (the desired workflow language). The feature constraints and global constraints include both functional constraints and QoS attributes. The problem of resource selection has been formulated differently. A commonly used formulation is *shortest path finding in a weighted graph*, in which the available services are represented as a directed graph according to the service types, and the graph nodes are labelled by the quality attributes of the service [19]. Well known shortest path finding algorithms include Bellman-Ford and Dijkstra's. These algorithms exhibit optimal performance because of their greedy search strategy and avoid backtracking operations during the search; however, the minimal cost path found by the algorithms is often not the most optimal solution if there are multiple constraints on the quality attributes. Therefore, the problem has also been formulated as a multi constraint optimal path problem [20], or multi objective optimization problem. Ant colony optimization (ACO) is a meta heuristic search approach proposed in [21], [22] for discovering minimum cost path in a graph, and for solving NP-hard combinatorial optimization problems. Fang et al, [23] applied ACO in service selection and proposed a multi objective ACO approach which can simultaneously optimize several objectives. Genetic algorithm in searching optimal paths, and constraint programming or Integer programming methods are also widely used for the multi objective optimization problem.

C. QoS aware workflow execution

Workflow execution is the mapping of workflow processes to underlying computing resources and the scheduling of the execution sequence. Task based scheduling is a straightforward approach, in which the workflow tasks are submitted to the local manager of the computing infrastructure. Several researchers have instead proposed a workflow level scheduling that takes into account future task performance [24]; this approach will achieve higher performance and better resource utilization than only using local resource managers. Multi objective optimizations are widely used to formulate the problem of QoS aware scheduling. Avanes proposed a constraint programming based approach to search for best match between workflow requirements and the available computing resources [25]. The basic idea is to describe the quality requirements and resource dependencies as constraints by partitioning the workflow into different parts based on its patterns and their QoS requirements. One of the contributions from Avanes work is that the network dynamics has been also included in the procedure of constraint resolving. Resource provision plays an important role to improve the fault tolerance and the performance of the workflow [26]. Basically, provisioning can be either static or dynamic. Advance reservation is a typical static provisioning mechanism, and several batch based schedulers support it. Based on the high level quality requirements, the workflow engine reserves computing resources and time slots from the Grid resource manager. One of the disadvantages of static provisioning is its overhead on the total cost for computing the workflow. To improve this, Raicu et al. [27] proposed multi level scheduling strategies, in which the application level scheduler is able to interact with the low level resource manager to tune the requirements at runtime. This approach introduces a dynamic component in the provisioning process.

D. QoS and provenance

The provenance service tracks the events occurred in the workflow execution, and allows scientists to trace the evolution of data computed in the workflow and to obtain insights in the experiment processes. Moreover, provenance data can also be used to debug errors of the workflow execution and optimize the workflow design. The Open Provenance Model (OPM) [28] emerges as a standard model to represent workflow provenance information. Including QoS information of the workflow processes and the execution in the provenance model allows scientists to analyze the quality of the services and the workflow scheduling. In [29], the provenance service is provided by a QoS aware middleware, which records the changes of the service quality as events. Evaluating trust and reliability of the provenance data itself has also been discussed in the literature [30]. However, research on the provenance model which includes the QoS information of the workflow processes is still in its very early stage.

III. Problem and challenges: network QoS and scientific workflows

Our research interest focuses on the inclusions of the network quality of service in the high level e-Science workflows. Workflow systems have been recognized as an important tool to manage the invocation of lower level network services and the security related routines [31]. In SC09, the VLAM workflow system [32] demonstrated allocation of workflow modules over specialized network connections. Several e-Science environments already recognized the importance of the network resources reservation in the context of workflow scheduling, for instance in [33]. When including network resources QoS in a scientific workflow, we have to solve several issues:

1) We need a representation and mapping mechanism between the user level requirements and the quality model of underlying network resources. When mapping high level workflow processes to low level resources, the abstract quality descriptions of the workflow also have to be translated to the proper quality constraints on the underlying resources.
2) A good network resource monitoring system is essential in using QoS in the system design, execution and evaluation [34]. A workflow description composed from carefully selected resources or services does not imply that the actual execution will achieve the desired performance. The quality guarantee of the workflow execution is a dynamically adapting procedure between the workflow engine and the actual state of the resources involved in the workflow execution, unless one can guarantee a priori that all the resources perform at their best state. A problem is that monitoring quality metrics of the network resources in the workflow context is also in its initial state. Truong proposed a monitoring framework for Grid services, which monitors the resources from four different levels: machine, network path, middleware and application [34], and derives the QoS value for the reliability and availability of the machines, network paths and middleware from the monitoring.
3) We need to optimize the negotiation between the workflow engine and the underlying network resource manager, for both resource provisioning and tuning the scheduling of workflow (or replacing workflow components). This is made more difficult by the ad-hoc APIs adopted by the different network providers, which prevent a generic solution.
4) In a scientific workflow we need to schedule multi level resources to support real time applications. Solving multi-level constraints between heterogeneous resources requires different optimization searching technologies in the problem space and it is time consuming compared to the state updates of the underlying resources. For instance, discovering suitable network paths often involves not only searching connected intra-domain paths between network devices, but also inter-domain exchange of topologies. Further, the evolution of the underlying network state make scheduling more difficult.
5) We need to adapt the provenance model to include the (network) quality of service information, and to let the logging facility obtain such information and record it in the database. The current provenance model mainly focuses on the evolution of the data, and uses certain partial relation to indicate the dependencies of the data process; it does not explicitly include the quality of actual services when the data was produced. Such information can be very important for improving the scheduling strategies utilised by the workflow engine.

We propose a QoS ware planner which covers the life-cycle of workflow not only at composition, but also scheduling and execution. Our system tackles the issues we just summarised above.

IV. Design requirements and system design

We had two alternatives when we looked at the inclusions of QoS aware functionalities in a scientific workflow system: 1) re-engineer the functional components of an existing workflow system to include the QoS support, or 2) consider the existing workflow systems as legacy systems, and provide QoS support as plugged components to the system. Each alternative exhibits advantages and disadvantages. In the context of our research, we chose the second approach; one of the motivations is that generic functional components can be encapsulated as reusable tools which can serve different specific scientific workflow systems to get QoS support.

A. Design requirements

We can highlight three scenarios where network QoS support can be applied: QoS aware resource selection, resource provisioning and quality assured workflow execution. The designed system thus needs to meet the following functional requirements:

1) The system must include QoS aware resource discovery and selection of network resources. Network resources and the quality attributes are necessarily described, and a search tool is provided to check the suitable resources based on the input requirements.
2) The system should be able to generate a resource provisioning plan for selected resources based on the input requirements. The plan is made based on the provisioning services that the available network infrastructure provides.
3) The system should be able to generate workflows handling large data movement between network resources with guaranteed data transfer quality, and wrap the generated workflow as a service which can be executed standalone or included in a third party workflow.
4) At runtime, the system should provide monitoring services to track the actual quality of the network resources. It should also provide interfaces for third party workflows to invoke during their provenance procedure to record all the runtime information.

B. Agent based technologies

The Agent Oriented (AO) methodology complements the object and component oriented methods with knowledge related notions to manage system complexity [35], and emerges as an important modelling and engineering approach for constructing complex systems, such as workflow management systems. The concept of *agents* originated in the mid-1950s as *a 'soft robot' living and doing its business within the computer's world* [36]. Wooldridge distinguished three types of agent architectures: deliberative, reactive and hybrid [37]. The difference between the deliberative and reactive architectures is that the former incorporates a detailed and accurate symbolic description of the external world and uses sophisticated logic to reason about the activities, while the latter one only implements a stimulus-reaction scheme. Reactive architectures are easier to implement but lack a subtle reasoning capability. Hybrids of the two schemes are commonly used. During the past two decades, agent based models, in particular reactive models, have been applied as an advanced technology in modelling and constructing complex system. Agent frameworks, such as FIPA [38], abstract the structure of basic agents and define standardized communication languages to represent interactions between agents, which facilitate the implementation of agent based applications.

JADE (Java Agent DEvelopment Framework) is a free software and distributed by Telecom Italy [38]. Fully implemented in java, Jade realizes a FIPA compliant multi agent middleware. In our project, a number of reasons motivate us to choose JADE as the implementation framework. First, the Jade platform can be distributed across machines and the configuration can be controlled via a remote GUI. The Java language makes the development portable; the Jade framework allows agents move from one machine to another at runtime. Moreover, being compliant to the FIPA protocol, Jade provides a standard architecture for scheduling agent activities, which makes the inclusion of high level functionality easy, e.g., adding a Prolog module for activity reasoning. Finally, the ontology enabled agent communication between agents promotes seamless integration between the semantic network description, QoS aware searching modules, underlying models of workflow descriptions, and other necessary functional components of our system.

C. An agent based QoS workflow planner

We propose an agent based architecture, composed of a *QoS aware workflow planner (QoSWP)* and five more agents: *a Resource Discovery Agent (RDA), a Workflow Composition Agent (WCA), a Resource Provisioning Planner (RPP), a QoS Monitor Agent (QMA)* and *a Provenance Service Agent (PSA)*. Fig. 1 illustrates a conceptual schema of our agent system.

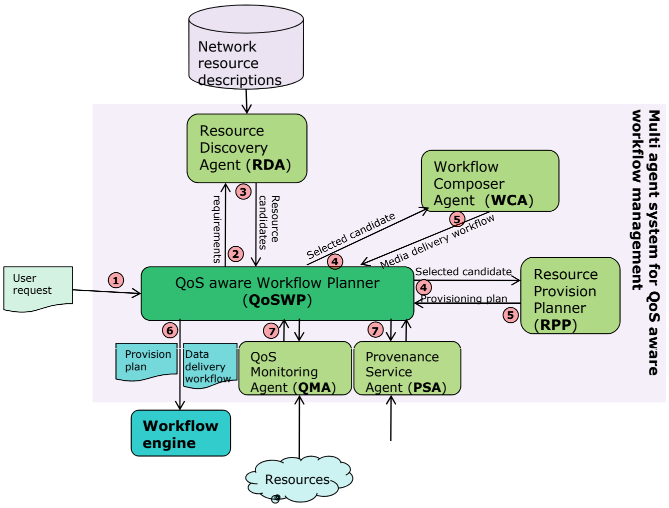

Network resource descriptions
Resource Discovery Agent (RDA)
Workflow Composer Agent (WCA)
Multi agent system for QoS aware workflow management
requirements
Resource candidates
3
2
Selected candidate
5
Media delivery workflow
4
User request
1
QoS aware Workflow Planner (QoSWP)
Selected candidate
4
Provisioning plan
5
Resource Provision Planner (RPP)
6
7
7
Provision plan
Data delivery workflow
QoS Monitoring Agent (QMA)
Provenance Service Agent (PSA)
Workflow engine
Resources

Fig. 1. An agent based solution to adaptive QoS aware workflow management.

The QoSWP coordinates the other agents to select suitable services, to propose optimal network connections between the services, and to create the necessary scripts for the workflow engine to invoke the requested services. A typical use case scenario will illustrate the role of each component (see Fig.1). The QoSWP receives the request for data process services and the service requirements from the user (step1). After that, the RDA reads the description of the resources and the network topologies from the registry, and searches suitable data sources and destinations, and network paths between them (step2). The RDA returns a list of qualified candidates, and sorts them based on the quality metrics of each candidate (step3). From the candidates, the QoSWP selects the best one, and request WCA and RPP to generate a resource provisioning plan and a data transfer workflow (step4 and step5), both of which will be executed by the workflow engine (step6). At run time, the QMA monitors the actual state of the resources and checks whether the global quality required by the workflow is satisfied (step7). Based on the states updated by the QMA, the QoSWP decides whether the resources of the workflow should be adapted. The provenance service records events in the resources provisioning, allocation, and combine the actual state of the quality attributes with the log data (step7).

V. Prototype and Use Case

The research is conducted in the context of CineGrid. An important mission of the CineGrid project is to provide a dedicated network environment to connect distributed parties from different domains to share large quantities of very-high-quality digital media, such as the high definition video material used in the movie industry.

We are prototyping our ideas using a small portion of the CineGrid infrastructure as a test bed. Four locations in Amsterdam host CineGrid resources and are connected via dedicated and configurable circuits provided by SURFnet [39]. The results reach beyond the workflow field, and they can be beneficial to understand how advanced network connections enhance the digital media delivery in the academic and education context.

In this section, we will demonstrate how the designed architecture works in a use case, and discuss the technical considerations to prototype the system.

A. A CineGrid use case: QoS guaranteed high quality media on demand

We are focusing on a *digital media delivery on demand* use case: the goal is to retrieve media material from the infrastructure, and request quality guaranteed connections to deliver the data to qualified nodes for further processing, such as playback or visualization. Using the proposed agent framework, the use case will be prototyped as follows:

1) The user uses the schema provided by the system to describe the name and properties of the media, and to specify the quality requirements for visualizing the data.
2) The QoSWP parses the user input and creates queries for the RDA to look for data sources of the media;
3) Based on the input requirements, the RDA looks for the data repositories which contains the required media, and the visualization devices which meet the required playback quality. Then the RDA looks for all possible network paths between the sources and the visualization devices.
4) The RDA returns a list of candidates in the form of (source, destination, path) triplets, and the candidates are ordered based on the quality they provide. The QoSWP selects the best candidate from the list and send it to the RPP and WCA to make resource provisioning plan, and to create a workflow which can deliver the media from the source to the visualisation device, and to play it back in the visualization device.
5) To help RPP and WCA make the provision plan and the workflow compliant to a specific workflow engine, the QoSWP also explicit tells the RPP and WCA what language of the third party engine will use.
6) After receiving the scripts generated by the RPP and WCA, the QoSWP sends them to the third party engine to execute the provisioning plan and the delivery plan.

B. Semantic resource description

The CineGrid community uses semantic web technologies to describe the services, devices and the network topology. The UvA team in the project have developed two ontologies. The Network Description Language (NDL) [40] models the different levels of a network infrastructure: physical, domain, capability, layer and topology[1]. The CineGrid Description Language (CDL) [9] describes the services and resources on top of the network infrastructure[2]. Fig. 2 shows concepts defined in the CDL. Using these two languages, we have described the resources in the research test bed. Four locations (UvA, SARA, De Waag and the Dutch Film and TV institute) in Amsterdam are connected with up to two dedicated switchable 1Gbit/s links, which can be dynamically changed between locations using the openDRAC [41] network provisioning software used by SURFnet.

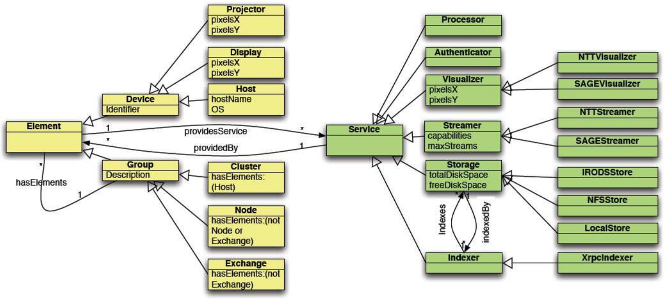


Fig. 2. The schema of the CineGrid description language.

C. QoSAWF: an abstract workflow description language

Based on the experience of early work [18], [42]–[44], we propose an ontology for describing abstract workflows (*qosawf.owl*). It defines the basic concepts of workflow processes, pre/post/execution conditions of the process, media data, and quality attributes, as shown in Fig 3. The description of the user request is described as an object of the *Request* class, and a *Request* consists of one or more *Processes* which can be accessed via the *request_Functionality* property. A *Process* class uses *pre_Condition* and *post_Condition* to indicate the requirements for *Data* the process requires and generates, and the quality for the required data. The *Process* class also uses *execution_Condition* to indicate the service quality for the process. In the current definition, *Data* contains two specific types: *Media* and *Scientific_Data*. And the service quality is modelled as set of *Quality_Attributes*. Based on the QoS taxonomy defined in [10], *Quality_attribute* can more specifically be *Precision*, *Timeliness*, *Reliability* and *Security_Level*. In our case, where the pre and post conditions consist of requirements for data and the data quality, *and_Condition* and *or_Condition* are the two most important types. Using the above ontology, a user is able to formulate a request for obtaining and playing

[1] Available at: http://cinegrid.uvalight.nl/owl/ndl-domain.owl and http://cinegrid.uvalight.nl/owl/ndl-topology.owl

[2] Available at: http://cinegrid.uvalight.nl/owl/cdl/2.0

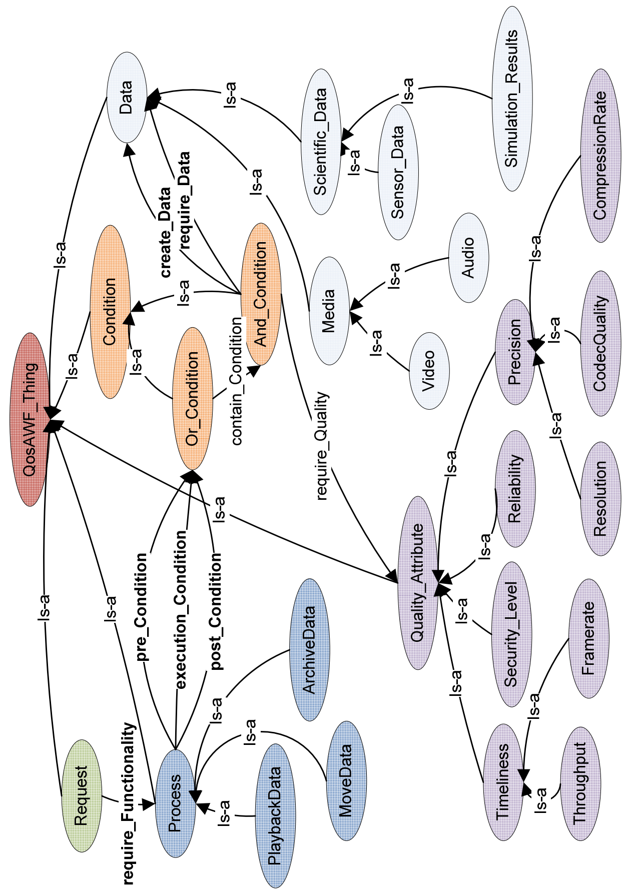


Fig. 3. The concepts defined in the qosawf schema.

back a specific video material with a minimal resolution and frame rate. Fig. 4 shows the graphical representation.

D. First prototype

We have compared different options to realise the resource search mechanism; we have evaluated several Query languages (RQL, RDQL, N3, Versa, SeRQL, SPARQL) and Rule languages (SWRL, Prolog/RDF lib, JESS etc.). We have finally chosen the RDF library of SWI-Prolog; its triple based manipulation interface is easy for the high level language we use to implement the agents (Java); it is also easy to access the runtime state of the triples. Finally, the Prolog language provides effective solutions to realise graph path findings.

The FIPA [38] standards provide a suitable architecture to implement distributed agents in our system. The Agent Communication Language (ACL) allows agents to exchange messages using an explicitly defined semantic schema, which allows seamless integration between agents and remote Ontology knowledge bases.

In the current prototype, the RDA receives the URI of the user requirements and network resources from the QoSWP. The RDA parses the given abstract workflow and searches the resource description; it returns results in the form of (storage host, visualization host, path, quality rank).

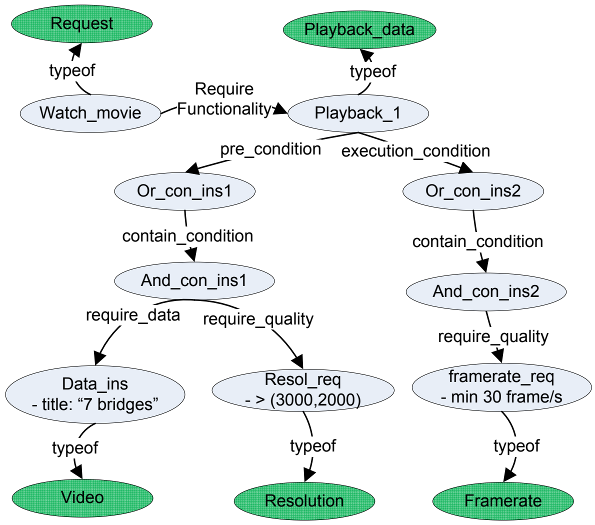


Fig. 4. An example of abstract workflow description.

VI. Summarizing remarks

In this paper, we have reviewed the state of the art of the QoS aware workflow management. We have discussed different technologies for realizing QoS aware resource discovery, workflow composition, scheduling and adaptation. In the review, we can summarize several issues:

1) Semantic technologies play an important role in modelling QoS attributes and mapping quality description between different layers of resources in workflow system.
2) QoS aware service selection has been tackled using different solutions: minimal cost path finding in a weighted graph, multi objective optimization or integer (constraint) programming. For selections with constraints of different QoS attributes, multi objective optimization is a better solutions.
3) The QoS of network resources have been tried in workflow system in different ways: using workflow to manage network services, for interactively creating network connections from the workflow, or scheduling part of the workflow in the resources which have special quality connections. These solutions focuses on runtime part of the network resource allocation, which is often tightly coupled with the specific network service.

Based on the existing work, we have proposed an agent based solution to include QoS aware network resources planning in the loop of workflow composition, provisioning and execution. We have proposed an RDF based schema for describing the data and QoS requirements in an abstract workflow.

The research is conducted in the context of the CineGrid project; however, the results of the research aim to be generic and applicable for any e-Science applications which suffer from the bottleneck of transferring large quantity data over not optimal networks. We will develop the QoSWP as a generic service for legacy workflow engines to support QoS aware network resource selection, provisioning and invocation of network services for transferring data. Moreover we will define a provenance model which includes the causality relations between scientific data and the quality of the utilised resources. This will allow scientists to investigate the dynamics of the underlying resources allocation and to optimize the workflow scheduling strategies. The work described in the paper is still in its early stage. The research will proceed as follows. Firstly, we will prototype the RDA to enhance an early version of the CineGrid portal developed in the UvA for QoS aware network resources discovery. Secondly, we will realise the WCA and RPP; we will use the VLEWF-Bus system [45] as the first test case for the underlying third party workflow engine. Thirdly, we will focus on the QMA and PSA. The OPM will be used as starting model to develop PSA.

Acknowledgment

We would like to thank the National e-Science Research Center and the Dutch national and education network SURFnet, through the GigaPort Research on Network (RoN) project, for sponsoring this research.

References

[1] Osamu Tatebe, Youhei Morita, Satoshi Matsuoka, Noriyuki Soda, and Satoshi Sekiguchi. Grid datafarm architecture for petascale data intensive computing. *Cluster Computing and the Grid, IEEE International Symposium on*, 0:102, 2002.

[2] T. Lavian, J. Mambretti, D. Cutrell, H. Cohen, S. Merrill, R. Durairaj, P. Daspit, I. Monga, S. Naiksatam, S. Figueira, D. Gutierrez, D. Hoang, and F. Travostino. Dwdm-ram: a data intensive grid service architecture enabled by dynamic optical networks. *Cluster Computing and the Grid, IEEE International Symposium on*, 0:762–764, 2004.

[3] Khaled Almi'ani, Javid Taheri, and Anastasios Viglas. A data caching approach for sensor applications. *Parallel and Distributed Computing Applications and Technologies, International Conference on*, 0:88–93, 2009.

[4] David Chiu and Gagan Agrawal. Hierarchical caches for grid workflows. *Cluster Computing and the Grid, IEEE International Symposium on*, 0:228–235, 2009.

[5] Ewa Deelman, James Blythe, Yolanda Gil, Carl Kesselman, Scott Koranda, Albert Lazzarini, Gaurang Mehta, Maria Alessandra Papa, and Karan Vahi. Pegasus and the pulsar search: From metadata to execution on the grid. In *PPAM*, pages 821–830, 2003.

[6] Robert Morris, Jennifer Dungan, and Petr Votava. A workflow model for earth observation sensor webs. *Space Mission Challenges for Information Technology, IEEE International Conference on*, 0:69–76, 2009.

[7] Cees de Laat, Erik Radius, and Steven Wallace. The rationale of the current optical networking initiatives. *Future Generation Computer Systems*, 19(6):999 – 1008, 2003. 3rd biennial International Grid applications-driven testbed even t, Amsterdam, The Netherlands, 23-26 September 2002.

[8] Zhenhai Duan, Zhi-Li Zhang, and Yiwei Thomas Hou. Service overlay networks: Slas, qos and bandwidth provisioning. *Network Protocols, IEEE International Conference on*, 0:334, 2002.

[9] Cinegrid initiative, http://www.cinegrid.org.

[10] Bikash Sabata, Saurav Chatterjee, Michael Davis, Jaroslaw J. Sydir, and Thomas F. Lawrence. Taxomomy of qos specifications. *Object-Oriented Real-Time Dependable Systems, IEEE International Workshop on*, 0:100, 1997.

[11] S. Frolund and J. Koisten. Qml: A language for quality of service specification, 1998.

[12] Ioannis V. Papaioannou, Dimitrios T. Tsesmetzis, Ioanna G. Roussaki, and Miltiades E. Anagnostou. A qos ontology language for web-services. In *In Proc. of 20th Intl. conf. on Advanced Information Networking and Applications (AINA*, page 2006, 2006.

[13] Glen Dobson, Russell Lock, and Ian Sommerville. Qosont: a qos ontology for service-centric systems. In *31st EUROMICRO Conference on Software Engineering and Advanced Applications*, pages 80–87, 2005.

[14] Glen Dobson and Alfonso Snchez-Maci. Towards unified qos/sla ontologies. *IEEE Services Computing Workshops*, 0:169–174, 2006.

[15] Tian Gramm Naumowicz, M. Tian, A. Gramm, T. Naumowicz, H. Ritter, and J. Schiller. A concept for qos integration in web services. In *In 1st Web Services Quality Workshop (WQW2003) at WISE*, pages 149–155. IEEE Computer Society, 2003.

[16] Freddy Lecue and Nikolay Mehandjiev. Towards scalability of quality driven semantic web service composition. *Web Services, IEEE International Conference on*, 0:469–476, 2009.

[17] Jorge Cardoso, John Miller, Amit Sheth, and Jonathan Arnold. Quality of service for workflows and web service processes. *Journal of Web Semantics*, 1:281–308, 2002.

[18] Florian Rosenberg, Philipp Leitner, Anton Michlmayr, Predrag Celikovic, and Schahram Dustdar. Towards composition as a service - a quality of service driven approach. *Data Engineering, International Conference on*, 0:1733–1740, 2009.

[19] Yingqiu Li, Minghua Chen, Tao Wen, and Lei Sun. Quality driven web services composition based on an extended layered graph. *Computer Science and Software Engineering, International Conference on*, 3:153–156, 2008.

[20] Jia Yu, Michael Kirley, and Rajkumar Buyya. Multi-objective planning for workflow execution on grids. *Grid Computing, IEEE/ACM International Workshop on*, 0:10–17, 2007.

[21] Youmei Li and Zongben Xu. An ant colony optimization heuristic for solving maximum independent set problems. *Computational Intelligence and Multimedia Applications, International Conference on*, 0:206, 2003.

[22] In Alaya, Christine Solnon, and Khaled Ghdira. Ant colony optimization for multi-objective optimization problems. *Tools with Artificial Intelligence, IEEE International Conference on*, 1:450–457, 2007.

[23] Fang Qiqing, Peng Xiaoming, Liu Qinghua, and Hu Yahui. A global qos optimizing web services selection algorithm based on moaco for dynamic web service composition. *Information Technology and Applications, International Forum on*, 1:37–42, 2009.

[24] Fumiko Harada, Toshimitsu Ushio, and Yukikazu Nakamoto. Adaptive resource allocation control for fair qos management. *IEEE Transactions on Computers*, 56:344–357, 2007.

[25] Artin Avanes and Johann-Christoph Freytag. Adaptive workflow scheduling under resource allocation constraints and network dynamics. *Proc. VLDB Endow.*, 1(2):1631–1637, 2008.

[26] Gideon Juve and Ewa Deelman. Resource provisioning options for large-scale scientific workflows. In *ESCIENCE '08: Proceedings of the 2008 Fourth IEEE International Conference on eScience*, pages 608–613, Washington, DC, USA, 2008. IEEE Computer Society.

[27] Ioan Raicu, Yong Zhao, Catalin Dumitrescu, Ian Foster, and Mike Wilde. Falkon: a fast and light-weight task execution framework. In *SC '07: Proceedings of the 2007 ACM/IEEE conference on Supercomputing*, pages 1–12, New York, NY, USA, 2007. ACM.

[28] Luc Moreau, Juliana Freire, Joe Futrelle, Robert E. Mcgrath, Jim Myers, and Patrick Paulson. The open provenance model: An overview. pages 323–326, 2008.

[29] Anton Michlmayr, Florian Rosenberg, Philipp Leitner, and Schahram Dustdar. Service provenance in qos-aware web service runtimes. *Web Services, IEEE International Conference on*, 0:115–122, 2009.

[30] Shrija Rajbhandari, Arnaud Contes, Omer F. Rana, Vikas Deora, and Ian Wootten. Trust assessment using provenance in service oriented applications. *Enterprise Distributed Object Computing Workshops, International Conference on*, 0:65, 2006.

[31] Y. Demchenko, L. Gommans, C. de Laat, A. Taal, A. Wan, and O. Mulmo. Using workflow for dynamic security context management in grid-based applications. *Grid Computing, IEEE/ACM International Workshop on*, 0:72–79, 2006.

[32] Adam S. Z. Belloum, David L. Groep, Zeger W. Hendrikse, Bob L. O. Hertzberger, Vladimir Korkhov, Cees T. A. M. de Laat, and Dmitry Vasunin. Vlam-g: a grid-based virtual laboratory. *Future Gener. Comput. Syst.*, 19(2):209–217, 2003.

[33] Christoph Barz, Markus Pilz, and Andr Wichmann. Temporal routing metrics for networks with advance reservations. *Cluster Computing and the Grid, IEEE International Symposium on*, 0:710–715, 2008.

[34] Hong-Linh Truong, Robert Samborski, and Thomas Fahringer. Towards

a framework for monitoring and analyzing qos metrics of grid services. *e-Science and Grid Computing, International Conference on*, 0:65, 2006.

[35] Philippe Massonet, Yves Deville, and Cèdric Néve. From aose methodology to agent implementation. In *Proceedings of the first international joint conference on Autonomous agents and multi agent systems*, pages 27–34. ACM Press, 2002.

[36] A. Kay. Computer software. *Scientific American*, 251(3):53–59, 1984.

[37] M. Wooldridge and N. Jenings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2):115–152, 1995.

[38] Fabio Bellifemine, Agostino Poggi, and Giovanni Rimassa. JADE: a FIPA2000 compliant agent development environment. In *Proceedings of the fifth international conference on Autonomous agents*, pages 216–217. ACM Press, 2001.

[39] The SurfNet. The surfnet homepage. In *http://www.surfnet.nl/*, 2002.

[40] Jeroen van de Ham. *A semantic model for complex computer networks the Network Description Language*. PhD thesis, University of Amsterdam, 2010.

[41] The opendrac network provisioning software. http://www.opendrac.org.

[42] Doina Caragea and Tanveer Syeda-Mahmood. Semantic api matching for automatic service composition. In *WWW Alt. '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 436–437, New York, NY, USA, 2004. ACM Press.

[43] Matthias Klusch, Benedikt Fries, and Katia Sycara. Automated semantic web service discovery with owls-mx. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 915–922, New York, NY, USA, 2006. ACM.

[44] Marian Bubak, Tomasz Gubala, Michal Kapalka, Maciej Malawski, and Katarzyna Rycerz. Workflow composer and service registry for grid applications. *Future Gener. Comput. Syst.*, 21(1):79–86, 2005.

[45] Zhiming Zhao, Suresh Booms, Adam Belloum, Cees de Laat, and Bob Hertzberger. Vle-wfbus: a scientific workflow bus for multi e-science domains. In *Proceedings of the 2nd IEEE International conference on e-Science and Grid computing*, pages 11–19, Amsterdam, the Netherlands, December 4- December 6 2006. IEEE Computer Society Press.

International Workshop on Advances in Business ICT

ABICT is organized within a framework of the International Multiconference on Computer Science and Information Technology (IMCSIT), and is focuses on Advances in Business ICT approached from a multidisciplinary perspective. The ABICT 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

Ajith Abraham, Norwegian University of Science and Technology, Norway

Witold Abramowicz, Poznań University of Economics, Poland

Rainer Alt, University of Leipzig, Germany

Amelia Badica, University of Craiova, Romania

Giuseppe Berio, Universite de Bretagne Sud, France

Witold Bielecki, Kozminski University, Poland

Rimantas Butleris, Kaunas University of Technology, Lithuania

Witold Byrski, AGH-University of Science and Technology, Poland

Gerardo Canfora, University of Sannio, Italy

Longbing Cao, University of Technology, Australia

Miriam Capretz, University of Western Ontario, Canada

Dickson K. W. Chiu, Dickson Computer Systems, Hong Kong

Dimitar Christozov, American University in Bulgaria, Bulgaria

Flavio Corradini, University of Camerino, Italy

Emanuele Della Valle, Politecnico di Milano, Italy

Nirmit Desai, IBM India Research Labs, India

Petr Dostal, Brno University of Technology, Czech Republic

Marek Druzdzel, University of Pittsburgh, Biaystok Technical University, USA

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

Bogdan Franczyk, University of Leipzig, Germany

Jozef Goetz, University of La Verne, USA

Adam Grzech, Wrocław University of Technology, Poland

Ryszard Janicki, McMaster University, Canada

Stanisław Jarząbek, National University of Singapore, Singapore

Joanna Józefowska, Poznań University of Technology, Poland

Pontus Johnson, Royal Institute of Technology, Sweden

Janusz Kacprzyk, Institute of Computer Science, Polish Academy of Sciences, Poland

Pawel J. Kalczynski, California State University, USA

Maria Katharaki, University of Athens, Greece

Mieczysław Kłopotek, Institute of Computer Science, Poland

Waldemar Koczkodaj, Laurentian University, Canada

Mieczysław Kokar, Northeastern University, USA

Beata Konikowska, Institute of Computer Science, Poland

Michael L. Korwin-Pawlowski, Universite du Quebec en Outaouais, Canada

Marek Kowalkiewicz, SAP Research, Australia

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

Jie Lu, University of Technology Sydney, Australia

Abdel-Badeeh M. Salem, Ain Shams University, Egypt

Zakaria Maamar, Zayed University, United Arab Emirates

Maria Antonina Mach, Wrocław University of Economics, Poland

Lech Madeyski, Wroclaw University of Technology, Poland

Yannis Manolopoulos, Aristotle University of Thessaloniki, Greece

Florian Matthes, TU Munchen, Germany

Mieczysław Muraszkiewicz, Warsaw University of Technology, Poland

Markus Nüttgens, University of Hamburg, Germany

Andreas Oberweis, Karlsruhe Institute of Technology (KIT), Germany

Mitsunori Ogihara, University of Miami, USA

Mieczysław L. Owoc, Wrocław University of Economics, Poland

Tomasz Pełech-Pilichowski, AGH-University of Science and Technology, Poland

Vassilios Peristeras, National University of Ireland, Ireland

Lyubomyr Petryshyn, AGH-University of Science and Technology, Poland

Carlos Andre Reis Pinheiro, Dublin City University, Ireland

Andrzej Pownuk, University of Texas at El Paso, USA

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, USA

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

Jan Vanthienen, Katholieke Universiteit Leuven, Belgium

Krzysztof Węcel, Poznań University of Economics, Poland

Hans Weigand, Tilburg University, Netherlands

Mathias Weske, University of Potsdam, Germany

Stanislaw Wrycza, University of Gdansk, Poland

Slawomir Zadrozny, Polish Academy of Sciences, Poland

John Zeleznikow, Victoria University, Australia

Jerzy S. Zieliński, University of Łódź, Poland

Jozef Zurada, University of Louisville, USA

Organizing Committee

Abdel-Badeeh M. Salem, Ain Shams University, Egypt

Maria Antonina Mach (Chairman), Wrocław University of Economics, Poland

Tomasz Pełech-Pilichowski, AGH-University of Science and Technology, Poland

ISBN 978-83-60810-27-9
ISSN 1896-7094

A method for consolidating application landscapes during the post-merger-integration phase

Andreas Freitag, Florian Matthes, Christopher Schulz
Technische Universität München
Boltzmannstrasse 3, 85748 Garching bei München
Email: {Andreas.Freitag, Florian.Matthes, Christopher.Schulz}@in.tum.de

***Abstract*—Mergers and acquisitions (M&A) have become frequent events in today's economy. They are complex strategic transformation projects affecting both - business and information technology (IT). Still, empirical studies reveal high failure rates regarding the achievement of previously defined objectives. Taking into account the role and importance of IT in modern business models, the consolidation of application landscapes and technical infrastructure represents a challenging exercise performed during the post-merger-integration. Unfortunately, not many artifacts in the form of tangible concepts, models, and methods exist facilitating the endeavors of merging IT.**

After providing a broad overview on relevant literature in the area of M&A from a business and IT perspective, this article presents a method artifact for consolidating application landscapes in the course of a merger. It originates from the approach applied during a case study in the telecommunication industry where the application landscapes of two formerly independent lines of business have been merged.

I. Introduction

For almost 100 years *mergers and acquisitions* (M&A) have been used as a strategic management instrument in many enterprises [1]. In the 21st century, the appearance of corporate consolidations and reorganizations remains remarkably high, whereas M&A are not single events, but rather an integral part of modern business strategies [2]. A typical driver for M&A is the realization of increased market power through inorganic growth, resulting in economies of scale and cost reductions [3]. Penzel [4] for instance, speaks of annual cost savings between 10% and 20%. Moreover, new markets may be conquered through the enlargement of the product and service portfolio in order to realize economies of scope [4].

Although the two terms "merger" and "acquisition" in M&A are often used as synonyms, both words denote slightly different things and should not be misperceived [5]. Whereas the terms are sufficiently defined and consistently applied in the Anglo-Saxon publications (especially in the United States), German literature still lacks a commonly accepted distinction between both concepts [6]. However, looking on M&A from an *information technology* (IT) perspective, it is sufficient to consider M&A as any type of enterprises' fusion under one economic authority, independent from the legal status of the participants.

Despite their frequent occurrence, approximately 50% of all M&A succeed [7]. Gerds [1] even reports a failure rate higher than 60% which is confirmed by a multitude of empirical surveys [8], [9]. Several studies evaluate risks and common pitfalls in order to identify key success factors for M&A planning and *post-merger-integration* (PMI) projects. Our literature research resulted in the subsequent list, which is mutually agreed upon by the majority of authors ([10], [11], [12], [13], [14]), even though they apply different languages and terminology:

- Clear business vision - committed, explicitly described, and measurable
- High aspiration level, definite directions, common performance indicators
- Stakeholder management, effective communication, and corporate cultures
- Project organization (structures and processes)
- Coordinated and holistic planning of business and IT
- Consistent decisions for business and IT
- Knowledge management
- Risk management
- Realize growth and demonstrate early wins

Given that the majority of enterprises are characterized by intensive usage of IT today [15], [16] as well as regarding the items of above's list, it becomes obvious that IT also should be taken into consideration during M&A. According to the Gartner study "IT Spending and Staffing Report 2008" [17], typical investments for IT account for around 3.4% of the annual revenue. Even if for many of these enterprises' IT presently still plays an inferior role by supporting main business processes only, medium to large size companies cannot be imagined without it. Unfortunately, the significance of IT is often undervalued in the course of M&A and resulting *integration* endeavors [18], [16]. This argument is underpinned by the work of Johnston and Yetton [19], who emphasize that the IT division can be critical to merger success in M&A of large enterprises particularly during the intractable activities of the PMI phase.

In this article, the term integration refers to post-merger-integration phase of a merger or an acquisition. Although sometimes the term is differently defined when taking a closer look on current literature [20], [21], integration can be regarded as a logical consequence realizing a strategic decision in comprising the totality of changes and process steps necessary for the consolidation of two different entities. In the following, the terms integration, merge, and consolidation are used interchangeably for ease of reading. Furthermore, this

978-83-60810-27-9/09/$25.00 © 2010 IEEE

article considers an integration as a complete amalgamation of two or more entities resulting in one remaining entity.

The integration of IT includes the consolidation of two or more heterogeneously evolved application landscapes which previously supported different businesses. In the case of a complete integration, those landscapes are consolidated entirely, hence coupling solutions and green field approaches are not considered. Consequently, in the aspired future application landscape each single functionality is realized non-redundantly by one dedicated application. When it comes to specific artifacts facilitating the integration of application landscapes in the course of M&A projects, little literature exists, both in academia as well as in practice [22], [23], [19]. In this regard, an artifact refers to all innovations attempting to create utility for an organization: constructs, models, methods, and instantiations as specified by [24]. The present article proposes a method for consolidating historically independently grown application landscapes originating from a case study in the telecommunication industry in which two *lines of business* (LoB)s have been integrated. Thereby, the planning and implementation was based on the *enterprise architecture* (EA) framework *TOGAF* (The Open Group Architecture Framework) [25], which was tailored in order to fit to the specific merger context.

The remainder of this article is structured as follows: Section II provides a solid overview on existing M&A literature covering the business and IT view on the topic. In Section III, a method for integrating two different application landscapes is presented as applied in the case study. Finally, Section IV concludes by summarizing the article and outlining further fields of research.

II. Related Work

When addressing the challenge of IT integration in the course of the merger business and IT related sources have to be taken into account. First group of literature focuses on the overall M&A process and conditions from a business point of view while the second group explicitly copes with merger relevant IT topics. Complementing both views, we also examine a representative subset of *enterprise architecture management* (EAM) literature, which provides a holistic view on an enterprise with regard to concepts and ideas addressing M&A challenges like consolidating application landscapes.

Approaching M&A from a business stance, the work of Bänzer et al. [26] constitutes a comprehensive and widespread overview. By differentiating between a general planning, execution, and integration phase, it thoroughly investigates on the different forms, activities, organizational impacts, constraints, and artifacts which designate M&A projects. Nonetheless, the role and importance of IT is solely motivated by a high-level IT due diligence checklist [26]. The in German-speaking countries well known book of Jansen [6], gives a systematic introduction to the topic of M&A from a business point of view. Once again, IT is not in the scope of this work. Gerds and Schewe [1] shed light on M&A by elaborating a so-called “recipe for success” regarding beneficial post merger achievements. In providing several case studies from global positioned enterprises, the work points out main differences between top performer and M&A average. Nonetheless, the significance of IT is not elaborated on in detail. Further literature proposes specific taxonomies, calculation rules, and financial metrics to evaluate the outcomes of a merger [7]. However, IT mostly plays a minor role [27] or is even entirely omitted [28]. In summary, due to their business focus this group of literature sets the overall context of the merger but does not provide specific artifacts for the consolidation of IT.

Tackling M&A from an IT perspective, the work of Miklitz and Buxmann [22] points out four different integration strategies for application landscapes. The authors present a concrete design artifact for selecting applications expressed by a decision model which targets at the standardization of the landscape. Unfortunately, their article refrains from evaluating the model in practice. Penzel and Pietig [14] propose a so-called “Merger Guide” structuring bank mergers into time slices and dimensions. The authors highlight the importance of IT, represented through a proper dimension in the merger process and spend a dedicated chapter dealing with IT during M&A. Besides pointing out relevant system integration strategies, system transition plans, and a layer model of a bank’s system architecture, the work also considers data migration and the shutdown of obsolete systems. Nevertheless, Penzel and Pietig do not provide concrete methods or key deliverables to carry out the transformation from multiple application landscapes to one.

Considering current literature in the domain of EAM, M&A is mostly addressed as one possible field of application and in a brief manner only. Ross et al. [29] observe, that a certain maturity level of EAM is a prerequisite to manage M&A. Nevertheless, the authors do not explicitly address EAM processes or methods, but rather present several M&A case studies. Niemann [30] shortly sketches a merger situation as well as the implications on the application and infrastructure landscape during the development of planning scenarios without providing a method to deal with this type of situation. Focusing on the general management of integration projects, Winter examines a series of case studies. He identifies M&A as one major trigger for integration projects [21] and motivates the need for a situation specific integration method being applied in the course of a PMI phase. Keller [31] dwells on mergers within IT application portfolio management. He consciously creates the link between EAM and M&A by presenting the ladder of integration and the basic pattern of application consolidation. Still, no specific method is proposed aiming at consolidating application landscapes.

Due to their high practical relevance and continual increasing awareness amongst academia, EA frameworks are a valuable source when it comes to M&A. While frameworks such as Zachman [32] only classify the descriptions of an EA, The Open Group Architecture Framework (TOGAF) [25] also provides elements to establish a sustainable architecture function in an organization and proposes an *Architecture Development Method* (ADM). However, since EA frameworks are a collection of best practices covering a broad range of use cases, relevant parts of the chosen framework have to be selected and explicitly tailored to the specific needs. The same applies to the case of M&A.

In the reviewed literature, concrete M&A artifacts for merging IT are rarely addressed. Most notably, the consolidation of application landscapes is not elaborated in detail. In the majority of cases, the authors differentiate between the general strategies for IT consolidation: cherry picking, steamroller, co-existence and green field approach [19], [31], [22], [33]. Unfortunately, these depicted suggestions remain rather general and abstract. In contrast, the method proposed in subsequent Section explicitly copes with the consolidation of independently evolved application landscapes, ranging from initial clarification of the common business vision to planning of a roadmap on an application level.

III. Method to consolidate application landscapes

A. Preliminary considerations

A M&A situation between two companies whose business models rely on IT inherently entails the complete integration of the application landscapes during the PMI phase in order to realize the intended synergies. Following a theory-building approach from one or more case studies as motivated by Eisenhardt and Graebner [34], this section suggests a method artifact for application landscape consolidation. Thereby, the method is based on a case study from the telecommunication industry, hence the focus lies on theory building rather than testing the designed artifact. After a brief introduction to the case study, the resulting method is presented in the first part of this section. Subsequently, the article continues by describing each method step in detail. In doing so, every single step is accompanied by the respective part of the case study printed in italic letters.

B. A case study from the telecommunication industry

The telecommunication group comprises two lines of business (LoB)s - fixed line and mobile business. The newly-defined corporate strategic goal driving the merger of both LoBs was to increase customer satisfaction achieved by high service quality during each contact. Another major requirement was the responsive support of personalized marketing campaigns providing customers with product and service offers in a timely manner. Consequently, the need for an integrated *customer relationship management* (CRM) had been identified. Both LoBs acted in different competitive environments with individual business models, products, and processes but with partly identical customers. Furthermore, their CRM application landscapes have been developed independently to a large degree in the past.

In order to establish an integrated CRM supported by a common application landscape, the commissioned CRM project team used the presented method to fulfill subsequent core tasks:

- develop a comprehensive and corporate-wide approved vision for CRM consisting of the business target picture and the architectural blueprint
- gain transparency about the current CRM application landscapes of both LoBs in addition to their associated costs
- develop the target CRM application landscape as well as general architecture principles for the subsequent implementation initiatives
- elaborate an implementation roadmap, taking into account the existing CRM roadmaps of each individual LoB

Thereby, it was especially important to balance between strategic corporate goals and operative process and data requirements of the various sales and service divisions at the two LoBs. Both had to be considered in the target application landscape. At this point in time, the parallel business project which was in charge to work out the common target business processes was not completed. Therefore, a stable structure to coordinate the business and the IT project was required to start work immediately and integrate the different requirements relevant to the common CRM later on.

C. Method overview

The presented method is based on the EA framework TOGAF [25] and provides an approach for consolidating application landscapes driven by a merger and acquisition activities. Typically, the main IT integration work is performed during the PMI phase by means of one to more dedicated project(s). Figure 1 shows the stringent top-down approach which has been derived from the TOGAF Architecture Development Method (ADM). The ADM, as one core element of TOGAF, describes the holistic development of architectures (i.e. business, information systems, and technology architecture) following 11 distinct phases. Therefore, the method allows to interlink IT consolidation activities with general integration work conducted in other domains, e.g. business processes, resources, or staffing.

The final deliverables of the presented method artifact consist in a *business target picture*, an *architecture blueprint*, and an *implementation roadmap*. Table I provides short a definition of each term. After pointing out basic information and main context of the case study in which the method was successfully applied, the different steps of the artifact are explained in detail. At the same time, each individual step is exemplified by the experiences made and the challenges encountered during the execution of the case study.

D. Detailing the method

1) Design and establish governance model: Before working on the project's core task, i.e. the consolidation of application landscapes in the course of the PMI phase, an overall *governance model* is established. The main rationale behind is to provide a binding working environment and to form a foundation for all subsequent method steps performed by the responsible project team. As major constituents, the governance model gives information about the project organization, clear

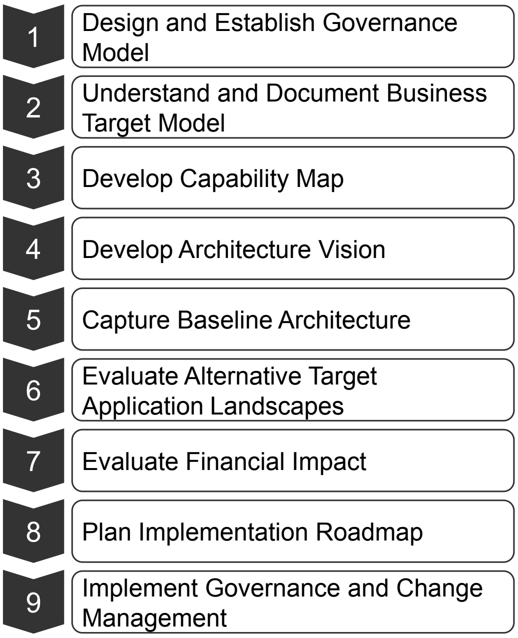

1
Design and Establish Governance Model
2
Understand and Document Business Target Model
3
Develop Capability Map
4
Develop Architecture Vision
5
Capture Baseline Architecture
6
Evaluate Alternative Target Application Landscapes
7
Evaluate Financial Impact
8
Plan Implementation Roadmap
9
Implement Governance and Change Management

Fig. 1. A method for consolidating application landscapes

TABLE I
KEY METHOD DELIVERABLES AND THEIR DEFINITION

Name	**Description**
Business target picture	An explicitly documented common corporate vision for the respective functional scope.
Architecture blueprint	A description of the target application landscape on a logical level.
Implementation roadmap	A list of individual steps of change laid out on a time line to show progression from the baseline application landscape to the target application landscape.

responsibilities, conductive rules for collaboration, reporting, as well as effective escalation paths. Besides the formal governance model, the overall project success strongly relies on the management of multidisciplinary stakeholders and the establishment of adequate communication and information activities.

The project steering committee of the telecommunication company was given the mandate to act as the required cross-LoB decision board by forming an interim architecture board for the time of the project. In total, the CRM project had to manage a group of nearly 50 stakeholders. This included representatives of various business units (e.g. CRM, marketing, sales, and product management department), IT, and controlling from both LoBs, who had to be regularly informed about the transformation progress. Fortnightly information sessions were scheduled to present and discuss relevant architecture views.

2) Understand and document business target picture: During the second step, the mandate of the application consolidation project including the functional scope, rights, and responsibilities, is specified in more detail and formally agreed upon. The functional scope is defined with the help of *architecture segments*, which according to TOGAF are *"a detailed, formal description of areas within an enterprise, used at the program or portfolio level to organize and align change activity."* [25]. The business target picture is thoroughly analyzed and documented in order to derive the strategic requirements, which will drive the development of the *target application landscape*.

The clarification of the business target picture for CRM of the company was based on a study about CRM market trends in the telecommunication sector, an analysis of the company strategy and business goals, as well as interviews with selected executive management representatives of both LoBs. It included the substantiated requirements from a business point of view, which had to be addressed by the target application landscape. The definition of the functional scope of CRM was achieved in close coordination with the corporate-wide enterprise architecture initiative. This initiative had the mandate to develop an overarching architecture model consisting of non-overlapping segments which represent distinct business domains (e.g. CRM, billing, product management, or logistics). Based on the elaborated segment structure, responsibilities considering the business and IT requirements could be non-ambiguously mapped. By this means, the CRM project team was able to develop the target application landscape for the agreed CRM segment, while routing requirements to their respective projects. For instance, requirements which resulted from the business target picture for CRM but related to different segments. Additionally, a set of architecture principles was derived from the business target picture as main guidelines to ensure a strategy-aligned execution of the implementation roadmap.

3) Develop capability map: In this step, a common language and structure for the relevant segments of the consolidation project is established among the multidisciplinary stakeholders. This is especially important to ensure a high degree of acceptance and sustainability for the solution to be developed in the course of the project. A *capability map* is used to break down the relevant architecture segment. Again, this article adheres to the definition of TOGAF, where a capability represents *"an ability that an organization, person, or system possesses"* [25]. According to TOGAF, capabilities are typically expressed in general and high-level terms, e.g. customer contract management or campaign management.

The defined CRM segment was detailed using a CRM capability map to provide a common terminology and structure among the different stakeholders from business units, IT, and controlling of both LoBs. The commonly agreed cross-LoB definitions for the CRM capabilities have been identified during a series of workshops with business and IT representatives. Afterwards, the functional view of the capabilities was complemented with major business objects, including definition of ownership and depending information flows. As a mean of communication, a graphical representation of capabilities and assigned business objects was elaborated.

4) Develop architecture vision: The *architecture vision* depicts a high-level view on the as-is and target enterprise architecture, according to the priorly elaborated business target picture. As one key element of the architecture vision, the architecture blueprint describes the target application landscape on a logical level. It is needed to analyze and compare existing application landscapes in order to support the selection of the target applications. To facilitate the comparison of the different applications, logical *architecture building blocks* (ABB)s which cluster functional and non-functional requirements, are assigned to the capabilities identified in previous step. Thereby, an ABB *"represents a (potentially re-usable) component of a business, IT, or architectural capability"* [25] as defined by TOGAF.

The CRM target application landscape was described on a logical level, according to the formulated CRM business target picture. The architecture blueprint was worked out in a series of workshops with subject matter experts and business and IT representatives of both LoBs. The various requirements from the CRM business target picture could ultimately be classified and consolidated on the basis of the capabilities. Afterwards, the planned IT support for the elaborated capabilities was described in the form of ABBs before key business objects were mapped to those ABBs in order to define data mastership and information flows derived from data usage. To ensure consistency regarding further IT initiatives in the company, the resulting architecture blueprint was also cross-checked in terms of feasibility against other segments. At this point, the business requirements which have been refined and detailed by the parallel ongoing business project were incorporated in the identified ABBs.

5) Capture baseline: In order to select the applications that optimally support the elaborated architecture blueprint, the baseline of existing applications is captured. Different applications are compared on the basis of information about their lifecycle, functional, non-functional, and financial criteria. Thereby, the developed architecture blueprint including capabilities and architecture building blocks is applied as a common reference structure to make results comparable.

To define the major applications from both LoBs that support the described architecture blueprint, the baseline of existing CRM applications was captured. To compare nearly 150 applications, a tool-based inventory with lifecycle information, functional and non-functional requirements, as well as financial properties was created. The analysis of the baseline did also include already planned changes within the application landscapes of both LoBs and existing migration roadmaps. Depending on the main functionality identified with subject matter experts from business and IT of both LoBs, each application was assigned to those ABBs it supports. This allowed for a direct comparison between the functionality offered and as-is costs of both application landscapes on ABB and capability level.

A brief illustration of assignment of applications to ABBs is depicted in Figure 2. In this example, the business segment "Customer relationship management" consist of the two distinct capabilities: "Contract management" including three ABBs, and "Campaign management" containing two ABBs. In the depicted scenario 1, ABB 1 and ABB 2 of the capability contract management are realized by the functionally enhanced application of LoB2 (APP 2). For ABB 3, a new application (APP 7) is needed to meet the common requirements. The capability campaign management is best supported by an application of LoB 1 (APP 5). Note that the business objects described above are not shown in this example.

6) Evaluate alternative target application landscapes: To select the optimal target application landscape, different alternatives are evaluated against functional and non-functional requirements attached to the respective ABBs. In each case, the required migration steps towards the target application landscape are elaborated and documented (e.g. functional extensions, data migration, or retirement).

In the case of the telecommunication company, three different target scenarios were evaluated. In each case, the required migration steps towards the target application landscape have been derived by the project team, additional subject matter experts of business departments, and IT of both LoBs.

7) Evaluate financial impact: To support the decision for one target application landscape, a corresponding business case is worked out. The calculation has to encompass estimated transformation costs, current operation and maintenance costs, as well as estimated saving potentials. Costs are structured according to LoB-specific cost center structures, but for an in-depth comparison a common reference is necessitated. The different alternative target application landscapes can be evaluated using the formerly elaborated capability map which is extended through the application of a novel controlling approach allowing to analyze costs and benefits of the transformation on capability level [35]. Finally, the step is concluded with the decision for one target landscape.

Due to the project scope, an IT cost case model was applied in the case of the telecommunication company. The estimated costs and saving potentials on application level were structured and communicated using a CRM capability map which was extended by financial information. In follow-up projects, the IT cost case model was complemented to a full business case by incorporating benefits identified on the business side.

8) Plan implementation roadmap: Finally, an implementation roadmap (cf. Figure 3) for the selected target application landscape is elaborated. The business vision is broken down into major milestones, which realize concrete business value (e.g. establishment of a common information base). The required activities concerning the different application landscapes can be grouped in workpackages according to these milestones. In addition, those applications that have to be modified (e.g. functional extensions, data migration, or retirement) can be assigned to each of these workpackages.

The implementation roadmap represented a step-by-step migration plan for the preferred scenario. It points out the sequence of projects to be carried out in order to build

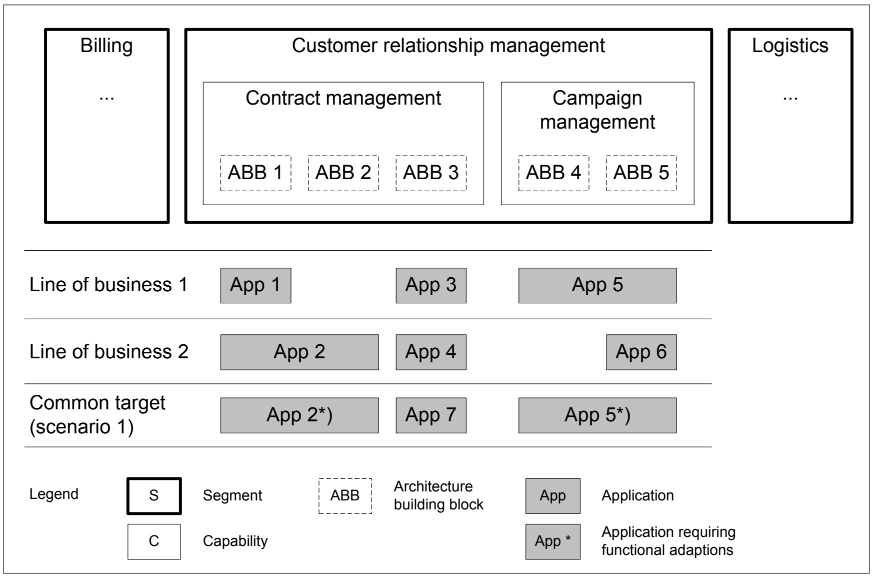


Fig. 2. Exemplary assignment of different LoBs' applications to the elaborated architecture building blocks

one common CRM application landscape. Additionally, the developed roadmap considered all formerly existing projects of each LoB and highlighted resulting dependencies.

9) Implementation governance and change management: Lastly, an adequate *implementation governance* has to be established in order to guide the following implementation projects. According to TOGAF [25], implementation governance provides an architectural supervision of the implementation. Therefore, a common set of recommendations and guidelines is formulated. Regular checkpoints are established along the implementation process to guarantee conformance with the defined target architecture and ensure the realization of the estimated business value. Furthermore, a proper change management establishes procedures to identify needs and manage changes in order to adjust the implementation roadmap if necessary. Implementation governance and change management have to be closely integrated into general integration activities.

The set of architecture principles defined in step 2 ensured a strategy-aligned execution of the developed implementation roadmap and change management was organized. By reasons of the continuous character of these two activities, the project organization was formally closed and the responsibility was handed over to the line organization.

E. Conclusion

This section presented a method for consolidating application landscapes by following a theory-building research approach. In the presented case study, the defined core tasks have been achieved in time and budget. Due to the successful accomplishment, the method has been debriefed as a reference method for the respective telecommunication company. The main benefits of the method perceived by the project sponsor and the participating stakeholders were

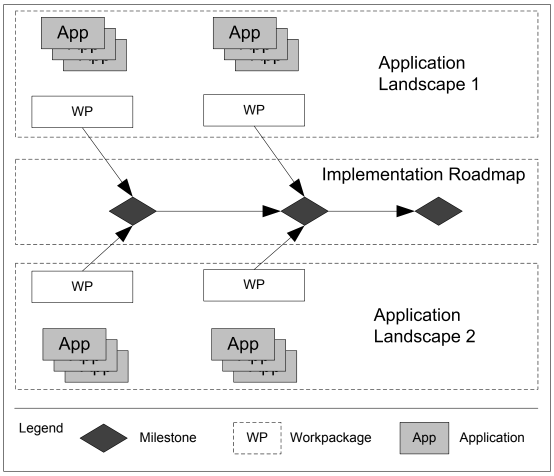


Fig. 3. Exemplary implementation roadmap

- the consistent planning, from corporate strategy for CRM to IT implementation,
- the stringent methodology, transparent and traceable for business and IT,
- the establishment of a common terminology and a common understanding for CRM, and
- the strong involvement of key stakeholders.

These benefits generated in this PMI project tie in with the general key success factors for M&A presented in Section I.

IV. Outlook & Discussion

M&A can be seen as complex and intricate company-wide transformation projects attempting to integrate two formerly disjunctive business entities. Unfortunately, they are often leading to disillusioning economical results or complete failure. Due to the fact, that IT is an integral part of the business model in many industries, its importance during an M&A should not be underestimated. However, the selected literature analyzed in this article does not provide relevant artifacts, i.e. concepts, models, and methods helping to meet the challenges of an IT integration. In particular with regards to the complete consolidation of different application landscapes in the course of a merger, no comprehensive approach exists to the knowledge of the authors.

By examining a real-world case study in which two differently administered lines of business had to be integrated from a business and IT perspective, this article proposes a method artifact aiming at consolidating formerly independently evolved application landscapes. As one core concept of the presented method, capabilities proved to be valuable in serving as a stable foundation between business and IT when assessing two landscapes from a functional, non-functional, as well as financial point of view. Furthermore, the document also showed how an adapted TOGAF Architecture Development Method (ADM) can be successfully applied in the context of M&A.

The method requires further evaluation and justification in order to prove its applicability and relevance for the merger context. While the artifact has been established as a standard method in the respective telecommunication company, it is currently re-applied in the course of an application landscape consolidation project of two German software companies. In this vein, the artifact could be further on extended by a distinctive role and organizational model depicting the different actors and their respective points of action during the method. Moreover, specific context factors of M&A as described by business resources (e.g. [1], [15]) should be taken into account when refining the method. In a subsequent step, concrete architecture viewpoints validated by means of complementary case studies should be designed and evaluated.

In all, this article presents an initial foundation when studying IT integration during M&A situations. The depicted method is one of several artifacts which is useful in supporting the consolidation of application landscapes during the PMI phase.

References

[1] J. Gerds and G. Schewe, *Post Merger Integration: Unternehmenserfolg durch Integration Excellence*, 3rd ed. Berlin, Heidelberg, Germany: Springer-Verlag, 2009.

[2] R. Sperry and A. Jetter, *Mergers and Acquisitions: Team Performance*. Portland, USA: IEEE, 2007.

[3] C. B. Bark, *Integrationscontrolling bei Unternehmensakquisitionen: Ein Ansatz zur Einbindung der Post-Merger-Integration in die Planung, Steuerung und Kontrolle von Unternehmensakquisitionen*, 1st ed. Frankfurt am Main, Germany: Peter Lang, 2002.

[4] H.-G. Penzel, "Post Merger Management in Banken – und die Konsequenzen für das IT-Management," *Wirtschaftsinformatik*, vol. 41, no. 2, pp. 105–115, 1999.

[5] D. DePamphilis, *Mergers, Acquisitions, and Other Restructuring Activities, Fifth Edition: An Integrated Approach to Process, Tools, Cases, and Solutions (Academic Press Advanced Finance Series)*, 5th ed. Burlington, MA, USA: Academic Press, 2009.

[6] S. A. Jansen, *Mergers & Acquisitions: Unternehmensakquisitionen und -kooperationen. Eine strategische, organisatorische und kapitalmarkttheoretische Einführung*, 5th ed. Wiesbaden, Germany: Gabler, 2008.

[7] M. Koetter, "Evaluating the German bank merger wave," H. Herrmann, T. Liebig, and K.-H. Tödter, Eds., vol. 12. Frankfurt a. M., Germany: Deutsche Bundesbank, 2005, p. 44.

[8] R. A. Chang, G. A. Curtis, and J. Jenk, *Keys to the Kingdom: How an Integrated IT Capability Can Increase Your Odds of M&A Success*, New York, USA, 2002.

[9] A. Watkins and S. Copley, "Operational and IT Due Diligence," 2004. [Online]. Available: http://www.pwchk.com/home/eng/oprisk.html\#F

[10] M. J. Epstein, "The Drivers of Success in Post-Merger Integration," *Organizational Dynamics*, vol. 33, no. 2, pp. 174–189, 2004. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0090261604000142

[11] B. Görtz, *Due Dilligence als Schlüssel zum Erfolg von Mergers & Acquisitions*. Wiesbaden, Germany: Wirtz, Bernd W., 2006, pp. 519–532.

[12] M. M. Habeck, F. Kröger, and M. Träm, *Wi(e)der das Fusionsfieber. Die sieben Schlüsselfaktoren erfolgreicher Fusionen.*, 2nd ed. Wiesbaden, Germany: Gabler, 2002.

[13] G. Picot, *Handbuch Mergers & Acquisitions*, 4th ed. Stuttgart, Germany: Schäffer-Poeschel, 2008.

[14] H.-G. Penzel and C. Pietig, *MergerGuide*. Wiesbaden, Germany: Gabler Verlag, 2000.

[15] G. Kromer and W. Stucky, "Die Integration von Informationsverarbeitungsressourcen im Rahmen von Mergers & Acquisitions," *WIRTSCHAFTSINFORMATIK*, vol. 44, no. 6, pp. 523–533, 2002.

[16] M. H. Larsen, "ICT Integration in an M&A Process," in *Proceedings of the Pacific Asia Conference of Information Systems (PACIS)*. Copenhagen Business School, 2005, pp. 1146–1159.

[17] M. Smith, B. Gomolski, J. P. Roberts, and R. D. Souza, "IT Spending and Staffing Report," 2008. [Online]. Available: http://www.gartner.com/DisplayDocument?id=607608

[18] D. Brown, "Don't overlook IT in the merger," 2001. [Online]. Available: http://www.computerweekly.com/Articles/2001/05/31/180596/dont-overlook-it-in-the-merger.htm

[19] K. Johnston and P. Yetton, "Integrating information technology divisions in a bank merger Fit, compatibility and models of change," *The Journal of Strategic Information Systems*, vol. 5, no. 3, pp. 189–211, September 1996.

[20] T. Schäfer, *Stakeholderorientiertes Integrationsmanagement bei Fusionen und Akquisitionen*, 1st ed., A. Picot, R. Reichenwald, E. Franck, and K. Möslein, Eds. Wiesbaden, Germany: Gabler, 2008.

[21] R. Winter, *Management von Integrationsprojekten: Konzeptionelle Grundlagen und Fallstudien aus fachlicher und IT-Sicht (Business Engineering) (German Edition)*, 1st ed., H. Österle, R. Winter, and W. Brenner, Eds. St. Gallen, Switzerland: Springer, 2009.

[22] T. Miklitz and P. Buxmann, "IT standardization and integration in mergers and acquisitions: a decision model for the selection of application systems," in *Proceedings of the 15th European Conference on Information Systems (ECIS)*, St. Gallen, Switzerland, 2007, pp. 1041–1051.

[23] P. Wirz and M. Lusti, "Information technology strategies in mergers and acquisitions: an empirical survey," in *Proceedings of the winter international synposium on Information and communication technologies (WISICT)*, vol. 58. Cancun, Mexico: Trinity College Dublin, 2004, pp. 1–6.

[24] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[25] The Open Group, "TOGAF "Enterprise Edition" Version 9," San Diego, USA, 2009. [Online]. Available: http://www.opengroup.org/architecture/togaf9-doc/arch/http://www.togaf.org

[26] B. Bänzer, E. Bartels, H. Bergmann, C. Brugger, and D. Classen, *Handbuch Mergers & Acquisitions: Planung - Durchführung - Integration*, 4th ed., G. Picot, Ed. Stuttgart, Germany: Schäffer-Poeschel, 2008.

[27] P. Morosini and U. Steger, *Managing Complex Mergers (Financial Times Series)*, 3rd ed. Harlow, England: Financial Times/ Prentice Hall, 2003.

[28] C. Kummer and U. Steger, "Why merger and acquisition (M&A) waves reoccur: the vicious circle from pressure to failure," *Strategic Management Review*, vol. 2, no. 1, pp. 44–63, 2008.

[29] J. W. Ross, P. Weill, and D. C. Roberts, *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution*, 1st ed. Boston, MA, USA: Mcgraw-Hill Professional, 2006.

[30] K. D. Niemann, *From Enterprise Architecture to IT Governance: Elements of Effective IT Management*, 1st ed. Wiesbaden, Germany: Vieweg Verlag, 2006.

[31] W. Keller, *IT-Unternehmensarchitektur. Von der Geschäftsstrategie zur optimalen IT-Unterstützung*, 1st ed. Heidelberg, Germany: Dpunkt Verlag, 2007.

[32] J. A. Zachman, "A framework for information systems architecture," *IBM System Journal*, vol. 26, pp. 276–292, 1987.

[33] C. E. Rentrop, *Informationsmanagement in der Post-Merger Integration*, 1st ed. Berlin, Germany: Erich Schmidt Verlag, 2004.

[34] K. M. Eisenhardt and M. E. Graebner, "Theory Building From Cases: Opportunities And Challenges," *Academy of Management Journal*, vol. 50, no. 1, p. 25âĂŞ32, 2007.

[35] A. Freitag, *A Controlling Model for the Enterprise Architecture and SOA: Increased Cost Transparency for Modular IT Architectures*. Vdm Verlag Dr. Müller, 2008.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Hybridization of Temporal Knowledge for Economic Environment Analysis

Maria Antonina Mach
University of Economics
Department of Knowledge and Information Management
53-345 Wroclaw, Poland
Email: maria.mach@ue.wroc.pl

***Abstract*—the paper is devoted to the concept of hybridization of temporal knowledge in an intelligent reasoning system. Hybridization is a special kind of integration, where heterogeneous knowledge is transformed in order to obtain a uniform one, but information on temporal characteristics and on core features of knowledge being transformed is preserved, and may be also used for reasoning. It may be said, that integration serves as means of hybridization, but only if at least two conditions are fulfilled: if integration is source-oriented, and the knowledge sources are kept autonomous.**

In the paper we present in detail the concept of integration, the conditions that are to be fulfilled if the integration is to serve as the means of temporal hybridization. We show an application area for a hybridized knowledge, namely the analysis of economic environment of an enterprise. We present an example of a hybridization procedure.

Keywords: integration, hybridization, temporal knowledge, temporal intelligent system.

I. Introduction

THE increasing complexity of an environment, in which modern enterprises operate, entails the more and more complex formal descriptions of this environment. This in turn leads very often to a problem of heterogeneous representations. Economic environment is of heterogeneous nature, so is its formal representation (based very often on different sources) and in order to perform a coherent reasoning by an intelligent system, all fragments of the representation must be unified.

In the paper we present the problem of using knowledge coming from different sources to perform an analysis of enterprise's environment. This environment changes in time, therefore we assume, that the analysis is performed by a temporal intelligent system that is a system, which explicitly and directly performs temporal reasoning. Such a system contains not only a fact base, a rule base and an inference engine – as any intelligent system – but also contains explicit time references both in representation and reasoning layers.

The reminder of the paper is organized as follows. In Section one we show how integration may be perceived as a mechanism of temporal knowledge hybridization. Section 2 is devoted to basic approaches to data and knowledge integration – they are very shortly presented and discussed in context of hybridization. In section 3 we present in detail the idea of temporal knowledge hybridization. In the same section we present an example of the integration process leading to a hybridized knowledge obtained from two temporal knowledge bases. The paper ends with a summary section. Some additional information, needed to make the example in Section 3 more readable, is presented in Appendix A and Appendix B.

II. Integration as Mechanism of Knowledge Hybridization

There can be several sources of knowledge about enterprise's environment: databases, data warehouses, Business Intelligence systems, or other information systems used by an enterprise, as well as text documents.

For an intelligent system to reason correctly, it is necessary that it reasons not only about economic environment as a whole, but also about its fragments – that is, about particular fragments of knowledge encoded in the representation layer. Therefore the integration process must preserve the original knowledge fragments. The process of transforming heterogeneous temporal knowledge, leading to a uniform representation, but at the same time preserving information on temporal characteristics and on core features of knowledge being transformed, is called a temporal hybridization. It can be therefore said that the integration process in a temporal intelligent system is a mechanism of knowledge hybridization, where knowledge concerns enterprise's environment.

III. Basic Approaches to Data and Knowledge Integration

Lawrence and Baker [LABA01] perceive integration as a process of merging notions and knowledge from single sources, resulting in a coherent view of the whole knowledge contained in these sources. It is a point of view of knowledge-based intelligent systems. It stresses a specific role of knowledge integration in management supporting intelligent systems. Generally speaking, by a heterogeneous knowledge base we mean a base of a diversified form, structure and origin [OWOC01].

The question of knowledge integration can be viewed from different perspectives and in different contexts. The problem is discussed in detail e.g. in [MAOW04], here we present a short summary in Table 1.

TABLE 1.
TYPES OF INTEGRATION. SOURCE: OWN ELABORATION.

CRITERION	TYPES OF INTEGRATION
Integrating system operations	virtual materialized
Direction of integration process	Source-oriented Client-oriented
Changes in sources being integrated	Domain integration Semantic integration
Aim of integration	Procedural integration Declarative integration
Objects being integrated	Mono-object integration Poli-object integration
Context of integration	Functional database text agent, other

In case of the task of temporal analysis of enterprise's environment, it seems that one should speak about the following types of integration:

a) virtual – because temporal representations of environment's elements are integrated before reasoning is proceeded, therefore there is a mediating system between knowledge sources and an user;
b) source-oriented – because the tasks of the systems are left unchanged, while changes appear in the sources, thus reflecting changes in the environment, and the system has to respect these changes;
c) semantic – while integrating knowledge before reasoning, a temporal intelligent system has to solve conflicts arising from e.g. different granulations of changes of elements in enterprise's environment;
d) declarative – the main aim of the integration process is to obtain an uniform representation for further reasoning, the tasks of the system remain unchanged during the integration process, while the system is not aimed at delivering the *at hoc* information;
e) integration that keeps the sources autonomous but at the same time integrates them (hybridization);
f) mono-object – the sources being integrated before the reasoning process starts are of the same type: temporal knowledge bases.

The most important question in the process of hybridization of temporal knowledge is to respect changes in knowledge sources and to keep the sources autonomous. Only in such case information on temporal characteristics and on crucial features of knowledge being integrated is preserved. Therefore if the integration is to be used as a mechanism of hybridization, should be, first of all, source-oriented (b), and should keep the sources autonomous (e) – these are the two basic conditions of a temporal hybridization.

IV. TEMPORAL HYBRIDIZATION PROCESS

A temporal intelligent system, aimed at analyzing changes in enterprise's environment, uses many different kinds of knowledge, because the environment is heterogeneous, complex and dynamic. Elements of enterprise's environment may be of qualitative, quantitative or mixed nature. To represent them in a temporal intelligent system, one has to use also qualitative, quantitative and mixed temporal formalisms. Moreover, the system should be able to reason both about each group of elements (a so-called detailed view), and about all the elements treated as a whole (a so-called general view). To obtain a coherent view of all the elements, the hybridization of temporal knowledge about particular elements of the environment has to be done.

From the economic point of view, the heterogeneity of knowledge in the system may be perceived as:

a) heterogeneity of sources: knowledge can be gained from different sources – see introduction;
b) heterogeneity of features: because elements of the environment can be viewed as different features to be represented in a different way.

From the computer science point of view, in a temporal intelligent system we may find the following types of heterogeneity:

a) logical heterogeneity – because we assume, that elements of the environment are represented by means of logical formalisms;
b) temporal heterogeneity – because the above formalisms are temporal logics, as the features represented have an explicit temporal aspect to be captured;
c) reasoning heterogeneity – because for each group of environment's elements a different inference mechanism is needed. Each group of elements is represented in a separate knowledge base, each of which has a separate inference mechanism linked with it. Moreover, taking into account different characteristics of environment's elements, the inference mechanisms do not need to use the same inference strategy.

In our opinion there is no need to develop a new particular technique for integration of representations. The integration itself serves only as a mechanism of hybridization of knowledge about enterprise's environment. Any integration tool or technique may be used, if only it fulfills the two main conditions, pointed out in Section 2 (namely the condition <b> and <e>). Of course, the more conditions (formulated in section 2) fulfilled by the integration tool/technique, the more useful it is for a temporal intelligent system.

Consider a simple example of an integration procedure, leading to hybridization of temporal knowledge. Assume there is an enterprise that wishes to start a TV channel. The enterprise has two knowledge bases:

a) a base of legal knowledge concerning licenses – as a license is required to start a TV channel;
b) a base of knowledge concerning the influence of dollar's exchange rate on capital barriers to entry to a mass-media market (knowledge gained from an expert).

The knowledge bases are of the following form (simplified to make the example clear):

a) legal knowledge base – formalized in the LTR language[1]

```
if TT1: license(issuing authority, enterprise)
     TT2: is_issued(TT1)
     Occurs(TT2)
 Then occurs(valid(TT1), instant(TT2))

 If  TT2: valid(TT1)
      Occurs(TT2)
 Then Holds_on(valid(TT1), period(TT3))
 Period(TT3) Equals(5y, 50y)
```

b) knowledge base on capital barriers to entry – formalized in the Prolog+CG language [KABB00], designed to formalize conceptual graphs[2]. As the knowledge base is temporal, the original conceptual graphs notation has been augmented with fuzzy temporal references, taken from [KALC05].

```
  [CRB]  -  actn  ->  [change]  -  manr  ->
[proportionally] :-
  [sit  =  [fluctuations]  -  attr  ->  [fre-
quent],
  -  ptnt  ->  [DER]  ]  <-  tref  -  [TR  =
[[GCGty: #ctx] - TAgo -> [month: {}@3]]
<- TOvr].
  (if  during  the  past  3  months  frequent
fluctuations  of  DER  has  been  observed,
then CRB changes proportionally)

  [CRB] - actn -> [lower] :-
    [sit = [decrease] - ptnt -> [DER]] <-
tref  -  [TR  =  [Year:  2004]  -  Tsnc  ->
[GCGty: #end]].
  (if since the end of 2004 the decrease
of  DER  has  been  observed,  then  CRB  low-
ers)
```

where: CRB – capital barriers to entry, DER – dollar exchange rate.

In this example we observe the problem of the logical heterogeneity (two different representation formalisms), the heterogeneity of features being represented (two different kinds of barriers to entry – legal and capital ones), and the heterogeneity of temporal granulation – for the legal knowledge, the granulation may be established for one year, while for the knowledge about DER the more accurate granulation is one day (this is so because of a different pace of changes in the case of law and in the case of DER).

A sample procedure of integration of the above knowledge bases may be as follows:

a) transformation of granulations to obtain an uniform one;
b) establishing a reference time point;
c) choosing a final representation formalism;
d) transformation of time ontologies to obtain an uniform one;
e) conversion onto the final formalism and merging of the two knowledge bases.

Ad a)

It seems natural that a common granulation for both knowledge bases should be one day. It is so because it is easier to present years in terms of days than inversely. Under such circumstances the knowledge base about the DER remains unchanged, while the second rule in the legal knowledge base is transformed according to the new granulation:

```
 If  TT2: valid(TT1)
      Occurs(TT2)
 Then Holds_on(valid(TT1), period(TT3))
 Period(TT3) Equals(1825d, 18250d)
```

We assumed, for simplicity, that each year equals 365 days.

Ad b)

A reference time point is needed for further ontology standardization. We propose that the reference point be a day on which integration is done, e.g. r_0= 07/09/2006.

Ad c)

The choice of the final temporal representation formalism depends on time ontology and on the integration technique. In our example we deal mainly with time intervals (validity period of a license, a period in which changes of DER are observed). Therefore one of the possible common formalisms may be Allen's interval algebra [ALLE81], [ALLE83]. There are several reasons that justify this choice:

- the LTR theory (used to formalize the legal knowledge base) is partly based on Allen's algebra,
- Allen's algebra in turn derives from the first order predicate calculus,
- the conceptual graphs (used to formalize the second knowledge base) are easy to convert on 1st order predicate calculus – see Appendix B.

Ad d)

Ontology standardization means establishing, what temporal objects are in both knowledge bases, and converting them onto the final temporal formalism. In the example there are the following temporal objects:

- event: license issuing, proportional change of a barrier, lowering of a barrier, lowering of DER, fluctuations of DER,
- fact: validity of a license;
- agent: issuing authority,
- constant: interval [1825d, 18250d] – to be denoted as t_2,
- variable: enterprise, license,
- interval: 3 months before the reference point r_0 – to be denoted as t_3, an interval from the end of 2004 to the reference point r_0 – to be denoted as t_5.

Ad e)

[1] The basic concepts of the LTR theory are explained in Appendix A.

[2] A short explanation on conceptual graphs is given in Appendix B.

To convert both knowledge bases onto Allen's interval algebra, we introduce two predicates: HAS(what, who) and DURATION (interval, length). After conversion all the rules (both bases merged together), using the time ontology established in step d) are of the following form:

OCCURS(ACAUSE(issuing_authority, license_issuing), t_1) ⇒ OCCURS(license_issuing, t_1)

OCCURS(license_issuing, t_1) ⇒ HOLDS(license_validity, t_2) ∧ HAS(license, enterprise) ∧ MEETS(t_1, t_2) ∨ BEFORE(t_1, t_2) ∧ DURATION(t_2, [1825, 18250])

OCCURRING(frequent_fluctuations_DER, t_3) ⇒ OCCURS(change_CRB, t_4) ∧ DURATION(t_3, 90) ∧ BEFORE(t_3, t_4) ∨ MEETS(t_3, t_4)

OCCURS(decrease_DER, t_5) ∧ ECAUSE(decrease_DER, t_5, lower_CRB, t_6) ⇒ OCCURS(lower_CRB, t_6) ∧ BEFORE(t_5, t_6) ∨ MEETS(t_5, t_6) ∨ OVERLAPS(t_5, t_6) ∧ DURATION(t_5, [(2004 12 31), r_0])

The main advantages of the above sample integration procedure are standardization of formalisms and preserving temporal information. The main disadvantage, on the other hand, is that in the final knowledge base there is no possibility to encode fuzzy temporal references. This means that we have lost a part of information. This disadvantage is due partly to the choice of Allen's algebra, partly to the problems with formalizing fuzzy intervals, and partly to simplification of the procedure, to make the example clear.

What needs to be stressed is that after the above integration procedure both original knowledge bases remain unchanged, the third knowledge base is added to them, and therefore the two basic conditions of temporal hybridization are fulfilled.

V. Conclusions

In practice the most popular and the most widely used are the tools and techniques of semantic integration. If it is assumed that the main aim of integration is to create a coherent representation of information from different sources, and to enable reasoning based on heterogeneous sources, it becomes obvious that semantic, conceptual integration is indispensable.

There are many advantages of using hybridized knowledge in enterprise's intelligent systems. Let us point out the following ones (see also [IWFA95], [OWOC03]):

- "openness" of knowledge – caused by its variety – allows for a more accurate reflection of a domain being analyzed;
- a model of hybrid knowledge makes analysis and prediction of domain's behavior easier. This has been proven in the context of hybrid systems [IWFA95], but is valid also in context of hybrid knowledge;
- knowledge built upon independent sources is more unquestionable, reliable and trustworthy than knowledge built upon a single source.

The other advantages of using hybridized knowledge and maintaining heterogeneous knowledge bases in the reasoning system are as follows:

- each element of the economic environment is represented in a separate knowledge base (knowledge "openness") which leads to a precise representation of elements having different characteristics;
- easy analysis of economic elements – when elements are represented separately, the system can easily capture their features and take them into account during the reasoning process;
- reliability of knowledge about economic environment – using several knowledge sources before integration, if these sources concern the same economic element, allows to improve the level of knowledge reliability. Of course it is assumed that in case of any contradictions there is a method of eliminating them, but this question is beyond the scope of this paper.

We do not address in the paper problems concerning maintaining hybridized knowledge bases, such as problems of knowledge coherence or knowledge validation. The detailed discussion on these problems can be found e.g. in [OWOC01] or [MAOW04] – the procedures used to ensure knowledge coherence, and the procedures of knowledge validation and verification are similar to those used in the case of typical, non-hybridized knowledge bases.

Appendix A. Basic Concepts of the LTR Theory

The detailed description of the theory can be found in [VIYO98]. Table 2 contains a summarization of the most important LTR features.

Table 2.
Basic features of the LTR theory. Source: [VIYO98].

Legal Temporal Representation (LTR)	
Time ontology	Elements: points, intervals, durations, clock-calendar constants Relations: <, *begin, end, next, previous, ImmediateBefore, ImmediateAfter*
Time theory	IP theory axioms + discreteness axioms + „immediateness" axioms
Temporal constraints	Qualitative point-point, metric over points, qualitative interval-interval, qualitative point-interval, unary over durations
Temporal qualification	Temporal tokens
Incidence theory	Predicates: holds, occurs, holds_at, holds_on Axioms: *holds* and *holds_on* homogeneity

Some features in the above table need a few words of commentary.

a) IP theory – point-interval time theory by Vila and Schwalb [VISC96];

b) Discreteness axioms – the authors of the LTR theory assume time to be discrete, although not always discrete time is sufficient. In our example however we can have discrete time, as law does not change in a continuous way (or at least we may assume it doesn't);

c) Predicates – based on Allen's time theory. The Holds predicate is used both for points and intervals. The holds_on predicate concerns holding of a feature over an interval, and holds_at – holding of a feature in a time point. It may be disputable, whether there really is a need for multiplying the variants of the Holds predicate.

The most important feature of the LTR theory, that needs more explanation here, concerns the so-called temporal tokens. They are used to link propositions with their times, which means a temporal qualification of propositions. Tokens may also be predicates' arguments. The method comes from a temporal arguments method (see e.g. [HAUG87]), where time is introduced as one or more additional arguments, for example:

Valid(act, t1, t2)

Tokens, instead, link propositions with the time of occurrence, e.g.:

Valid(act, tt1)

Which allows for the following interpretations: begin(tt1) = 01/01/1990 – the starting point for token tt1; or period(tt1) = [01/01/1990; 31/12/1990] – an interval over which a token tt1 is valid. It can be therefore said that a token represents a special temporal case of a temporal proposition.

A law article is formalised in the LTR theory as a rule or rules that express relations between the occurrences of events (under certain conditions) and their effects, being the holding of certain features. The LTR language is a rule one, and does not need any assumption on reasoning method.

Appendix B. Some Remarks on the Conceptual Graphs

The conceptual graphs notation has been originally introduced by Sowa [SOWA00]. A conceptual graph is a bipartite, directed graph with two kinds of nodes: notions and relations. It may be encoded graphically, with rectangles representing notions and ovals representing relations; or as text, where notions are in square brackets and relations are in parenthesis. The formalism derives from semantic networks. We have chosen this notation because conceptual graphs have some advantages, such as:

- they are legible, easy to understand, and at the same time it is a strictly formal notation,
- they are useful for representing AI problems, e.g. formalizing natural language sentences,
- they are easily expressed in KIF and first order logic, which allows using CG rules in many types of reasoning systems.

The version based on conceptual graphs needed slight modifications in relation to widely accepted CG notation, because of Prolog+CG language's particular requirements. They are as follows:

a) we could not use the AGNT relation, because, according to formal requirements, only a living being may be an agent. Therefore, instead of writing e.g. [remain] – AGNT -> [OP], we used to write: [OP] – ACTN -> [REMAIN], where ACTN stands for action;

b) the premises are time-stamped (relation Tref, graph name: TR), while conclusions are not – there are only general references such as "past", "future", linked with the CRB variable by a PTIM relation. This is so, because the validity period of conclusions is not always the same as the validity period of premises. In our opinion, there is no justification for using the same time reference in premises and conclusions;

c) according to the Prolog+CG notation, relations' names are given without parenthesis;

d) according to the Prolog+CG notation, if the graph has several branches, they are separated by commas.

As we stated earlier, the original Sowa's notation was augmented with fuzzy temporal references.

References

[ALLE81] Allen J. F., *An Interval-Based Representation of Temporal Knowledge*. Proc. IJCAI-81 – 7th International Joint Conference on Artificial Intelligence, Morgan Kaufmann 1981.

[ALLE83] Allen J. F., *Maintaining Knowledge about Temporal Intervals*. „Communications of the ACM", Vol. 26 No. 11, November 1983.

[HAUG87] Haugh B. A., *Non-standard semantics for the method of temporal arguments*. Proc. IJCAI-87: 10th International Joint Conference on Artificial Intelligence. Morgan Kaufmann Publishers, 1987, pp. 449-455.

[IWFA95] Iwasaki Y., Farquhar A., Saraswat V., Bobrow D., Gupta V., *Modeling Time in Hybrid Systems: How Fast is „Instantaneous"?* Proc. IJCAI-95: 14th International Joint Conference on Artificial Intelligence, Montreal, Canada, Vol. 2, pp. 1773-1780, Morgan Kaufmann.

[KABB00] Kabbaj A., *Prolog+CG User's Manual. Version 2.0., 01-12-2000.* www.insea.ac.ma/CG-Tools/PROLOG+CG.htm downloaded July 30, 2005.

[KALC05] Kalczynski P., *Temporal Semantic Networks for Business News*, University of Toledo Research Paper 2005.

[LABA01] Lawrence R., Barker K. *Integrating data sources using a standarized global dictionary*. W: Abramowicz W., Zurada J. (Eds.), *Knowledge discovery for business information systems*. Kluwer Academic Publishers, USA, 2001. pp. 153-172.

[MAOW04] Mach M., Owoc M. L., *Integracja wiedzy ze źródeł heterogenicznych*. Wydawnictwo BEL, Warszawa 2004 [*Integrating knowledge from heterogeneous sources*].

[OWOC01] Owoc M. L., *Podejścia do weryfikacji heterogenicznych baz wiedzy*. In: Baborski A. (red.), *Pozyskiwanie wiedzy z baz danych.* Prace naukowe AE Wrocław nr 891, Wrocław 2001. [*Approaches to heterogeneous knowledge bases verification*].

[OWOC03] Owoc M. L., *Knowledgebases: A Management Context and Development Determinants*. Proc. IS-2003: Informing Science Conference, Pori, Finland, June 2003, pp. 1193-1199.

[SOWA00] Sowa J. F., *Knowledge representation*. Brooks/Cole, Pacific Grove, CA, 2000.

[VISC96] Vila L., Schwalb E., *A Theory of Time and Temporal Incidence based on Instants and Periods*. TIME-96: Third International Workshop on Temporal Representation and Reasoning. Key West, Florida, USA, May 19-20, 1996.

[VIYO98] Vila L., Yoshino H., *Time in automated legal reasoning*, w: Martino A., Nissan E. (eds.), “Information and Communications Technology Law”, Special issue on formal models of legal time. Vol. 7 No. 3, 1998.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Independent Operator of Measurements as a Virtual Enterprise on the Energy Market

Bożena Matusiak
University of Lodz,
Department of Computer Science
Matejki 22/26 Str.
90-237 Łodź
bmatusiak@wzmail.uni.lodz.pl

***Abstract*—The independent Operator of Measurements (IOM) is a new energy market actor that will implement a remote access to the metering data and stores, as well as aggregating and delivering them in real time to all market participants as an independent service-placed "above the market" enterprise.**

In that sense IOM, with an adequate infrastructure of ICT has become a virtual company with real effect - where, thanks to its distributed and virtual activities on the energy market - achieves it, synergizes the whole market and activates the necessary and accurate real-time decisions of other market participants.

IOM is a virtual unit (not necessarily constrained by any single location) from the point of view of the access to the Market.

The Energy market (EM) in Poland, where IOM will be set up, has become closer to the full liberalization, energetic efficiency, enhanced competitiveness and contribute to the smart grids potential.

This article presents the issues of ICT and the creation of new business models for the above-described virtual enterprise, which manages measurement data at the EM in Poland.

I. Introduction

A FUTURE model of EM assumes a development towards to the smart grid together with: an implemented AMR-AMM infrastructure (fig.1), intelligent bi-directional communication power meters for metering load power, many sources of distributed generation using among other renewable energy sources and their settlements with market aggregators on an intra day (real-time) market. The current market design in Poland is presented in some detail in: [7], [12], [13].

An essential element of this market will be the virtual access to the measurement data, collected and aggregated from the bi-directional smart multimeters measuring the media (like electricity, gas, heat, heating), aggregate of any scale, which share any eligible market participant as a service.

Real Smart Utility – a model for the future utilities of any type on the EM — -is an actor trading and marketing on the energy market, which, thanks to virtual technologies manages the flow of information in real time. It can precisely manage and balance its needs as well as the manufacture and sale of energy (prosumers, system manufacturers, small manufacturers or EM traders).

There are some barriers for the development of such market (technical, financial, legal, information technology) that need to be resolved, and that are associated with the following problems:

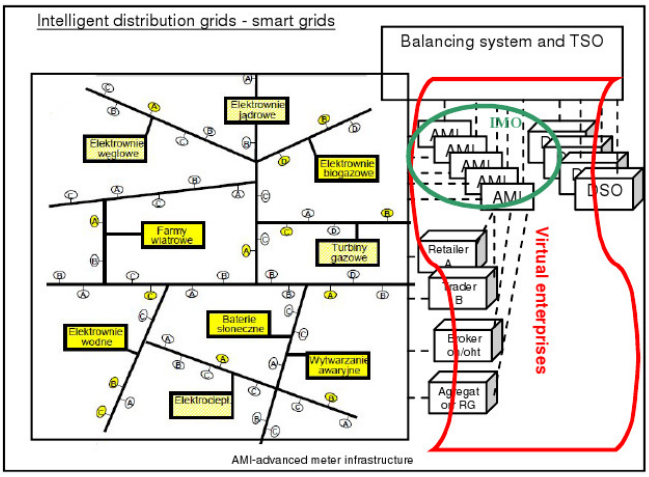


Fig. 1. The technical infrastructure required for the realization of intelligent network-general concept and the location for the IOM. (A, B, C, D: energy consumers), source: the basis on information from: www.ure.gov.pl

- Bi-directional flow of energy and information in real time (demand response) and implementation of the intelligent bi-directional remote power meters.
- Communication between market operators in real time.
- Monitoring network in real time.
- Operating and monitoring systems - operating in real-time and also effective integrating with AMI infrastructure.
- Adequate telecommunication infrastructure for data management systems.
- Power meters data management systems for the independent metering operator (IOM) - as a company with access to market participants and to the high quality, useful measurement data in real time.

Directive EU 2006/32 p. 13 defines the basis for the use and access to smart meters for customers and points out the need for billing based on actual energy consumption.

The main barrier to progress in work on smart metering are investment costs and scale of the complexity of possible solutions and the number of participants as well data obtained for instance every day for each hour (or more) for each end users.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

II. Metering Market Model

Currently, the market value of the energy is built around a primary object of manufacturing and trade - that is energy.

The analysis a market value defined in such a way reveals the weaknesses of the current market model. Mainly these arise from the presence of many distributors on the market which are operating on different, often internally regulated basis. This results in poorly organized connections between the other market participants - especially energy traders. Formal and legal rules which are contained in the Comprehensive Distribution Agreements (CDA) as well as in the IRiESDs only require from operators to expose the measurement data at specified times schedule but they are often in different formats, and in practice the timetable is not respected. The companies send the estimate and poor data to fulfill the terms of settlements with clients because of the lack of the adequate data. In addition, current regulations and a system of metering and billing create difficulties, if not actually preventing prosumers activities.

In the proposed target market model, where the value chain on the EM has been defined there is space for the market integrator and operator of measurement data - IOM company, which is separated from transmission and distribution operators (TSO and DSOs). A key role here is, however, the implementation of smart metering and data management system for all EM. The value chain on the EM is presented in Fig 2.

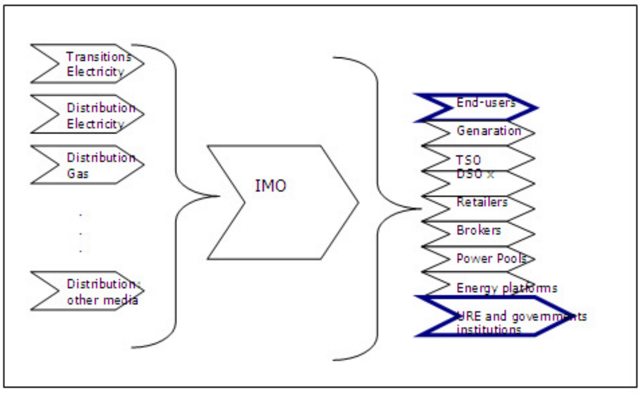


Fig 2. Metering market value chain (with IMO) in the target model of energy market. Development on the basis of [2],[3].

The role of the Independent Operator of Measurements will be: [1], [2]:

- Conducting regulated business activities centered around the acquisition and data processing devices (all tariff groups and data from the station medium / low voltage) obtained from the DSOs and the TSO and forwarding to the relevant players, empowered by law now and for future market players.
- Keeping the central repository of data measuring on the energy market.
- Measurement data processing and aggregating for those who needs the data from the IOM.
- Provision of services in information sharing on the energy market.
- Support competitiveness within the energy market through the actual separation and maintenance of the independent flow of information from the DSOs to the retailers.
- Providing a unified communication channel and standards, providing a base of information to the seller, regardless of his area of operation.
- Keeping archives of the measurement data, thus relieving DSOs from the necessity of maintaining a measurement data store other than necessary for their own needs.

III. Business Model for IOM

In Fig. 3 the main business processes for IOM are presented and described.

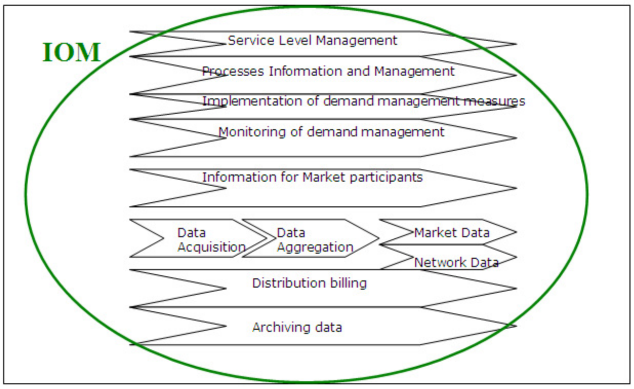

IOM
Service Level Management
Processes Information and Management
Implementation of demand management measures
Monitoring of demand management
Information for Market participants
Data Acquisition
Data Aggregation
Market Data
Network Data
Distribution billing
Archiving data

Fig 3. IOM- Main Business Processes. Development on the basis of [2], [3].

The most important business process, however, for the IOM is providing the highest quality and accuracy of the measurement data to all stakeholders of the market.

IV. ICT for Advanced meters infrastructure

ICT needs to be considered in specific areas of innovative technology solutions: the measurement data terminals for market participants, technical infrastructure for data transmission, information systems management measurements, users tools for measurement data and their household appliances etc.

Focusing only on the functionality of the data transmission system (excluding technology and communication standards, measuring meters the same: for all client household appliances, data reading and MDMS system) should be taken into account in following technological solution [5]:
Matching transition capacity to telecommunications traffic expected at different levels and using different technologies and network (LV and MV Network, plc, GPRS, LAN, wireless SDR, dedicated wired network, etc.):

- Respect traffic at one MDMS - up to 1 million meters - the reading of load profiles once a day every 5 hours, Standards: EDIFACT standard, IEC 61986, XML (plain) PTPiREE standards.

- Respect traffic at a hub of data - for 1000 meters. Standard Communications: DLMS / COSEM.

MDMS requirements:

- Scalability up to several million customers.
- Response time to an event with the counter <1min.
- The possibility of transmission of the command "off" in time <5min.
- The precise specification of the interfaces at the design stage to the meter reading systems, billing systems, SCADA and / or EMS systems, asset management systems.
- An Ease of publication / access data.
- MDM Systems towards to Architecture Service Oriented - SOA (Service Oriented Architecture)
- Corporate Bus Services (eng. Enterprise Service Bus) - adds a layer of a multilayer of information systems architecture that enables the use of the concept of SOA in a corporate environment. Its task is to join and disconnect services for the corporate information system.

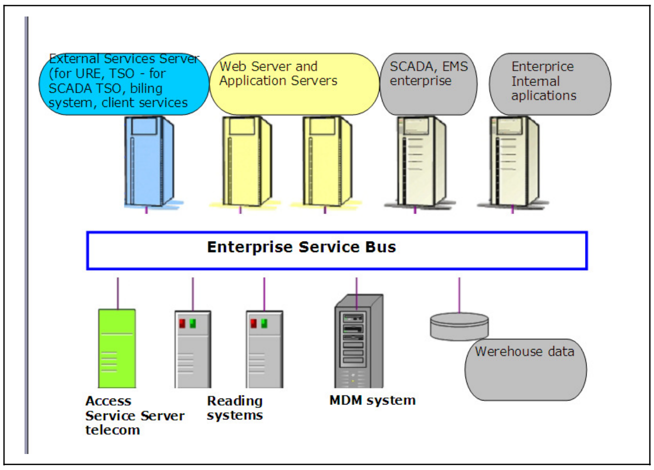

External Services Server (for URE, TSO - for SCADA TSO, billing system, client services
Web Server and Application Servers
SCADA, EMS enterprise
Enterprise Internal aplications
Enterprise Service Bus
Werehouse data
Access Service Server telecom
Reading systems
MDM system

Fig 4.ICT architecture design for IOM. Development on the basis of [5].

Currently, software solutions offered projects for Smart Metering and MDM systems are based on SOA (fig. 4).

In Poland, first projects presented by IBM and WINUEL are combining their software components such as a solution to support measurement and data acquisition (applications from Winuel) with WebSphere ESP solutions. The aim is jointly constructed and offered by the system (presentation - May 2010) designing its modularity, orientation towards support of open standards and flexibility to adapt to the requirements of the final customer.

Connected components there are:

1. Data acquisition system with different types and models of energy meters.
2. The system of central collection and validation of measurement data in a uniform format.
3. Handling the system of measuring consumers.
4. Grid measurement service system.
5. Bus Integration -WebSphere ESB that enables the connection of multiple acquisition systems and the conversion of interfaces to a common standard CIM. Bus will also allow the connection of other systems belonging to other utilities participating in the typical business processes of the AMM
6. WebSphere Process Server system to automate common business processes of the AMM
7. Dashboard's Websphere Business Space – allowing end users a comprehensive surveillance and monitoring of the AMM system based on defined KPI's (Key Performance Indicators), portfolio (procedures) tasks according to defined business processes AMM.
8. System Websphere Business Monitor - for the calculation of defined KPIs (Key Performance Indicators).

V. Conclusion

The main advantages of setting up the IOM according to the energy market analysts are as following [2]:

1. Providing quality service to the mandatory exchange of the measurement data in the process of mutual settlements, in the process of changing the supplier (a temporary regime of access to data) etc.

2. Ensuring neutrality and equal access to information for all market participants:

- Enhancing the competitiveness of the market: a real separation from the DSOs.
- Increased attractiveness for new entrants due to the elimination of barriers to the access of data.

3. Maintenance and use of open standards for the exchange of information between market participants:

- The advantage is the smaller number of standards to develop.
- Maintaining the freedom of market participants to choose their supplier of measuring tools (DSO can select the setting of solutions providers of smart meters).

4. Reducing the amount of work needed (changes) in existing systems:

- Each participant in the market fully integrates with only one player: the IOM;
- Reducing costs and the elimination of financial barriers for the entry of new players (especially for retailers).

5. Ensuring finance transparency of new business projects (smart projects) by all market participants.

6. Simplifying data collection on the market and customer behavior.

7. Opening up the EM in terms of power system data exchange with network operators in Europe.

However pilots are needed to implement, both remote reading (such as the implementation of the company Energa: Converge system, system-Amateur Radio) as well as ICT systems, built in SOA architecture, where virtual platforms for trade and management of the measurement data acquisition and archiving are in the IOM set of services.

Work on standardization, smart metering, acquisition, storage, sharing and reporting, and management of the measurement data through the ICT systems are currently in progress. However, due to the complexity of the issues involved and the scale and diversity of systems already operating on the market this is envisaged to be a highly expensive and complicated design.

Appendix

Short definitions of the term "**virtual**":

1. *Virtual*- means: such as: almost, virtually- so it is something that has potential (through the development and implementation of the new ICT tools-it is greater then ever before). The characteristics of the object does not physically exist, or exists in a sparse and economical form. 'Nearly' and 'almost' does not mean 'better' or 'worse'. Usually it is something more than just a physical object, being value-added and built on its virtual features.

2.*Virtual* means digital – that is, a facility that can be stored in computing systems and databases (broad approach).

Virtual Energy Market: (virtual actors), - approach to the construction of the structure of the market applying a similar model to the Internet and using its potential of communication and information processes for the management at the EM.

CVPP – Commercial Virtual Power Plant- - distributed generation, the object dispersed, potentially possible, flexible, dynamically variable in its action, quickly responding to market needs and the energy balance system. CVPP applies intelligent information management systems and knowledge. All processes can not be managed without the support of ICT tools. CVPP is a virtual organization dispersed with EM actors, managed by information management systems that aggregate the market in terms of "above the network" [9].

Abbreviations

DSOs – Distribution System Operators

TSO – Transition System Operator

AMI – Advanced Metering Infrastructure

AMM –Advanced Metering Management system

AMR- Advanced meter reading

MDMS Measurements Data Management System

CVPP Commercial Virtual Power Plant

IRiESD - the policy for DSO activities

CDA – Comprehensive Distribution Agreement (in Polish: GUD- Generalna Umowa Gystrybucyjna)

URE- Energy Regulator on the EM which regulate a policy and principles of market (in Polish: Urząd Regulacji Energetyki).

References

[1] Open Meter (2009): Report on regulatory requirements

[2] Raport PTPIREE, (Instytut Energetyki Oddział Gdańsk Jednostka Badawczo Rozwojowa, Ernst & Young Business Advisory, 2010): „Studium wdrożenia inteligentnego pomiaru w Polsce".

[3] Ogólny model rynku opomiarowania, HP dla PSE operator ,projekt, marzec 2010.

[4] Ustawa Prawo Energetyczne 1997 wraz z ostatnimi zmianami (2010).

[5] A. Babś, Instytut Energetyki Gdańsk 2010, materiały internetowe

[6] Matusiak B. E., Pamuła A. Zieliński J. S.; New Idea in Power Networks Development, X Międzynarodowa Konferencja Naukowa "Planowanie rozwoju, eksploatacja i zarządzanie w energetyce, PE2010, Wisł,a 8-10 września 2010

[7] Pamuła A., Zieliński J., S.; Electric energy – importance, problems and solutions. Technology Policy and Innovation, 6-8 July 2005, Łódź, 77-81.

[8] Matusiak B. E., Soltowski J.; Modelowanie rynku energii elektrycznej dla potrzeb inwestycyjnych w energetyce. „Technologie wiedzy w życiu publicznym, red. J. Goluchowski, Katowice 2009" ISBN 978-83-7246-595-5.

[9] Matusiak B. E., Pamuła A. Zieliński J. S.; Rola zarządzania informacją w procesie kreowania wirtualnego rynku elektroenergetycznego. „Komputerowo Zintegrowane Zarządzanie", Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją", Opole 2008, t.II, 113-123.

[10] Zieliński J. S.; Rola teleinformatyki w środowisku sieci inteligentnych „Rynek Energii" nr.1, luty 2010, 16-19. (czasopismo z listy)

[11] Pamuła A. Zieliński J. S.; Mikrosieci – racjonalne wykorzystanie lokalnych źródeł energii odnawialnej. Technologie wiedzy w życiu publicznym, red. J. Goluchowski, Katowice 2009" ISBN 978-83-7246-595-5. 423-430

[12] Matusiak B. E., Pamuła A. Zieliński J. S.; Technologiczne i inne bariery dla wdrażania OZE i tworzenia nowych modeli biznesowych na krajowym rynku energii, X Międzynarodowa Konferencja Naukowa "Planowanie rozwoju, eksploatacja i zarządzanie w energetyce, PE2010, Wisł, 8-10 września 2010

[13] Jagoda G., Pamuła A. Zieliński J. S.; Some Remarks on Microgrid Penetration in Polish Distribution Networks. „The European Electricity Market EEM07", May 23-25, Cracow, 105-109.

[14] Zieliński J. S.; Review of Selected Problems in Distribution Networks with Dispersed Generation. "The European Electricity Market EEM07, May 23-25, Cracow, 79-87

[15] Dyrektywa EU 2006/32

[16] http://www.skaden.pozyton.com.pl/read.htm

[17] IT for Power Sector, http://ogrzewnictwo.pl/index.php?akt_cms=4887&cms=35, [25.05.2010]

[18] Wdrożenia wspomagające zarządzanie. Systemy ERP w energetyce http://ogrzewnictwo.pl/index.php?akt_cms=5217&cms=35, [25.05.2010]

[19] Standardy techniczne systemu WIRE ; na stronie PSE operator; http://www.cire.pl/RB/komunikaty/Standardy_techniczne_systemu_WIRE [25.05.2010]

ISBN 978-83-60810-27-9
ISSN 1896-7094

A two-level algorithm of time series change detection based on a unique changes similarity method

Tomasz Pełech-Pilichowski
AGH University of Science and Technology
Department of Applied Computer Science
Faculty of Management
ul. Gramatyka 10, 30–067 Krakow, Poland
Email: tomek@agh.edu.pl

Jan T. Duda
AGH University of Science and Technology
Department of Applied Computer Science
Faculty of Management
ul. Gramatyka 10, 30–067 Krakow, Poland
Email: jdu@agh.edu.pl

***Abstract*—In the paper, a novel two level algorithm of time series change detection is presented. In the first level, to identify non-stationary sequences in processed signals preliminary detection of events is performed with short-term prediction comparison. In the second stage, to confirm changes detected in first level a unique changes similarity method is employed. Detection of changes in non-stationary time series is discussed, implemented algorithms are described and results produced on exemplary four financial time series are showed.**

I. Introduction

INFORMATION on changes in analysed time series is relevant to detect alarm situations, in particular, when signals are processed in real-time systems. Implementation of dedicated algorithms to event detection from non-stationary time series requires considering many factors, such as statistical and frequency-domain data characterisation, different type of short- and long-term changes, time lags between events or sampling frequency. Results given with such algorithms are helpful in advanced data analysis, prediction and finally—decision making process.

Our research are aimed at advanced algorithms of time series change detection, based on statistical approach, signal analysis [4] and employing immune paradigm to event detection support [15]. In particular, we proposed a method to gain information on early symptoms of significant changes by analysis of short-term prediction efficiency [14], [16], [13], [15] through comparison between zero-order-prediction (ZOP)/zero-order-hold (ZOH) model [1] and adaptive Holt predictor [9]. Proposed method is suitable for a particular type of changes in signals, for example occur in financial time series which consist of many non–random components. We also showed that application of artificial immune systems techniques [8], [21] is a way to improve event detection and thus prediction efficiency (especially prediction error variance). In the paper [15] we proposed original idea of implementation immune based approach to early detection of significant changes in time series (long term changes of mean value), where detection is decomposed into two stages: fast detection of nonselfs (employing fast statistical and prediction procedures) and then more accurate (and more time–consuming) recognition of the nonself type and the system recovery. In the papers [15], [18], [17] an improvement of this idea aimed at long term event detection were described.

The aim of this paper is to provide detection method of changes in diagnostic signals to early detection of emergency situations. It has been designed for monitoring a set of diagnostic signals (medical, technical, financial ones) to capture differences, i.e. unusual behavior of selected signal (process variables). Such solution may be implemented as detector of rapid changes in signal (for example alarm situations as dangerous changes of technical process temperature, liquid pressure) or exceeding the threshold valued of process limitations (technology losses minimisation).

We present and analyse two-level algorithm of change detection in time series, based on simultaneous processing of two time series of fixed length in a moving window, to capture unique changes—which occur only in one of two processed diagnostic signals, i.e. are not results of the same external factors. In the first stage, preliminary detection of events is performed with one-step-ahead prediction comparison given with methods dedicated to stationary or non-stationary data [19], [1], [13]. In the second stage, to confirm changes detected in first level, a unique changes similarity method is employed.

Proposed event detector is tested on financial time series—stochastic processes with unknown random inputs, which are hard-predictable and difficult to perform reliable statistical analysis, compared to technical ones. Therefore, efficient event detection in financial ones with proposed algorithm will indicate its applicability to technical time series (as a starting point for further implementation and adjustment in order to detect well-defined, specific changes).

In this paper, we describe the detection algorithm as a part of large detectors set capable of—through two signals

This work was supported by the European Regional Development Fund, Grant no. UDA–POIG.01.03.01–12–171/08/00.

concurrent processing—handling possible event patterns in processed time series. Such idea can be further developed as an autonomous event detector and implemented as one component (e.g. agent) of available set supervised by T-lymphocyte (immune-like event detector [15]).

II. Event detection from time series

Event detection from time series is aimed at identifying short and long term periods of uncommon series behaviour, analysed in moving window, to detect a change in stationarity due to random external factors. Event detection task may be viewed as finding corresponding probability distribution [13], [14], [16] or as unsupervised classification task (one-class classification) where one may describe only one class and a method to distinguish between possible object (decision boundary between normal and anomaly class) with appropriate (training set tested) mapping function [19].

Considering long-term signal changes, detection of short-term "announcing" events may be a way to perform faster computation compared to classical, but robust statistical procedures of long-term changes detection. Such events may precede statistical properties changes in processed series and generate the need of adjust of assumed analysis window width, i.e. change of detection delay thus probabilities of false alarm and undetected events [12].

An implementation of given algorithmic method depends on input data properties (statistical, frequency, dimensionality, completeness), attributes of events (amplitude, duration, periodicity, applicability, coincidence, delay) or permissible detection error. Many algorithms require adjusting parameters and signals selection to specific conditions of dataset. In practice, usually the most important is appropriate selection of classification algorithm which directly affects the detection reliability and a possibility of implementation for heterogeneous data sets.

An application of a detection method suitable for dataset requires is to find specific attributes of events and the use of dependencies between processed signals and occurring events (often delayed). To achieve this aim, a number of methods can be exploited to time series quantitative analyses, such as analysis of the frequency of events [10], analysis of trends, patterns and characteristics similarity [12] or statistical methods for testing deviants and outliers, algorithms based on neural networks, genetic algorithms [11] and other data mining techniques applied to event detection from time series [7], [10], including methods based on similarity measures, in particular distance ones [22]. However, reliable results produced by such algorithms usually depend on length of series or learning [15] and highly accurate estimation of parameters such as mean value, information about trend, dynamics parameters and random component dispersion—which are an input for statistical tests [20]. In this paper we focus on methods based on similarity measures and signal analysis.

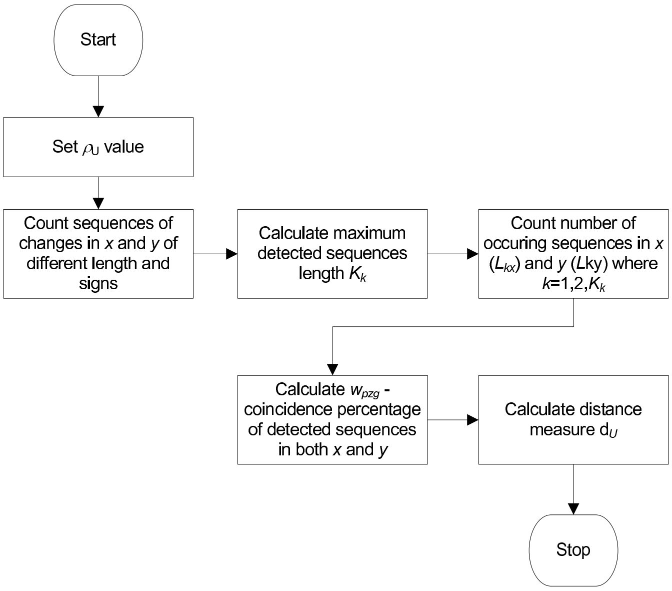

Start
Set ρ_U value
Count sequences of changes in *x* and *y* of different length and signs
Calculate maximum detected sequences length K_k
Count number of occuring sequences in *x* (L_{kx}) and *y* (*L*ky) where *k*=1,2,K_k
Calculate w_{pzg} - coincidence percentage of detected sequences in both *x* and *y*
Calculate distance measure d_U
Stop

Fig. 1. A block diagram of unique changes similarity algorithm.

III. A unique changes similarity method

Considering event detection from one time series, unusual behavior (non-stationarity) may be recognised as abrupt changes of mean value (visible as deviations exceeding fixed threshold), sequences of changes (of the same or different signs) or patterns understood as specific configuration of deviations. Considering a pair of time series, original (unique) event detection may be viewed as a selection of events occurring only in one processed dataset. To capture such events we propose distance-based similarity measures [5] applying in a moving window.

One of novel measures, designed and developed for a particular problem is an unique changes similarity method (distance-like one), first mentioned in our earlier papers [5], [17], denoted as method U and measure dU. It is dedicated to identify unique changes which are considered as sequences of deviations of the same sign with different sequence length (1,2,...,N) exceeding an arbitrary fixed threshold—ρ_U in two processed diagnostic signals of the length N (constant moving window width N is assumed).

The goal is to calculate a similarity coefficient—distance measure dU (see equations 1–3) based on counting of sequences of fixed length (subsequences of different length or signs are treated as different ones). A block diagram of such detection for one sample (for assumed window width N) is shown in figure 1.

'Distance measure' term is used instead of *distance* or *metric* because all required conditions related to formal definition of metric are not satisfied, in particular symmetry condition.

To compute dU value three partial calculations are performed (eq. 1–3). Coincidence percentage of detected sequences of deviations in analysed signals is calculated as:

$$w_{pzg} = w_p \sum_{k=1}^{K_k} k \cdot \min(L_{kx}, L_{ky}) \tag{1}$$

where L_{ks} denotes a number of sequences of length k in series s (x or y). To compute coefficient wp the following formula is employed:

$$w_p = 1/(\sum_{k=1}^{K_k} k \cdot \max(L_{kx}, L_{ky})) \tag{2}$$

Finally, distance measure dU is computed as:

$$\mathrm{d}U = 1 - w_{pzg} \tag{3}$$

Notice, in case of similar sequences in both series dU will have 0 value ($wpzg$ near 1). As threshold ρ_U value, standard deviation in moving window or its multiplicity may be used, however, arbitrary adjusting may be valuable.

To avoid an impact of time delay between events, for instantaneous values of computed distance measures dU, a tolerance (denoted as L_{tol}) is assumed as permissible delay between analysed signals. For each time instant t, a number (L_{tol}) of distance measures dU_n is calculated for time instants n ($n = t - L_{tol} + 1, \ldots, t$). Finally, d$U_n$ for which the lowest value was obtained is taken as the measure dU between two subseries for sample t.

IV. Two-level change detection algorithm

Proposed algorithm is designed to different types of diagnostic signals. We have chosen financial time series as difficult ones to perform event detection or forecasting because of existing many non-random components. Notice, that financial time series are available from heterogeneous sources, however, causal-consecutive dependencies between time-lagged events are possible.

Proposed hybrid algorithm performs detection in two steps:

1) preliminary detection of short-tem non-stationarities in signals;
2) unique changes detection with method U (see eq. 1–3).

Considering detection **in the first level**, we have chosen an approach based on one-step-ahead prediction error comparison with methods efficient for stationary and non-stationary data [14], [16], performed for both signals separately.

Autocorrelation of daily increments of typical financial time series is statistically insignificant [3], thus during usual behavior no autocorrelation of daily returns may be expected [1]. The most efficient short-term prediction (the minimum variance error) is achievable by extrapolation of averaged returns [15], i.e. by the ZOP or ZOH model [1], [15]. To predict non-stationary data, three-parameters adaptive Holt method is employed (parameters α, β, γ adjusted for each sample) [13], which is more flexible than classical one [9]. The comparison allows to basic identification of non-stationarities, i.e. non-random components (advantage Holt method over ZOP/ZOH) in relatively short time and low computational resources consumption.

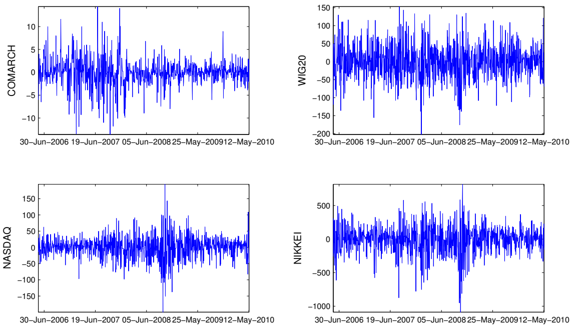


Fig. 2. Examined financial time series - daily returns (one-day increments).

In the second level, two signals are taken to calculate unique changes similarity measure (dU) thus differences between found sequences of deviations. Finally, change in a signal is found after non-stationarity confirmation with dU value exceeding fixed threshold ρ_U. In this stage, only unique (original), non-stationary sequences are selected (accuracy depended on ρ_U).

Detection schema described above provides a possibility of minimizing false alarms as a result of identifying changes that are, in fact, random fluctuations. In addition, proposed two-level algorithm decrease computation time consuming which through switching similarity detection method U for relatively small amount of samples only.

V. Calculation results

Four financial time series, individual (Comarch) and aggregated (WIG20, Nikkei, Nasdaq), has been used as exemplary dataset to analyse the proposed two-level detection algorithm. We have chosen local (Comarch, WIG20) and global time series (Nikkei, Nasdaq) from 2006–05–22 to 2010–05–14 (1040 samples).

Financial time series (see daily returns depicted in fig. 2) represent stochastic processes with unknown random inputs (in the simplest approach, such series can be considered as Wiener processes). Acceptable detection results obtain on such hard-predictable sequences of samples allow to assume that proposed solution applied to diagnostic signals generated by technical devices (which don't contain many variable components) will result in more accurate event detection.

For each processed time series, subsequent samples were taken to compute one-sample-ahead prediction with naive method (level 1; exemplary results of preliminary detection are shown in table I—for all analysed series non-stationarity identified for about 5%) while distance measure dU was calculated in moving window of constant length $N = 22$ samples (see fig. 1) between series 1 and 2 and in a reverse order (because of asymmetry of dU measure mentioned above). Processed signals were unified with standard deviation to achieve comparable orders of magnitude.

In the example discussed in this paper, to provide detection reliability, threshold sequence length was fixed to $\rho_{PC} = 3$.

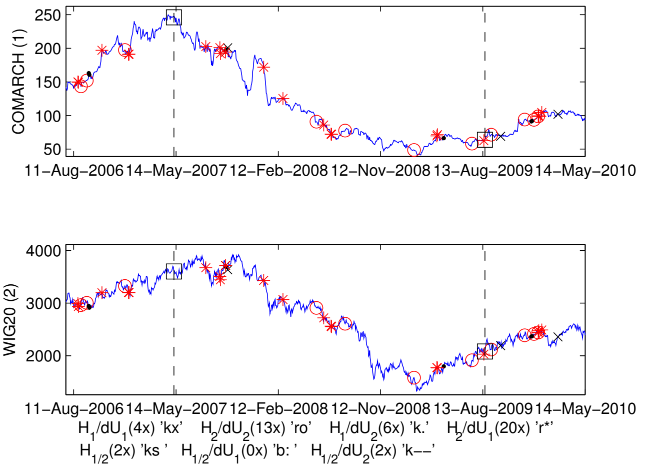


Fig. 3. Results of change detection for Comarch (1) and WIG20 (2)—depicted with solid lines—with proposed two-level algorithm. Recognized non-stationarity in series no.1 confirmed with dU_1 denoted as 'cross' (H_1/dU_1) and dU_2—as 'dot' (H_1/dU_2); non–stationarity in series no.2 confirmed with dU_1—'asterisk' (H_2/dU_1) and dU_2—'circle' (H_2/dU_2); changes H_1 and H_2 detected for the same samples—'square' ($H_{1/2}$), changes ($H_{1/2}$) confirmed with dU_1—'dotted line' and with dU_2—'dashed line'.

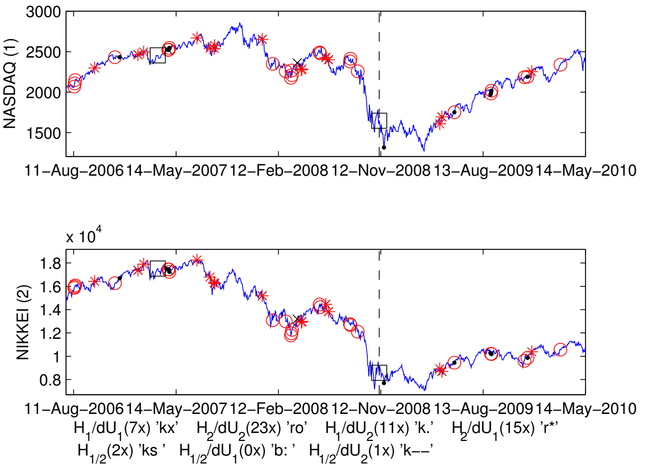


Fig. 4. Results of change detection for Nasdaq (1) and Nikkei (2)—depicted with solid lines—with proposed two-level algorithm. Description of symbols: see fig. 3.

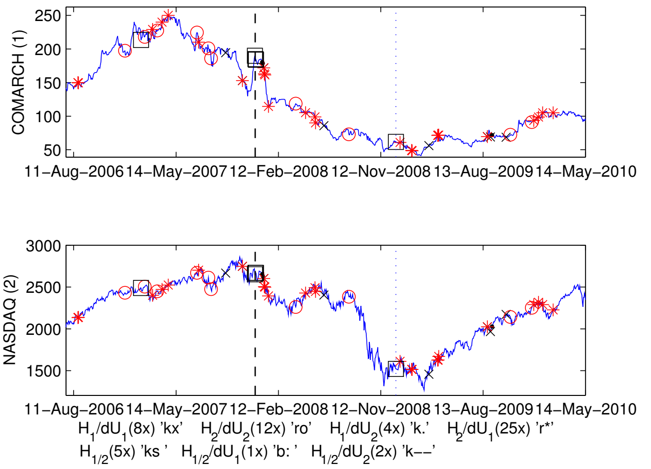


Fig. 5. Results of change detection for Comarch (1) and Nasdaq (2)—depicted with solid lines—with proposed two-level algorithm. Description of symbols: see fig. 3.

The main parameter of the second-stage detection level (ρ_U) was set to 1 as an equivalent of signals standard deviation after performed unification. Values of these parameters were selected in an experimentally way (related to properties of four analysed time series) to achieve the smallest number of false alarms and undetected events.

Detection results produced with proposed two-level algorithm are depicted in figures 3–6 (with the sampled series in the background). The figures show confirmations of changes with short-term prediction comparison in the first level (denoted as H_1 and H_2) and distance measure dU in the second level (eq. 1–3).

It may be seen in figures 3–6 that proposed method allows to detect local changes (denoted as 'cross', 'asterisk', 'circle' and 'dot'—see figure 3 caption), obtained with simple H_1 or H_2 confirmation with method U, and long-term changes, i.e. reversal in a trend (see vertical lines in fig. 3–6).

In the first case, a number of short-term changes were detected correctly, however, false alarms were found for some samples. Application of non-stationary detection confirmation (level 1) significantly reduces overall number of detected events (see table I and figures 3–6 caption).

In the second case, H1 and H2 changes are synchronised (non–stationarities detected in one signal only are rejected), which is illustrated as 'squares' in figures 3–6, and then—confirmed with dU_1 or dU_2 measure. This is relevant step to eliminate false alarms.

VI. Conclusions

Proposed two-level algorithm of change detection in time series may be viewed as an inspiration for construction efficient detectors, adaptable to signal properties. It was found as a suitable for short- and especially long-term changes of mean value through confirmation of non-stationary subsequences of processed signal. Acceptable detection results obtained on financial time series (stochastic processes with unknown random inputs, with many non-random components) allow to assume applicability end effectiveness for other types of time series, in particular, diagnostic signals generated by technical devices.

Calculation results presented in this paper show that to achieve reliable detection of symptoms of rapid changes in long-term trends in one processed time series, processing of series set of the same class (in the paper—financial ones) is valuable. Depending on the set (pair) of signals taken to calculate distance measure, detected changes may occur for different samples. This issue will be developed towards (1)

TABLE I
EXEMPLARY RESULTS OF DETECTION OF NON-STATIONARY SUBSEQUENCES IN PROCESSED TIME SERIES

Time series (total number of samples: 1040)	Comarch	WIG20	Nasdaq	Nikkei
Number of detected non-stationary subsequences with one-step-ahead prediction comparison	60	50	42	59
Percentage	5.77%	4.81%	4.04%	5.67%

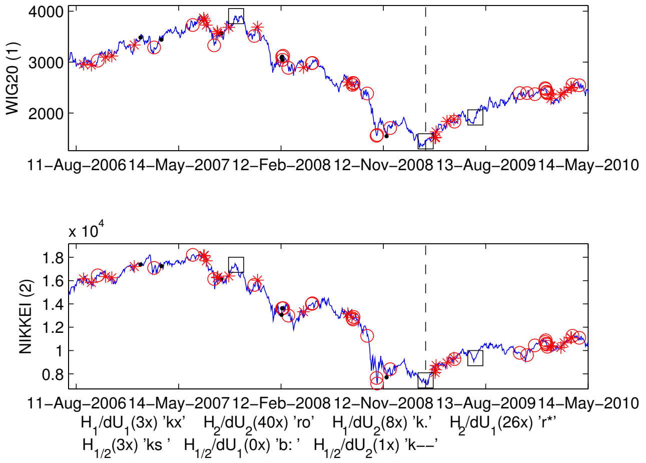


Fig. 6. Results of change detection for WIG20 (1) and Nikkei(2)—depicted with solid lines—with proposed two-level algorithm. Description of symbols: see fig. 3.

events recognition and (2) forecasting for one selected (based) signal through processing signals sets consist of many time series datasets.

Further research will be focused on implementing and testing of the proposed solution for technical diagnostic signals, especially processed in real-time systems. Studies will be required on implementation and testing of similarity methods dedicated to specific changes and patterns appearing in analysed signals, including time delays between events and elimination of false alarms. Further improvement appears to be obtained by modification of described distance measures towards variable length of subsequences taken to compute measure dU (to this aim, fuzzy logic rules may be employed).

Possibility of adjusting parameters (threshold values, window widths, etc.) seems to be a promising way for further enhancement of the presented method which is suitable for the idea of immune-based adaptive detection, in particular as one of many detectors supervised by T-lymphocytes.

REFERENCES

[1] Box G .E. P. and Jenkins G. M., *Analysis of Time Series* PWN, Warszawa, 1983

[2] Duda J. T., *Dobór parametrów algorytmu Page'a-Hinkleya przy ustalonych prawdopodobieństwach I i II Rodzaju* Unpublished paper, AGH–UST, 2005

[3] Duda J. T. and Augustynek A., *On possibilities of improvement of short-term predictions of stock indices with regression models* Company Management in Conditions of European Integration—Part 2. Economy, Informatics and Numerical Techniques, Ed. M.Czyz and Z.Cieciwa, AGH–UST University Press, 2004

[4] Duda J. T. and Pełech T., *Wykrywanie zdarzeń w szeregach finansowych z wykorzystaniem metod statystycznych* [In:] Inżynieria wiedzy i systemy ekspertowe, T.2, Edt. Grzech A., Oficyna Wydawnicza Politechniki Wrocławskiej, 2006

[5] Duda J. T. and Pełech-Pilichowski T., *Miary podobieństwa szeregów czasowych w detekcji zdarzeń* [In:] Systemy wykrywające, analizujące i tolerujące usterki, red. Kowalczuk Z., PWNT, 2009

[6] Farmer J. D., Packard N. and Perelson A., *The immune system, adaptation and machine learning* Physica D, vol.22, 1986, pp.187–204

[7] Guralnik, V. and Srivastava, J., *Event detection from time series data* [In]: Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data mining, San Diego, California, USA, 1999, pp. 33–42

[8] Hofmeyr S. A. and Forrest S., *Immunity by Design: An Artificial Immune System* Proceedings of the Genetic and Evolutionary Computation Conference, San Francisco, 1999

[9] Holt C. C., *Forecasting seasonals and trends by exponentially weighted moving averages* Carnegie Institute of Technology, Pittsburgh, Pennsylvania, 1957

[10] Keogh E., Lonardi S. and '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, pp. 550-556, Edmonton, Alberta, Canada, 2002

[11] Kingdon J., *Intelligent Systems and Financial Forecasting* Springer, 1997

[12] Mahoney M. and 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

[13] Pełech T., *Adaptive Holt's Forecasting Model Based on Immune Paradigm* Problemy osvoenia poleznyh iskopaemyh. Zapiski Gornogo Instituta, Sankt Petersburg State Mining Institute, 2006

[14] Pełech T. and 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

[15] Pełech T. and 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

[16] Pełech T. and Duda J. T., *Immune Algorithm of Stock Rates Parallel Monitoring* Information Systems and Computational Methods in Management, Ed. Duda J.T., AGH-UST University Press, 2005

[17] Pełech-Pilichowski T, *Zastosowanie metod algorytmicznych w prognozowaniu szeregów finansowych* [In:] Zarządzanie organizacjami - finanse, produkcja, informacja, red. Howaniec H., Waszkielewicz W., Wyd. Akademii Techniczno-Humanistycznej, Bielsko-Biała, 2009

[18] Pełech-Pilichowski T. and Duda J. T., *Wykorzystanie podejścia immunologicznego do prognozowania szeregów czasowych* Automatyka: półrocznik Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie, 2009

[19] Somayaji A. B., *Operating System Stability and Security through Process Homeostasis* Ph.D. thesis, University of New Mexico, 2002

[20] Wetherill G. B. and Brown D. W., *Statistical Process Control. Theory and Practice* Chapman and Hall, 1991

[21] Wierzchon S.T., *Sztuczne systemy immunologiczne. Teoria i zastosowania* Wyd. Exit, 2001

[22] Yang K. and Shahabi C., *A PCA-Based Similarity Measure for Multivariate Time Series*, ACM International Workshop on Multimedia Databases, 2004

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 265–269

ISBN 978-83-60810-27-9
ISSN 1896-7094

STRATEGOS: A case-based approach to strategy making in SME

Jerzy Surma
Warsaw School of Economics
Collegium of Business Administration, Al.
Niepodleglosci 162,
02-554 Warsaw, Poland
Email: jerzy.surma@sgh.waw.pl

Abstract—**Making strategic decisions in an enterprise is one of the most difficult problems of management. It is a result of unstructured character of such decisions which are made in conditions of high uncertainty. This issue is particularly important in case of Small and Medium Enterprises (SME), where Chief Executive Officers (CEO) are lacking support in this area and most often act intuitively, being convinced that their business is unique. Recent researches on decision-making point out the substantial influence of referring to analogies and self-experience in strategic problems. Accordingly it is proposed to use case-based reasoning to build STRATESOS - the strategic decision support system. Then, the system was verified in a survey by dozens of CEOs from SME. The results of the survey are promising and show the remarkable correspondence of the proposed solution with expectations and strategic behavior of CEOs.**

I. Introduction

SMALL and medium enterprises play a key role in the contemporary market economy. Strategic decision making in this kind of companies is one of the crucial issues taken into consideration in management [1]. In order to cope with the uncertainty and complexity, SME decision-makers seem to behave according to bounded rationality theory [2]. They determine certain level of preferences and as soon as the choice which satisfies the set up criteria becomes available, they accept it. In this approach CEOs are looking for satisfactory choices, not for optimal ones. The major challenge of decision making is uncertainty, and the major goal of decision analysis is to reduce this uncertainty [3]. To deal with the increasing complexity of the environment, the SME entrepreneurs create ‘shortcuts’ in their thinking. In this context there are two possible approaches [4]: routine (habitual and reactive) and intuitive decision making. As we can see developing and formulating the strategy of an enterprise is related to the fundamental questions concerning experience, knowledge and intuition of managers. In case of any new problem, when deductive reasoning is limited, it turns out that appealing to experience is by all means a rational behavior. As Thagard points out, “analogies can be computationally powerful in situations when conceptual and rule-based knowledge is not available” [5]. As strategic management is concerned, the team from Harvard Business School [6] made similar statements: „Reasoning by analogy is a common form of logic among business strategists. Facing a novel opportunity or predicament, strategists think back to some similar situation they have faced or heard about, and they apply the lessons from that previous experience“. Using analogies in strategy planning has been thoroughly researched both by means of case study analysis [7] and experimental research with the use of the NK-model [6]. Using the NK-model and referring to the concept of business landscape made possible to conduct formal research on using analogies in complex decision situations. On other the hand this approach has been subject to criticism, which suggested referring to other paradigms and hybrid approaches to reasoning [8].

The use of the mentioned model of CEOs behaviors may be used in creation of information decision support systems. In this context, the Case-Based Reasoning (CBR) seems to be the suitable paradigm for practical usage–vide chapter II. The concept and the prototype of strategic decision support system will be presented in chapter III (implementation issues) and IV (case-based reasoning cycle). The empirical verification is described in chapter V, and final conclusions are presented in chapter VI.

II. CASE-BASED REASONING FRAMEWORK

The research problem mentioned above might be solved, at least partially, by applying the case-based reasoning. It results from the following characteristics of these system [9]. Firstly, a particular case is the basic element of knowledge representation. Subsequently, the acquisition of knowledge consists of analyzing the particular cases from past experience and therefore it is not necessary to establish rules in order to generalize knowledge. Secondly a relatively easy update/expansion of the system through adding new cases, which follows the process of remembering one’s experiences. And finally excellent and credible justification for the recommendations (solutions) for business users. These ex-

This work was financially supported by the Ministry of Science and Higher Education in the budget for 2008-2010 as a research project.

ceptionally favorable characteristics result, first of all, from expert knowledge gained through relying on specific, individual cases solved by an expert in the past. Additionally a great opportunity is based on the CBR working cycle described in the figure 1, reflecting reasoning by analogy [10], where: New Case – the new problem, Retrieved Case – the retrieved case that is similar to the New Case , Solved Case – the solution that is adapted from Retrieved Case and is proposed to the New Case, Tested/Repaired Case – verify/test of the Solved Case, Learned Case – retain in the case based Tested/Repaired Case. However, so far there has been no thorough research on applying case-based reasoning to support strategic decision making. There has also been no reliable empirical research conducted to verify this approach and to assess how useful it actually is in the strategy making practice. At the same time global consulting companies have been building systems of databases containing information on the strategy consulting projects they have carried out with a possibility to adapt them for new clients.

III. Strategos Implementation

Based on the case-based reasoning framework, the prototype of strategic decision support system STRATEGOS was created. In order to function, case-based reasoning framework requires three main components: case representation, general knowledge representation, and similarity measure.

Case representation

The case representation should reflect the company itself (company description), its market environment (context description), and one or more strategic decisions taken in this particular situation. In order to establish it properly we conducted several surveys with CEOs of the selected SMEs. Based on those interviews and the research on the case representation for SMEs [11] the following case description was established as the set of the attributes that are taken into account:

1. Company description: market share, location, products/services, number of employees, sales volume (trends in at least two years period), sales volume (export), EBITDA (trends in at least two years period), B2B/B2C.
2. Context description: industry, industry life cycle phase, Porter five forces analysis (threat of substitute products, threat of the entry of new competitors, intensity of competitive rivalry, bargaining power of customers, bargaining power of suppliers).

The act which is chosen as a solution for the current problem, is based on the list of all the combinations of product/market decision based on the Ansoff matrix (product × market) [12], and positioning decision based on the Porter's generic strategies [13]. The complete case representation is representing the case in time T_1 (planned act/decision – after reuse phase) and T_2 (realized act/decision – after revise phase) – see figure 2. Additionally, all information included in the case representation might be enhanced with text, images, files, hyperlinks, etc.

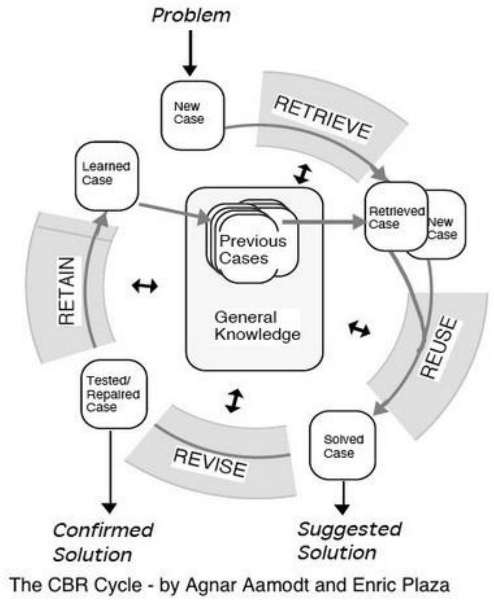


Fig. 1 Case-based reasoning cycle [10]

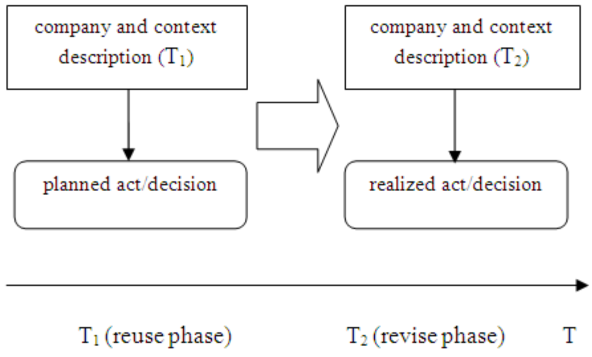


Fig. 2 Case representation in STRATEGOS

Similarity measure

Most of the case-based approaches retrieve a previous case based on the superficial syntactical similarities. The same is true in the STRATEGOS implementation where similarity measure merges the company and context description (in moment T_1 – fig.2) in one vector $a = (a_1, a_2, \ldots, a_n)$ of attribute values. If we denote by A_i the domain of the i-th attribute, then similarity measure is:

$$s: A_1 \times A_2 \times \ldots \times A_n \rightarrow [0,1]$$

and similarity between the input case a^I and retrieved case a^R is:

$$s(a^I, a^R) = 1 - \frac{\sum_{i=1}^{n} distance(a_i^I, a_i^R)}{\sum_{i=1}^{n} w_i}$$

where $w_i \in [0,1]$ is the weight of i-th attribute, and the *distance*$(a^I_i, a^R) \in [0,1]$ equals to the normalized Euclidean distance in the case of numeric attributes and the discrete distance measure (which takes the value 0 if $a^I_i = a^R_i$ and 1 otherwise) in the case of categorical variables [14]. Due to this similarity measure, it is possible at least to limit “false analogies” [6], cases that are importantly dissimilar and not useful as reference examples.

General knowledge

One of the critical issues is connected with general knowledge (domain ontology) representation. This knowledge is important during the re-use phase of the case-based reasoning cycle. Thanks to this knowledge it is possible to adapt the proposed solution from the retrieved case to the new case. Unfortunately, the strategic decision process is extremely complex, and it is difficult to represent the ontology completely. Despite this, we decided to use general knowledge in our approach as a warning system in situations where the solution proposed is unrealistic for formal reasons. There are the warning examples expressed as the if-then rules:

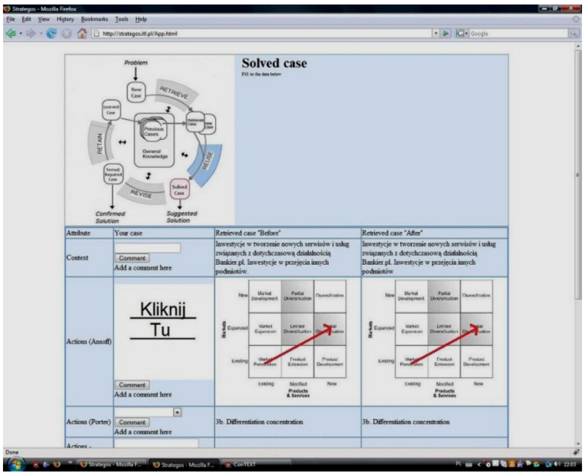
1. Inappropriate case retrieval based on the rule: “**if** the new case company and the retrieved company are in the different industry life cycle phase **then** the proposed decisions might be wrong**”**

2. Inappropriate proposed solution based on the rule “if the company has just started to penetrate the market with current products **and** proposed decision is: intensive foreign market development **then** this is risky and unrealistic proposal”

IV. STRATEGOS CBR CYCLE

Based on the CBR cycle (see fig.1) and the formal description the entire STRATEGOS decision making functioning will be described. We assume that the specific input problem is given by the CEO. The task is to establish the proper strategic decision (act) for the given problem. The STRATEGOS problem solving cycle consists of 4 phases:

1. **Retrieve:** The solution is retrieved from the case base basing on the similarity between the new case and cases already stored in the case base. The retrieved cases are shown to the user ranked based on the similarity value. Every choice is verified through the general knowledge in order to avoid unrealistic proposals.
2. **Re-use:** After the retrieve phase, it is possible to establish an act for the new case (see figure 3), after that the new case is called a solved case. The main goal of this phase is to give inspiration and/or verification, as well as propose rational choices based on the retrieved cases to the management board. Finally every proposed solution is verified by the general knowledge.
3. **Revise:** The solved case that was established in the previous phase has a planned strategic decision (see figure 2). This is a kind of proposal for the strategic actions plan. The most important goal of the revision phase is to recognize what has actually happened with that company after the strategic decision was taken. It is important to underline that the result does not depend only on described situation and made decision. In reality, there are several factors of different type such as economic trend, customers’ behavior, organizational atmosphere in the company etc. that have impact on the final result. Those factors may be provided in documents attached to case description.



Solved case

Fill in the data below

Attribute	Your case	Retrieved case "Before"	Retrieved case "After"
Context	Comment Add a comment here	Inwestycje w tworzenie nowych serwisów i usług związanych z dotychczasową działalnością Bankier.pl. Inwestycje w przejęcia innych podmiotów.	Inwestycje w tworzenie nowych serwisów i usług związanych z dotychczasową działalnością Bankier.pl. Inwestycje w przejęcia innych podmiotów
Actions (Ansoff)	Kliknij Tu Comment Add a comment here		
Actions (Porter)	Comment Add a comment here	3b. Differentiation concentration	3b. Differentiation concentration
Actions -			

Fig. 3 The view on the STRATEGOS user interface

4. **Retain:** Finally the tested/repaired case is placed into the case base as a lesson learned for the future re-use – a learned case. The quality of learned cases is a crucial problem in the whole CBR cycle, because quality of suggested solution directly depends on this. It should be emphasized that the lessons learned might be negative as well.

As we can see, maturity and quality of STRATEGOS as a decision making tool for supporting real life decisions, does not only depend on technical issues like similarity measure, knowledge representation, etc. The main issue is to gather a valuable set of cases, and this requires long and time consuming co-operation with management boards.

V. EMPIRICAL EVALUATION

First empirical tests of STRATEGOS system were conducted based on pattern cases and real cases. Case base of the system was completed with 454 pattern cases prepared on the basis of standard knowledge in the field of strategic management. Real cases have been prepared basing on 13 IT companies listed on the Warsaw Stock Exchange. The fundamental objective of the tests was to evaluate the quality of recommendations proposed by the system by using the quality measure reflecting similarities between test (real) case and sample case. The results of this technically-oriented test were positive and were already published [15].

The target user of the system is a CEO of a SME. In this context a survey among some CEOs was conducted. They were shown how the Strategos system prototype works. The test group was selected by the target selection and it is composed of 44 CEOs from SMEs operating mainly in TMT (Technology, Media, and Telecommunication) industry. That kind of industry was selected in such a way, to present CEOs functioning in a strongly competitive and innovative market, posing significant strategic challenges. It is important to underline that 14 out of 44 CEOs have analyzed head listed companies, which in turn involves the great level of transparency of that company and the ongoing verification of CEO's activities by the market. All analyzed CEOs were male, average age over 40 years, and in average more than 11 years experience as CEO. Almost everyone admitted lack of formal education in basics of strategic management.

Table I.
List of Variables For Strategos Evaluation

Variable	Explanation
Support	Strategos is supporting me in the real strategy decision problems
Education	Strategos is teaching me in strategy management
Decision making	Strategos is generating a final solution for my strategy problems

CEOs were also asked to evaluate the Strategos system after the presentation of entire working cycle simulation, using sample case as illustration. Table I presents variables for registered CEOs evaluations. Accordingly, table II contains the summary of CEOs evaluations in three key matters.

Table II.
Results of Strategos Evaluation By CEO's

Statement	Support	Education	Decision making
Definitely no (1)	0,0%	0,0%	40,9%
No (2)	2,3%	2,3%	56,8%
I do not know (3)	9,1%	4,5%	2,3%
Yes (4)	43,2%	54,5%	0,0 %
Definitely yes (5)	45,5%	38,6%	0,0 %
Summary	**100%**	**100%**	**100%**

As we can see the results of evaluation are positive, i.e. there is almost 90% approval in the context of decisions support, as well as education. To sum up, our system was warmly approved by potential users. The results mirror real CEOs approach, where Strategos is treated as strategic decision support system, and not a system presenting what has to be done. CEOs eagerly commented on the work of the system and stressed their interest in using it in reality, if the conditions of respective quality and cases capacity would be fulfilled, what was underlined in quality assessments.

What is particularly interesting is the analysis of CEOs evaluations concerning the inferences produced by Strategos system i.e. imitation (cases from the same industry) and similarity (cases from the different industries) – vide table III. Table IV presents the evaluations of CEOs in terms of using different inferences in the system.

Table III.
List of Variables: Imitation & Similarity

Variable	Explanation
S Imitation	I find useful cases of companies from my industry in STRATEGOS case base
S Similarity	I find useful cases of companies from other industries in STRATEGOS case base

Table IV.
Results of Evaluation: Imitation & Similarity

Statement	S Imitation	S Similarity
Definitely no (1)	0,0%	0,0%
No (2)	0,0%	0,0%
I do not know (3)	6,8%	9,1%
Yes (4)	38,6%	54,5%
Definitely yes (5)	54,6%	36,4%
Summary	100%	100%

The basic question was related to the interest and will to devote time for profound analysis of companies' cases in terms of imitation and similarity. As we can see from the received answers, both aspects generated big interest. Obviously, the interest in competition's cases was the biggest, but results were also high in case of similarity. It shows a great openness of CEOs to business inspirations, some of them were particularly stressing the importance of those inspirations. The attitude of CEOs, which was full of humility towards experience of other companies, is a very positive sign in favor of acceptance of deduction paradigm used in STRATEGOS system. Taking into account received evaluations, we can state that in the studied group of CEOs, the hypothesis that referring to deduction on the basis of cases (through implementation in the described system) is an understandable and acceptable support for strategic decisions making process in management.

VI. Final Remarks

The analyzed CEOs displayed humbleness towards experiences of other companies and many times they interpreted the case base as a sort of their own memory extension. The analyzed CEOs combined acceptance of STRATEGOS with a strong feeling of limitation of reasoning appealing solely to analogies and experience. They treated the system as an inspiration or verification for their actions, being fully aware that every decision-making situation is unique and unrepeatable. It seems that this awareness will not be the same for the entire population of CEOs of SMEs. It also seems probable that CEOs of SMEs would reject STRATEGOS as a theoretical tool or to fall for the opposite tendency, namely accept its suggestions automatically. Such automatic acceptance of the system, lack of knowledge connected with superficial similarities may lead to critical decision-making errors and to huge problems for the company.

The STRATEGOS system will be extended in two directions. The main effort is focused on the proper ontology, where the company can be represented in the case base by a set of cases ordered in time (episode), so we can have the whole life-time history of the company led by the strategic decision. This is a problem of building dynamics memories called in a literature episodic-based reasoning [16]. Secondly the complete approach in strategy decision making requires deeper semantic similarities based on the object-oriented similarity [17]. The current version of the system is available on the web page: www.strategos.pl [18].

Acknowledgment

Author thanks to thank MSc students Jakub Tomiczek and Mateusz Chrzęst, who implemented the STRATEGOS prototype during work on their master's thesis.

References

[1] Gibcus P., Vermeulen P.A.M., Jong J.P.J. Strategic decision making in small firms: a taxonomy of small business owners. International Journal of Entrepreneurship and Small Business, vol. 7(1), 2009, 74-91

[2] Simon, H. A. Bounded Rationality and Organizational Learning, Organization Science 2(1), 1991, 125-134.

[3] Harris R. Introduction to decision making. http://www.virtualsalt.com/crebook5.htm, 1998

[4] Ivanova E., Gibcus P. The decision making entrepreneur. Literature overview. SCALES-paper N200219. EIM Business & Policy research, Zoetermeer, 2003

[5] Thagard, P. Mind. Introduction to Cognitive Science. Cambridge: MIT Press, 1996

[6] Gavetti, G., Levinthal, D., Rivkin, J. Strategy Making in Novel and Complex Worlds: The Power of Analogy. Strategic Management Journal, 26, 2005, 691-712.

[7] Gavetti G., Rivkin J.W. On the Origin of Strategy: Action and Cognition over Time. Organization Science, 18 (3), 2007, 420-439.

[8] Farjoun, M. Strategy Making, Novelty and Analogical Reasoning – Comentary on Gavetti, Levinthal, and Rivkin (2005), Strategic Management Journal, 29, 2008, 1001—1016.

[9] Kolodner, J. Case Based Reasonig. CA: Morgan Kaufmann, 1993

[10] Aamodt, A., Plaza, E. Case Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications, IOS Press, 7, 1994

[11] Surma, J. Case-based reasoning for supporting strategic decision making in turbulent environments for SMEs, The 29th SMS Annual International Conference, Strategies in an Uncertain World, Washington, 2009

[12] Ansoff H.I. Corporate Strategy; An Analytic Approach to Business Policy for Growth and Expansion. New York: McGraw-Hill, 1965

[13] Porter M.E. Competitive Strategy: Techniques for Analyzing Industries and Competitors. New York: Free Press, 1980

[14] Hullermeier E. Case-Based Approximate Reasoning. Berlin: Springer Verlag, 2007

[15] Surma, J. Case-Based Strategy Decision Making. Proceedings of the 17th International Conference KAM'2010. Knowledge Acquisition and Managemen. Wroclaw University of Economics, Wroclaw, 2010 (in press)

[16] Sanchez-Marre, M., Cortes, U., Martinez M., Comas, J., Rodriguez-Roda, I. An Approach for Temporal Case-Based Reasoning: Episode-Based Reasoning, ICCBR 2005, LNCS 3620, Berlin: Springer-Verlag, 2005

[17] Bergmann, R., Stahl A. Similarity Measures for Object-Oriented Case Representations. In Proceedings of the 4th EWCBR, Berlin: Springer Verlag, 1998

[18] Surma J. Case-Based Reasoning for Supporting Strategy Decision Making in Small and Medium Enterprises. "Successful Case-based Reasoning Applications", ed. L. Jain, S. Montani, Springer Verlag, Berlin, 2010

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 271–278

ISBN 978-83-60810-27-9
ISSN 1896-7094

Support of E-business by business intelligence tools and data quality improvement

Milena Tvrdíková
VŠB – TU Ostrava, Ekonomická
fakulta, Sokolská 33, 701 21
Ostrava 1, Czech Republic
Email: milena.tvrdikova@vsb.cz

Ondřej Koubek
VŠB – TU Ostrava, Ekonomická
fakulta, Sokolská 33, 701 21
Ostrava 1, Czech Republic
Email: ondrej.koubek@vsb.cz

***Abstract*—The aim of this paper is to evaluate what electronic commercial and business opportunities firms and organizations in this modern world have and which opportunities are best suited for them. The paper categorizes and describes currently available e-business tools in the electronic business model proposed by Timmers. The next part discusses the tools and activities for creating successful implementations of e-business within companies and organizations, in particular, business intelligent tools, strategic management tools, electronic forms, and competitive intelligence. The paper then offers general recommendations and a summary of which tools used for which purposes depending on the size of business and organizations.**

I. Introduction

INDUSTRIAL society is changing in front of our eyes to the society, in which the information and knowledge play the major role. There is a global information infrastructure that processes and transmits an increasing amount of data and information. The order of the Industrial Revolution was to maximize the quantity with the least cost. The goal of the information revolution is the highest possible quality at a reasonable price in the shortest possible time.

In the information society a human is searching for his place again. New technology allows him to become a passive and easily driven consumer of information, on the other hand, however, allow him to become their active producer. Convergence of information, communication and multimedia technologies, create new business opportunities in the upcoming decades which will play a key role in the economy and public life.

Worldwide, we can see the results of massive expenditure on information and communication technologies that cause the changes of business and commerce. Between 1980 and 2005 the private business investment in information technology increased from 34% to 50% of the total capital invested. How to effectively invest the money? Companies invest into information systems to achieve strategic business and marketing objectives: operational excellence, perfection of new products, services and business models, intimate knowledge of customers and suppliers, improve decision making, competitive advantage and survival. The interdependence between business and information system capabilities is growing. Changes in strategy, rules and business processes require changes in hardware (HW), software (SW), in data storages and telecommunications equipment. Often, if the company wants to do something, it depends on what the information system allows.

II. Types of e-business

Since 2000, the Economist Intelligence Unit publishes an annual ranking of the world's largest economies in readiness for e-business. Criteria for assessing the readiness for e-business were developed by the Economist Intelligence Unit in cooperation with the IBM Institute for Business Value. This is a summary of factors that indicate how the market is open to Internet opportunities. This indicator express how the country is open in terms of e-business. Evaluation methodology is undergoing continuous modification.

E-business is a way of communication and commerce using the Internet as the main instrument or other networks and is seen on a wider scale than e-commerce, much as it is itself a subset of trading business. E-business is a series of processes, pursuing a specific goal, involving more than one agency and implemented by electronic means.

E-commerce means using the most modern information communication technologies to increase the effectiveness of relationships among companies and among individual consumers. It includes not only error-free electronic transmission of information and documents, but mainly signing contracts or strategic business partnership over the Internet. E-Commerce is a series of processes associated with the course of commercial transactions carried out by electronic means

E-marketplace is a virtual, on-line marketplace, where supply and demand meet online. Its main advantage to the ordinary "stone" market is the possibility of efficient and convenient prices comparison, and the comparison of delivery and payment conditions and especially the technical parameters of each product.

The **E-procurement** term can be very loosely described as "obtaining", "providing" something over the Internet. Practically this is the part of e-commerce, based on the needs of shoppers. Its basic feature is the creation of value, hence the cost savings for the buyer. For its realization the main reason is the reduction of transaction costs. Other advantages of e-procurement are rebate, reduced inventories at the customer, better control of the process, minimizing cash transactions and integration with the information systems of customer and supplier.

E-marketing is the opposite of the e-procurement. Businessmans are selling products through a network, where the driving force in this case is the seller (supply). Its essence is to attract customers and convince them about the quality and convenience offered to purchase goods or services. E-mar-

978-83-60810-27-9/09/$25.00 © 2010 IEEE

keting is primarily focused on the value creation (increase in turnover and revenue) for the seller.

A. E-commerce models

Business-to-business (B2B)

Business-to-business means secure communication, transmission of documents, concluding business contracts and establishing long term business relationships between companies. This kind of activity is particularly associated with using the Internet to facilitate communication in the supply chain. Relationships between individual firms on the B2B market are almost always formally adjusted by the contracts.

The specific models of the B2B are for example the e-commerce models by P. Timmers [10]

- Value chain service provider, specializes in one of the functions of the value chain (such as payment or logistics) in order to obtain a competitive advantage by differentiating.
- Value chain integrator, focuses on integrating multiple steps of the value chain and on evaluation of the potential of information flow between these steps as further added value.
- 3rd party Marketplace, in its basic form it is a user interface to the catalog of products or services which may be extended with special services such as advertising, brand, payment, logistics, orders or comprehensive service to ensure the implementation of secure transactions. An example of a business relationship B2B marketing can be a single event (e.g. conference, backed by well-known and trusted company in the field) if necessary.

Business-to-consumer (B2C)

Business-to-consumer means the sales of goods to the end user via the Internet or other information and communication technologies, without direct physical contact with the customer's dealer. B2C are typical for single networking vendor - the consumer, with no long term contractual underpinning.

The typical models of B2C by P. Timmers are:

- E-shop is comparable to the catalog sales, the catalog is in this case in the form of web pages and communications between buyers and sellers take place mostly in electronic form. E-commerce must address the issue of payment and delivery of goods. From the perspective of transaction costs seem ideally the intangible goods (tickets, software, trips). Transaction costs are often an impediment to trade in goods, whose unit price is very low.
- E-mall is a set of e-commerce under one roof or under the common umbrella of one brand. It is essentially analogous to the giant shopping malls. When you specialize in a particular market segment, the commercial center is becoming a center for the entire industry.

Consumer-to-business (C2B)

C2B model means that the customer offers in the system price he is willing to pay and the seller is considering whether to accept the bid.

Among the models of electronic commerce (according to the classification of P. Timmers) that make e-business from the e-commerce, that means those models that offer additional services, are:

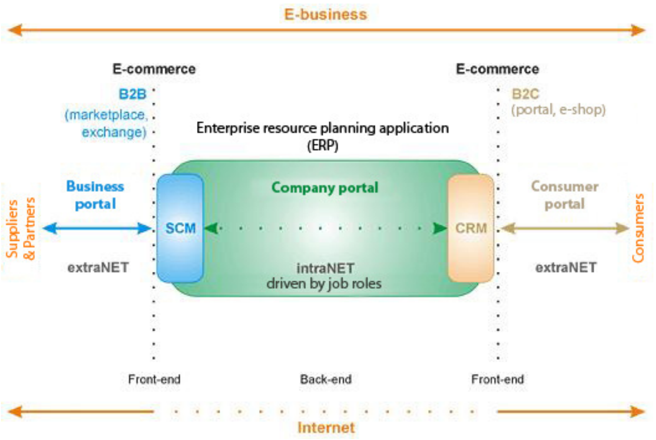
- Virtual communities, which is a fundamental value generated by members of the community (customers or partners) who add their information into the basic environment that guarantees by provider.
- Collaboration platform is a set of tools and information environment for cooperation between businesses. It may be focused on design for example. Business opportunities can be found in the management of all services (fees) and sales (licensing) of special tools.
- Information brokerage represents a wide range of new value-added services to the amount of data, which are located on open networks, or derived from integrated business operations such as drawing up a customer profile, brokering business opportunities, investment advice, etc. Specific categories of services are provided by certification authorities.

B. Implementing the concept of e-business

For the implementing of e-commerce in practice, it is necessary to address the following areas:

- Deployment of business and information portals. The portal is the gateway to information and serves the information via web browsers, allowing access from mobile devices. It represents a single and personalized access to information services and content that is targeted to the business processes of all participants - customers, employees, suppliers or partners - and with regard to the available means of communication. Through an information portal, users can create and manage content and content authors can generate and grade reports, documents, web pages, without deeper knowledge of Internet technologies.
- Support for modern forms of communication.
- Unification and management all communication channels.
- Investments in information systems security. To mitigate risk and minimize losses in case that a risk event occurs. Complement the specification of security policy elements of PKI (Public Key Infrastructure) enables the deployment of digital signature.
- Business Process Integration, Technology Integration (subject to satisfactory comprehensive e-business solution is the integration of the entire corporate information system at the user interface, data structures, data exchange and applications). There is also required integration with financial and logistical services. To benefit from the competitive and strategic benefits of e-business solutions, it is necessary to integrate fully all customers, partners and

suppliers into a single on-line network and focus on key processes. Others that are not directly related processes, they are left to specialized firms (e.g. logistics).



Note: SCM - Supply Chain Management; CRM - Customer Relationship Management

Fig. 1. E-business solutions, source [9]

All of the listed changes are linked to sufficiently significant organizational changes, and create conditions for digital business. Digital firm can be defined in different ways. For our purposes, we will apply the following definition: "Digital business is one in which nearly all organizationally significant business relationships with customers, suppliers and employees are digitally enabled and mediated, where the main business processes are implemented through a digital network embracing the whole organization or a multi-organization" [8].

Digital firms may react to their environment faster than traditional firms. They are more flexible in terms survival in a turbulent environment. The digital business is time-lag and shift in space standard. Time shift means that the transaction is a continuous 7 x 24 hours per week and not only on working days in a limited period of time. Spatial shift means that the sale takes place on the global market without national borders. Work is performed there, which is advantageously feasible in the world. E-business can expect intensified competition and interaction of traditional and new businesses, projects, policies and rules and end e-business in today's terms (every business is e-business).

III. Strategic management

Strategic or institutional management is the conduct of drafting, implementing and evaluating cross-functional decisions that will enable an organization to achieve its long-term objectives. [11] It is the process of specifying the organization's mission, vision and objectives, developing policies and plans, often in terms of projects and programs, which are designed to achieve these objectives and then allocating resources to implement the policies and plans, projects and programs. A balanced scorecard is often used to evaluate the overall performance of the business and its progress towards objectives.

Strategic management is a level of managerial activity under setting goals and over Tactics. Strategic management provides overall direction to the enterprise and is closely related to the field of Organization Studies.

A. *Strategy formulation*

Strategic formulation is a combination of three main processes which are as follows:

- Performing both internal and external as well as both micro-environmental and macro-environmental situation analysis, self-evaluation and competitor analysis.
- Concurrently with this assessment, objectives are set. These objectives should be parallel to a timeline; some are in the short-term and others in the long-term. This involves crafting vision statements (long term view of a possible future), mission statements (the role that the organization gives itself in society), overall corporate objectives (both financial and strategic), strategic business unit objectives (both financial and strategic), and tactical objectives.
- These objectives should, in the light of the situation analysis, suggest a strategic plan. The plan provides the details of how to achieve these objectives.

B. *Strategy evaluation*

Measuring the effectiveness of the organizational strategy, it is extremely important to conduct a SWOT analysis to figure out the strengths, weaknesses, opportunities and threats (both internal and external) of the entity in question. This may require to take certain precautionary measures or even to change the entire strategy.

Strategic management techniques can be viewed as bottom-up, top-down or collaborative processes. In the bottom-up approach, employees submit proposals to their managers who, in turn, funnel the best ideas further up the organization. This is often accomplished by a capital budgeting process. Proposals are assessed using financial criteria such as return on investment or cost-benefit analysis. Cost underestimation and benefit overestimation are major sources of error. The proposals that are approved form the substance of a new strategy, all of which is done without a grand strategic design or a strategic architect. The top-down approach is the most common by far. In it, the CEO, possibly with the assistance of a strategic planning team, decides on the overall direction the company should take. Some organizations are starting to experiment with collaborative strategic planning techniques that recognize the emergent nature of strategic decisions.

Strategic decisions should focus on Outcome, Time remaining, and current Value/priority. The outcome comprises both the desired ending goal and the plan designed to reach that goal. Strategic management requires paying attention to the time remaining to reach a particular level or goal and adjusting the pace and options accordingly. Value/priority relates to the shifting, relative concept of value-added. Strategic decisions should be based on the understanding that the value-added of whatever you are managing is a constantly changing reference point. An objective that begins with a high level of value-add may change due to influence of internal and external factors. Strategic management by definition is managing with a heads-up approach to outcome, time and relative value, and actively making course corrections as needed.

IV. BI tools purchase possibilities

If a company or organization decides to purchase business intelligence tools, the choice is always driven by the price, utilization and consequent size of the company. Even at the turn of the millennium, BI tools were a privilege of big companies with huge budgets only that made it worthwhile to invest considerable sums in implementing solutions to the corporate network or individual workstations. Today, this kind of acquisition of BI tools still has a lot to offer, but thanks to their robustness and costs it is unacceptable option for small businesses.

On the other hand, there is a form of Software as a Service, simply SaaS. SaaS is a quite new technique in the world of IT which allows you to hire a software application only when there is a requirement of such an utility. The main reasons for its popularity are high services, low costs and less maintenance.

In the software as a service model, the application, or service, is deployed from a centralized data centre across a network - Internet, Intranet, LAN, or VPN - providing access and use on a recurring fee basis. Users "rent," "subscribe to," “are assigned”, or "are granted access to" the applications from a central provider. Business models vary according to the level to which the software is streamlined, to lower price and increase efficiency, or value-added through customization to further improve digitized business processes. [18]

V. Improving the data quality

The main sources of information within companies and institutions are now both operating systems supporting the company (ERP, SCM, CRM) and external databases (dials address, municipality, telephone directories, business register, etc.). These resources are not usually able to provide questioner the information of desired quality. The reason is that data are stored in many places, on different platforms in different structures and formats. The problem is the quality of data. The data are often incomplete, contain errors, invalid values are stored in the structures unsuitable for analysis and do not contain history.

A. Tools for selection, transformation, transmission and data integration

Obtain quality information now means to transfer data into the relevant structures of the data warehouse. **Data warehouse** is a database containing the consolidated data from all available resources, optimized for reporting, analysis and archiving. The data warehouse integrates and stores data from both internal and external sources.

To transfer data into the data warehouse are used ETL (extraction, transformation, loading) tools. The extraction means the ability to take data from the widest range of data sources of different types. Transformation is a gradual series of operations to prepare the extracted data to be retrieved from the data warehouse. Many of the data obtained from the extraction is still not nearly ready to load into storages. Among the reasons why not, it is mainly the mismatch between data from different sources and their incompleteness. There are applied the checks, additions or changes in data transfer on the same formats and inconsistencies elimination, data consolidation - the unification of the main entities and the calculation of aggregation by major entities. To clean the data tools containing typical samples of impurities are used. Load means inserting data into its own physical space data warehouse.

Implemented ETL means primarily program implementation of data pumps, testing their time requirements and setting operating parameters. Parameters of the ETL tools are supported platforms and their connectivity (range of supported source and target systems), support for metadata management, the possibility of multiphase pump operation according to schedule, the level of support for workflow, logging level data pump, and support data processing in real time. Trend in the ETL tools is merging with the tools for managing metadata and tools for ensuring data quality, as well as their delivery, together with the standard database engine. [3]

EAI tools (Enterprise Application Integration) - EAI can be characterized as a set of approaches, methods and technologies that allow us to connect initially often incompatible solutions, or partial information systems.

The process of EAI from the perspective of the data is based on the following principles:

- elimination of semantic inconsistency of data that arises from different perspectives on the data in various applications (e.g. differences in customer address records)
- removal of content data inconsistency, which arises from the existence of duplication (e.g. two different applications registered address of the customer, but only one of them has been changed)
- minimisation the fragmentation of data (providing a comprehensive view of data)

EAI works unlike ETL tools in real time, they were created in a layer of transactional systems and their purpose is to integrate primary systems in a company or organization, and reducing their mutual interface. [5]

B. Applications for data storage

Applications for data storage provide the processes to storage, updating and data management. These include data warehouses, data marts, operational data store and data staging areas.

Data Warehouse is a central data repository, where the requirement for consistency is crucial (DW must provide "a single version of the truth"). Integrated data warehouses are presently considered to be the best solution.

Data Marts are separated data repositories for individual applications or departments. They are problem-oriented data warehouses to implement flexible ad hoc analysis.

Operational Data Store (ODS) are supporting analytical databases - a central repository, processing the key data in nearly real time. The data processing operations needed to ensure of data quality are carried out by ODS and it also integrates relevant data from different systems (examples might be the validation of address information identifying the type of identification number, or tracing and correction of incorrect data). At the operational level of the data storage are also clearly and conspicuously identified and described the various data elements; there are agreed technical and semantic definitions (metadata).

Data Staging Areas (DSA) are used for temporary storage of selected data from source systems from their own processing to other database components of BI solutions. Their purpose is to accelerate the selection of data used for initial storage of the non transformed data from these systems.

C. Data quality

Data quality is one of the basic characteristics of data warehouses and operational data stores. Data quality is not one of their automatic features, but nowadays the necessary one. Data quality can be defined in different ways, in this case we choose a simple definition - high quality data are those which correspond to reality, they are complete and consistent.

If you want to work with high-quality data, you must ensure that there are five basic characteristics:

- Completeness - the need to identify and treat the data that are missing or inapplicable,
- Standardization - all data should match the requested format,
- Consistency - no data may contain values that represent conflicting information,
- Uniqueness - if there are duplicate entries, they must be removed
- Integrity - data should include all the defined relationships to other data. [1]

Data quality is at present designed by the majority of suppliers of information technology as a set of two processes - **data profiling** and **data cleaning**. These processes are periodically repeated and reflected in the data warehouse environment. The result is a complex process in the literature called the Data Supply Chain. DSC is an automated process in which the first phase are extracted data from data sources, whether internal or external, in the second phase are analyzed the data sources and data are cleaning is performed. In this second phase the data profiling (identifying types of data defects, quantifying the number of individual defects, identifying synonyms and homonyms, the evaluating completeness in terms of supporting business processes and conformity assessment of attributes with their definitions) and data cleaning (standardization, verification against internal and external dials, correction of address data, identification of redundant records and householding identification are implemented. [2]

Thus processed data are then stored in the warehouse. Data warehouses are typically updated regularly it means that all processes of the DSC are recurrent.

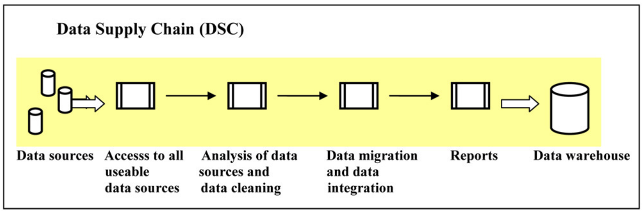

Data Supply Chain (DSC)
Data sources
Access to all useable data sources
Analysis of data sources and data cleaning
Data migration and data integration
Reports
Data warehouse

Fig.2. Data supply chain, source [author]

D. Metadata

As well as data quality, a quality description of its contents is a necessary component of any modern information system. It means how was the content created and how is the content being used - metadata. Recently the importance of metadata, as the principal means of determining the content and status of information systems, increased significantly. The underlying reason for the existence of metadata (defined as data about data) is that they add context and meaning into inaccurately described cluster of information. The main advantage of the existence of metadata is the ability to facilitate understanding of the principles, capabilities and content of the various information systems.

Metadata in the company can be divided into technical metadata - that is, information about setting up various information systems and relevant technical processes and substantive metadata - information about the substance of the solution, thus adding context and importance of individual values (and not only in terms of content meaning of that word - as understood in the organization, but also in the form of computing - how it is possible to reach a value of that expression).

Ensuring the sufficient quality of data and their description is mainly the task for their users. The main reason for the progressive transfer of responsibility for ensuring data quality and description of the importance of individual data elements is that the users who work with them in a long term know the data better than anyone else. The data are produced by information systems that are developed on the requirements of their users and the completeness, consistency, accuracy, uniqueness and integrity of data is ensured.

Metadata make sense to individual information. At the same time the tools and processes to ensure data quality, help you to capture, manage, share and provide error free and accurate data. In practice this means that by the involvement of these solutions into an integrated information system, we can obtain a simpler and better quality work of all components involved in the decision-making at all levels of management - if the person who decides on the basis of certain data understands the meaning of the data well, he or she can make the right decisions at the right time and right place. They also support the building of confidence in the data - if users understand the meaning of the data while the data is correct, then they gain confidence in these data and it will result in a streamlining of their work and increase of efficiency of the support elements (development of new solutions) - if end-users understand the state and content of the solutions, they can effectively specify their requirements for further development and this development can also be faster and more efficient. [5]

The final aim of the described tools is to provide readable, organized and analyzable in real-time, information available from a peak of corporate databases and external sources, which can be widely used at the management of a company or an institution. As the managers have ensured quality data, they gain a quick overview of the functioning of the company or institution and can devote their time to the processes leading to positive change of the situation. Quality data allow the management of an organization in accordance with knowledge.

VI. Competitive intelligence

A broad definition of competitive intelligence is the process of defining, gathering, analyzing, and distributing Intelligence about products, customers, competitors and any aspect of the environment needed to support executives and managers in making strategic decisions for an organization.

Key points of the definition above are:

- Competitive intelligence is an ethical and legal business practice, as opposed to industrial espionage which is illegal.
- The focus is on the external business environment. [14]
- There is a process involved in gathering information, converting it into intelligence and then utilizing this in business decision making. CI professionals emphasize that if the intelligence gathered is not usable (or actionable) then it is not intelligence.

A more focused definition of CI regards it as the organizational function responsible for the early identification of risks and opportunities in the market before they become obvious. Experts also call this process the early signal analysis. This definition focuses attention on the difference between dissemination of widely available factual information (such as market statistics, financial reports, newspaper clippings) performed by functions such as libraries and information centers, and competitive intelligence which is a perspective on developments and events aimed at yielding a competitive edge [15].

The term CI is often viewed as synonymous with competitor analysis, but competitive intelligence is more than analyzing competitors — it is about making the organization more competitive relative to its entire environment and stakeholders: customers, competitors, distributors, technologies, macro-economic data etc.

Organizations use competitive intelligence to compare themselves to other organizations ("competitive benchmarking"), to identify risks and opportunities in their markets, and to pressure-test their plans against market response (war gaming), which enable them to make informed decisions. Most firms today realize the importance of knowing what their competitors are doing and how the industry is changing, and the information gathered allows organizations to realize their strengths and weaknesses.

With the right amount of information, organizations can avoid unpleasant surprises by anticipating competitors' moves and decreasing time response. Major airlines change hundreds of fares daily in response to competitors' tactics. They use information to plan their own marketing, pricing, and production strategies.

Resources, such as the Internet, have made gathering information on competitors easy. With a click of a button, analysts can discover future trends and market requirements. However, competitive intelligence is much more than this, as the ultimate aim is to lead to competitive advantage. As the Internet is mostly public domain material, information gathered is less likely to result in insights that will be unique to the company. In fact there is a considerable risk that information gathered from the Internet will be a misinformation and will mislead users.

As a result, although the Internet is viewed as a key source, most CI professionals should spend their time and budget gathering intelligence using primary research — networking with industry experts, from trade shows and conferences, from their own customers and suppliers, and so on. Where the Internet is used, it is to gather sources for primary research as well as information on what the company says about itself and its online presence (in the form of links to other companies, its strategy regarding search engines and online advertising, mentions in discussion forums and on blogs, etc.). Online are subscription databases and news aggregation sources which have simplified the secondary source collection process are also important

Organizations must be careful not to spend too much time and effort on old competitors without realizing the existence of any new competitors. Knowing more about your competitors will allow your business to grow and succeed. The practice of competitive intelligence is growing every year, and most companies and business students now realize the importance of knowing their competitors.

VII. Conclusion

The goal of digital tools implementation into business processes is to get a good position in developing e-market as well as to offer better services for customers. Main attention is focused on the problems that have to be solved in connection with the implementation of e-business in companies and institutions.

Use of the business intelligence tools, along with the emphasis on the quality of strategic management in a company or institution and the high quality used and data sources relevant to the issue (competitive intelligence), creates conditions for the choice of form and development of the e-business within the company. Other critical attribute is the size of the company and especially its global strategy.

In Table I, you can see the opportunities and recommendations for various companies (according to the number of employees). One of the authors of this article is a chairman of the regional section of the Czech Society for Systems Integration for North Moravia and Silesia (www.cssi-morava.cz) and maintains frequent contacts with representatives of management of many companies. The results shown in the table were made on a series of guided interviews with these managers.

Current indications suggest that a firm that does not pay attention to the possibilities of e-business offers, might become unattractive to their customers, therefore, managers should set out the recommendations given due attention.

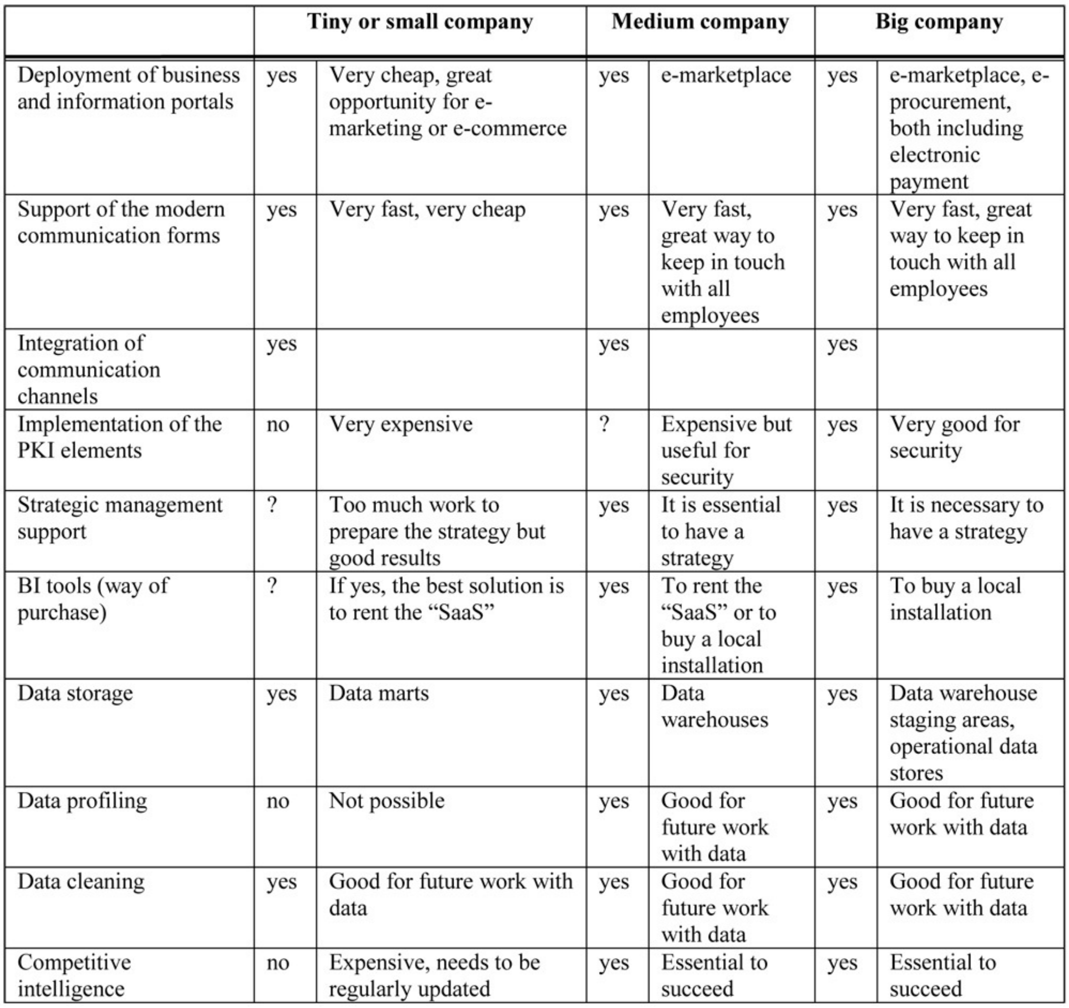
In recent years the use of the new information technologies leads to a significant increase in market places (globalization) and convergence fields. Innovation becomes a key factor in the development of companies and the role of the client (active customer) is being significantly changed. Due to information and communication technologies the network character of all industrial sectors is growing and ITC also changes the structure of costs in almost all sectors.

Economic laws of "a world of bits" are being gradually penetrated into the" world of atoms" and informatics does fundamental changes in almost all sectors of the economy [6]. By our team work intensively with three business intelligence tools of different formats. Unfortunately we have no detailed information directly from the private sector, because firms guard their sensitive data and don't let us look under the cover, but on the basis of many simulations we have obtained enough information on the findings above.

ACKNOWLEDGEMENT

This work was supported by project No.SGS SP/2010167. The authors would like to thank the reviewers for their extremely useful comments and suggestions that substantially improved the quality of the final version.

TABLE I.
OPPORTUNITIES FOR DIFFERENT COMPANIES AND ORGANIZATIONS, SOURCE [AUTHOR]



	Tiny or small company		Medium company		Big company	
Deployment of business and information portals	yes	Very cheap, great opportunity for e-marketing or e-commerce	yes	e-marketplace	yes	e-marketplace, e-procurement, both including electronic payment
Support of the modern communication forms	yes	Very fast, very cheap	yes	Very fast, great way to keep in touch with all employees	yes	Very fast, great way to keep in touch with all employees
Integration of communication channels	yes		yes		yes	
Implementation of the PKI elements	no	Very expensive	?	Expensive but useful for security	yes	Very good for security
Strategic management support	?	Too much work to prepare the strategy but good results	yes	It is essential to have a strategy	yes	It is necessary to have a strategy
BI tools (way of purchase)	?	If yes, the best solution is to rent the "SaaS"	yes	To rent the "SaaS" or to buy a local installation	yes	To buy a local installation
Data storage	yes	Data marts	yes	Data warehouses	yes	Data warehouse staging areas, operational data stores
Data profiling	no	Not possible	yes	Good for future work with data	yes	Good for future work with data
Data cleaning	yes	Good for future work with data	yes	Good for future work with data	yes	Good for future work with data
Competitive intelligence	no	Expensive, needs to be regularly updated	yes	Essential to succeed	yes	Essential to succeed

References

[1] Tvrdíková, M. (2007) Business Intelligence tool and their support first-rate data, In Information and Communication Technology for Practice, VŠB-TU Ostrava

[2] Ciarciello, R. (2006) Customer Data: Front and Center, DM Review

[3] Panec, Z. (2002) Jak poznat naše zákazníky dříve než konkurence, IT Systém, vol.4. (11), Brno

[4] Tvrdíková, M. (2004) Nástroje Business Intelligence, In Tvorba softwaru 2004, VŠB-TU Ostrava

[5] Zornes, A. (2006) Taking Customer Data Integration , Master Data Management Milestones, DM Review

[6] Negroponte, N.P. (1995) Being Digital, New York

[7] Knoblochová, C., Engelhardt, P., Faťun, M. (2001) E-tržiště mění výrazně tvář byznysu, Computerworld, Praha

[8] Laudon, K.C., Laudon J.P. (2006) Management Information Systems – Managing the Digital Firm, Prentice Hall, New Jersey

[9] Sculley, A.B., Woods, W.W.A. (2001) B2B internetová tržiště, Grada Publishing, Praha

[10] Timmers, P. (1999) Electronic Commerce: Strategies and Models for Business-to-Business Trading. John Wiley, Chichester

[11] David, F.(1989) Strategic Management, Columbus Merrill Publishing Company

[12] Tvrdíková, M. (2008) Aplikace moderních informačních technologií v řízení firmy- nástroje ke zvyšování kvality IS, GRADA Publishing, Praha

[13] Lamb, R.B., (1984) Competitive strategic management, Englewood Cliffs, Prentice-Hall

[14] Haag,S. (2007) Management Information Systems for the Information Age, Irwin/McGraw-Hill, Ryerson

[15] Gilad, B. (2008) "The Future of Competitive Intelligence: Contest for the Profession's Soul", Competitive Intelligence Magazine, 11(5), 22

[16] http://www.slideshare.net/Deltl/competitive-intelligence-english-presentation

[17] http://businessworld.cz/pruzkumy-a-analyzy, 14.04.2006

[18] http://www.siia.net/estore/ssb-01.pdf, 2001

Computer Aspects of Numerical Algorithms

Numerical 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

Programm Committee

Nail Akar, Bilkent University, USA

Pierluigi Amodio, Universita' di Bari, Italy

Zacharias Anastassi, Technological Educational Institute of Kalamata, Greece

Krzysztof Banaś, AGH, Poland

Michele Benzi, Emory University, USA

Dario Andrea Bini, University of Pisa, Italy

Luigi Brugnnano, Universita degli Studi, Firenze, Italy

Beata Bylina, Maria Curie-Sklodowska University, Poland

Jarosław Bylina, Maria Curie-Sklodowska University, Poland

YingShi Chen, grusoft, China

Tadeusz Czachórski, IITiS PAN, Poland

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

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

David R. Kincaid, University of Texas at Austin, USA

Steve Kirkland, National University of Ireland Maynooth, Ireland

Jerzy Klamka, Silesian University of Technology, Poland

William Knottenbelt, Imperial College London, United Kingdom

Stanisław Kozielski, Silesian University of Technology Institute of Informatics, Poland

Gertrud Kraut, USA

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

Beatrice Meini, University of Pisa, Italy

Peter Minev, University of Alberta, Canada

Yvan Notay, Universite Libre de Bruxelles, Belgium

Małgorzata Peszynska, Oregon State University, USA

Dana Petcu, West University of Timisoara, Romania

Svetozara Petrova, Germany

Eric Polizzi, University of Massachusetts, Amherst, USA

Ivana Pultarová, Czech Technical University in Prague, Czech Republic

Bianca-Renata SATCO, "Stefan cel Mare" Universityof Suceava, Romania

William E. Schiesser, Lehigh University , USA

Vladimir V. Sergeichuk, Institute of Mathematics, National Academy of Sciences, Ukraine

Dr. Tom E. Simos, University of Peloponnese, Greece

Natesan Srinivasan, Indian Institute of Technology, India

Przemyslaw Stpiczynski, Maria Curie-Sklodowska University, Poland

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-Sklodowska University, Poland

Marian Vajtersic, University of Salzburg, Austria

Luben Vulkov, Rousse State University , Bulgaria

Verena Wolf, Saarland University, Germany

Zlatev Zahari, National Environmental Research Institute, Aarhus University, Denmark

Organizing Committee

Beata Bylina, Maria Curie-Sklodowska University, Poland

Jarosław Bylina, Maria Curie-Sklodowska University, Poland

Przemyslaw Stpiczynski (Chairman), Maria Curie-Sklodowska University, Poland

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 281–288

ISBN 978-83-60810-27-9
ISSN 1896-7094

The experimental analysis of GMRES convergence for solution of Markov chains

Beata Bylina
Institute of Mathematics
Marie Curie-Sklodowska University
plac Marii Curie-Skłodowskiej 5, 20-031 Lublin, Poland
Email: beatas@hektor.umcs.lublin.pl

Jarosław Bylina
Institute of Mathematics
Marie Curie-Sklodowska University
plac Marii Curie-Skłodowskiej 5, 20-031 Lublin, Poland
Email: jmbylina@hektor.umcs.lublin.pl

***Abstract*—The authors consider the impact of the structure of the matrix on the convergence behavior for the GMRES projection method for solving large sparse linear equation systems resulting from Markov chains modeling. Studying experimental results we investigate the number of steps and the rate of convergence of GMRES method and the IWZ preconditioning for the GMRES method. The motivation is to better understand the convergence characteristics of Krylov subspace method and the relationship between the Markov model, the nonzero structure of the coefficient matrix associated with this model and the convergence of the preconditioned GMRES method.**

I. Introduction and Motivation

MARKOV chains are a particularly robust and wide used tool for analyzing a variety of stochastic (probabilistic) systems over time.

A CTMC (Continuous-Time Markov Chain) may be represented by a set of states and a transition rate matrix $\mathbf{Q}$ containing state transition rates as coefficients. To compute the steady-state probabilities we must solve a (homogeneous) sparse system of linear equations, of the form $\mathbf{Q}^{\mathbf{T}}\mathbf{x} = \mathbf{0}$, of size equal to the number of states in the CTMC. $\mathbf{Q}$ is a singular matrix demanding adequate methods to solve the equation. Solving the equation system generally requires applying iterative methods, projection methods or decomposition methods but occasionally (for the need of an accurate solution) direct methods are used as well. The rich material concerning the methods mentioned above can be found in [13].

In this article we consider one of the Krylov subspace methods, namely the GMRES method. This method was first introduced by Y. Saad in the article [12] as the method to solve linear systems of equations. GMRES for Markov chains was studied in the article [11]. The full GMRES algorithm is guaranteed to converge in at most n steps, but it is not useful for large systems of equations, because a good approximate solution is often computed quite early, after very few iterations. In the literature, we find results, which would provide an upper bound on the convergence rate of the GMRES [9]. The traditional bounds of the residual are expressed in terms of eigenvalues of $\mathbf{Q}$ and the condition number of the eigenvector matrix. It is of limited practical interest because we need the condition number, which is typically not known. For any matrix determination of its condition number is a task of the complexity $O(n^3)$. In practice, it is difficult to use the theoretical knowledge about the convergence of the GMRES method.

One of the tools used in the convergence analysis of GMRES are numerical experiments. We perform numerical experiments to help us understand the effect of nonzero structure of the matrix on the convergence characteristics of preconditioned Krylov subspace methods. We try to provide some properties of the coefficients of the matrix $\mathbf{Q}$, which affect the convergence of the method GMRES and the preconditioned GMRES.

One of the famous preconditioning techniques is incomplete factorization, for example IWZ factorization. The incomplete WZ factorization is originally described in our previous works [3], here we discuss its performance for Krylov subspace methods like GMRES.

Basing on our previous investigation we consider impact of the incomplete WZ factorization on the GMRES method for the numerical solution of Markov chains. We study relationship between the number of iterations, the convergence rate of the GMRES method and properties of the matrix $\mathbf{Q}$. Research was carried out for two models. The first model concern matrices associated with some abstract model. These matrices have not got any structure. The second model concern matrices known from the literature as the epidemic model and this matrix has got a structure.

The rest of the paper is organized as follows. Section II presents the problem. In Section III Krylov subspace are reminded. Section IV recall briefly the IWZ preconditioning. Section V presents two test models. Section VI describes conducted numerical experiments. Section VII contains some conclusions.

II. CTMCs and the Steady-State Solution

While modeling with Markov chains, in a steady state (independent of time), we obtain a linear equation system like following;

$$\mathbf{Q}^T\mathbf{x} = \mathbf{0}, \qquad \mathbf{x} \geq \mathbf{0}, \qquad \mathbf{x}^T\mathbf{e} = 1 \qquad (1)$$

where $\mathbf{Q}$ is a transition rate matrix, $\mathbf{x}$ is an unknown vector of states probabilities and $\mathbf{e} = (1, 1, ..., 1)^T$. The matrix $\mathbf{Q}$ is a square one of size $n \times n$, usually a big one, of rank $n - 1$, sparse, with dominant diagonal.

```
1) choose:
    • an initial approximation $\mathbf{x}^{(0)}$
    • a subspace $\mathcal{K}$ spanned by $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$
    • a subspace $\mathcal{L}$ spanned by $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_m]$
2) $\mathbf{r}^{(0)} \leftarrow -\mathbf{Q}^T\mathbf{x}^{(0)}$
3) $\mathbf{y} \leftarrow (\mathbf{W}^T\mathbf{Q^T}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{r}^{(0)}$
4) $\mathbf{x}^{(0)} \leftarrow \mathbf{x}^{(0)} + \mathbf{V}\mathbf{y}$
```

Fig. 1. A basic projection step for the equation $\mathbf{Q}^T\mathbf{x} = \mathbf{0}$

III. Krylov subspace methods — GMRES

In this section we recap the basics of projection methods.

The projection methods consist in approximating the solution vector with a vector from a small-dimension subspace. Such approximations are repeated until our approximation is sufficiently close to the solution — in some sense the projection methods are iterative methods.

The projection methods need more space than iterative methods (because they have to store huge basis vectors of subspaces), but can converge faster than classical iterative methods — although the convergence rate is much better for the matrices 'more beautiful' in their structure than the ones arising in solving Markov chains.

A. The Projection Step

To solve a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ by a projection method first we have to choose two subspaces of dimension m from the n-dimensional space:

- $\mathcal{K}$ which is a subspace containing the approximation;
- $\mathcal{L}$ which is a subspace defining constraints for selection of approximation from $\mathcal{K}$.

Let the subspace $\mathcal{K}$ be spanned by $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_m)$. The approximated solution is in $\mathcal{K}$ so it can be written

$$\mathbf{x} = \mathbf{V}\mathbf{y}$$

where $\mathbf{y}$ is an m-dimensional unknown vector. To find $\mathbf{y}$ we require that the residual vector $\mathbf{b} - \mathbf{A}\mathbf{x} = \mathbf{b} - \mathbf{A}\mathbf{V}\mathbf{y}$ be orthogonal to the subspace $\mathcal{L}$ spanned by $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_m)$, that is:

$$\mathbf{W}^T(\mathbf{b} - \mathbf{A}\mathbf{V}\mathbf{y}) = 0,$$

and then (if the matrix $\mathbf{W}^T\mathbf{A}\mathbf{V}$ is nonsingular):

$$\mathbf{y} = (\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{b}.$$

If we know an initial approximation $\mathbf{x}^{(0)}$ we will rather seek a difference $\mathbf{d}$ between the exact solution $\mathbf{x}$ and $\mathbf{x}^{(0)}$: $\mathbf{x} = \mathbf{x}^{(0)} + \mathbf{d}$. Setting $\mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(0)}$ we are to solve the equation

$$\mathbf{A}\mathbf{d} = \mathbf{r}^{(0)},$$

what can be done with the described above projection step.

A basic projection step for our equation 1 (where $\mathbf{A} = \mathbf{Q}^T$ and $\mathbf{b} = \mathbf{0}$) is shown in Figure 1.

```
1) $\mathbf{v}_1 \leftarrow \mathbf{v}/||\mathbf{v}||_2$
2) for $j = 1, 2, \ldots, m$:
    a) $\mathbf{w} \leftarrow \mathbf{A}\mathbf{v}_j$
    b) for $i = 1, 2, \ldots, j$:
        i) $h_{ij} \leftarrow \mathbf{v}_i^T\mathbf{w}$
        ii) $\mathbf{w} \leftarrow \mathbf{w} - h_{ij}\mathbf{v}_i$
    c) $h_{j+1,j} \leftarrow ||\mathbf{w}||_2$
    d) $\mathbf{v}_{j+1} \leftarrow \mathbf{w}/h_{j+1,j}$
```

Fig. 2. The basic Arnoldi process for a subspace $\mathcal{K}_m(\mathbf{A}, \mathbf{v})$

The most efficient method for general, non-symmetric coefficient matrices (like $\mathbf{Q}^T$) are methods based on Krylov subspaces. A Krylov subspace is defined by its dimension m, a matrix $\mathbf{A}$ and a vector $\mathbf{v}$:

$$\mathcal{K}_m(\mathbf{A}, \mathbf{v}) = \operatorname{span}\{\mathbf{v}, \mathbf{A}\mathbf{v}, \mathbf{A}^2\mathbf{v}, \ldots, \mathbf{A}^{m-1}\mathbf{v}\}.$$

Many of such methods require that an orthonormal basis be found for the Krylov subspace. Unfortunately, classical Gram-Schmidt orthogonalization is numerically poor. To deal with it there are two main kinds of methods: Arnoldi process (which is a modified Gram-Schmidt orthogonalization) and Lanczos methods (originally for symmetric coefficient matrices but generalized in some ways).

B. The Arnoldi Process

The Arnoldi process [1] on its own (see Figure 2) generates the orthonormal basis $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_m)$ for the subspace $\mathcal{K}_m(\mathbf{A}, \mathbf{v})$ and an upper Hessenberg matrix $\mathbf{H} = (h_{ij})$:

$$\mathbf{H} = \begin{pmatrix} h_{11} & h_{12} & h_{13} & \cdots & h_{1,m-1} & h_{1m} \\ h_{21} & h_{22} & h_{23} & \cdots & h_{2,m-1} & h_{2m} \\ 0 & h_{32} & h_{33} & \cdots & h_{3,m-1} & h_{3m} \\ 0 & 0 & h_{43} & \cdots & h_{4,m-1} & h_{4m} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & h_{m,m-1} & h_{mm} \end{pmatrix},$$

which represents the linear transformation $\mathbf{A}$ restricted to $\mathcal{K}_m(\mathbf{A}, \mathbf{v})$ with respect to the basis $\mathbf{V}$, that is $\mathbf{H} = \mathbf{V}^T\mathbf{A}\mathbf{V}$.

The original Arnoldi process applied to a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ is called *the full orthogonalization method* (FOM) [10] but a better approach is *the generalized minimum residual* algorithm (GMRES) [12]. Both the methods are shown in Figure 3. They differ only in one step — how to find the vector $\mathbf{y}$ (but both the procedures are projections [12]).

The GMRES algorithm is very popular in its iterative form. In the iterative GMRES after computing the new vector $\mathbf{x}^{(0)}$, the new residual $-\mathbf{Q}^T\mathbf{x}^{(0)}$ is checked if it is sufficiently close to $\mathbf{0}$. If not, the whole algorithm is repeated with the new $\mathbf{x}^{(0)}$ as the initial guess.

One of the advantages of this method is no fill-in generation (because the matrix $\mathbf{Q}$ is only used in the matrix-vector multiplication), the other is the fast convergence rate. The

```
1) choose $\mathbf{x}^{(0)}$ and $m$
2) $\mathbf{r}^{(0)} \leftarrow -\mathbf{Q}^T\mathbf{x}^{(0)}$
3) $\beta \leftarrow ||\mathbf{r}^{(0)}||_2$
4) $\mathbf{v}_1 \leftarrow \mathbf{r}^{(0)}/\beta$
5) for $j = 1, \ldots, m$:
    a) $\mathbf{w} \leftarrow \mathbf{Q}^T\mathbf{v}_j$
    b) for $i = 1, \ldots, j$:
        i) $h_{ij} \leftarrow \mathbf{v}_i^T\mathbf{w}$
        ii) $\mathbf{w} \leftarrow \mathbf{w} - h_{ij}\mathbf{v}_i$
    c) $h_{j+1,j} \leftarrow ||\mathbf{w}||_2$
    d) $\mathbf{v}_{j+1} \leftarrow \mathbf{w}/h_{j+1,j}$
6) FOM only:
   find $\mathbf{y} = (y_1, \ldots, y_m)$ from the $m \times m$ Hessenberg
   system $\mathbf{Hy} = \beta\mathbf{e}_1$
7) GMRES only:
   find $\mathbf{y} = (y_1, \ldots, y_m)$ minimizing $||\beta\mathbf{e}_1 - \bar{\mathbf{H}}\mathbf{y}||_2$
   where $\bar{\mathbf{H}} = (h_{ij})$ is an $(m+1) \times m$ upper
   Hessenberg matrix
8) $\mathbf{x}^{(0)} \leftarrow \mathbf{x}^{(0)} + \sum_{i=1}^{m} \mathbf{v}_i y_i$
```

Fig. 3. The FOM and GMRES methods for the equation $\mathbf{Q}^T\mathbf{x} = \mathbf{0}$

iterative GMRES algorithm is also convenient to vectorize [5] and parallelize [4]).

One of the problems which materialize when using the GMRES method is to select the optimal parameter m. The lower the value of the parameter m, the shorter loop and thus the less calculation time and space. Additionally, the vector $\mathbf{v}$ is also shorter.

IV. IWZ Preconditioning

The convergence rate of iterative methods depends on properties of the coefficient matrix of the linear system. If 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 (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:

$$\mathbf{M}^{-1}\mathbf{Q}^T\mathbf{x} = \mathbf{0}, \qquad \sum_{i=1}^{n} x_i = 1, \qquad \mathbf{x} \geq \mathbf{0}, \qquad (2)$$

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 computational less time) than the original problem.

There is a clear conflict among these three requirements, especially for the construction of general purpose preconditioners.

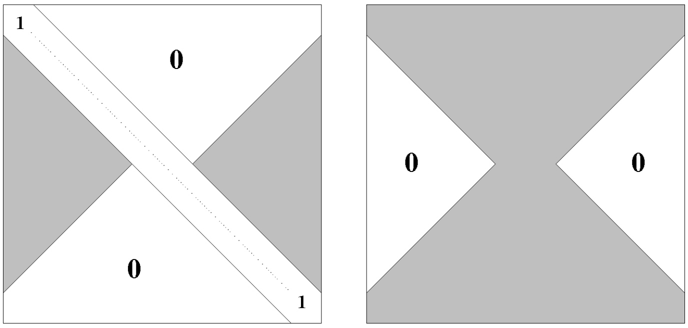


Fig. 4. The form of the output matrices in the WZ factorization (left: $\mathbf{W}$; right: $\mathbf{Z}$)

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 reduced number of iterations needed to acquire required accuracy — or by using the same preconditioner for various linear systems.

The preconditioner matrix is usually built on the basis of the original coefficients of the matrix $\mathbf{Q}$. In [2] preconditioners for Krylov subspace methods for solving large singular linear systems arising from Markov modeling are considered. The incomplete WZ factorization is originally described in our previous works [3]; here we only recall it. The WZ factorization consists in decomposition of the given matrix ($\mathbf{Q}^T$ in the paper) into a product of two matrices: $\mathbf{W}$ and $\mathbf{Z}$ (Figure 4).

Incomplete WZ factorization (denoted IWZ) is based on the described above WZ factorization, where we find matrices $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{Z}}$ (of the form of matrices $\mathbf{W}$ and $\mathbf{Z}$ shown in Figure 4) and the product $\widetilde{\mathbf{W}}\widetilde{\mathbf{Z}}$ is a kind of approximation for the matrix $\mathbf{Q}^T$.

In IWZ computations are conducted as in complete WZ factorization, but new non-zero 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 $\mathbf{Q}^T$. Hence, the factors together have the same number of non-zeros as the original matrix $\mathbf{Q}^T$. It is worth noting that we got the inverse of $\widetilde{\mathbf{W}}$, because [14]:

$$\widetilde{\mathbf{W}}^{-1} = (-1) \cdot (\widetilde{\mathbf{W}} - \mathbf{I}) + \mathbf{I} \qquad (3)$$

$$(\text{just like} \quad \mathbf{W}^{-1} = (-1) \cdot (\mathbf{W} - \mathbf{I}) + \mathbf{I}). \qquad (4)$$

After IWZ we have:

$$\mathbf{Q}^T = \widetilde{\mathbf{W}}\widetilde{\mathbf{Z}} + \mathbf{R}_{WZ}, \qquad (5)$$

where $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{Z}}$ are (respectively) matrices of the form of $\mathbf{W}$ and $\mathbf{Z}$ from Figure 4 and the remainder matrix $\mathbf{R}_{WZ}$ is supposed to be small in a sense.

V. The test models

Two models are chosen to test: an abstract model (Model I) and a model of epidemics known from literature (Model II). Parameter d was introduced for the characterization of the matrixes. So, d is an average number of non-zeros in a row/column of the matrix.

TABLE I
ESSENTIAL CHARACTERISTICS OF THE MATRICES USED IN TESTS

Group	Matrix ID	n	nz	d
I	1	100	1190	11.9
II	2	100	388	3.88
I	3	1500	37955	25.3
II	4	1500	5873	3.9
I	5	3000	120590	40.2
II	6	3000	11636	3.9

TABLE II
THE TEST MATRIX ATTRIBUTES FOR A 2D MARKOV MODEL.

ID	7
N_x	64
N_y	16
n	1105
nz	9457
d	8.56

A. Model I

In this section we describe matrices corresponding to the model I. Matrices (with IDs from 1 to 6) used in tests were generated by the paper authors on the basis of some abstract queuing models — the matrices are infinitesimal generators of Markov chains describing these models — and they are neither symmetric nor anyway structural. In Table I the essential characteristics of the matrices are presented (n is the number of rows/columns of the matrix, nz is the number of non-zeros 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 the parameter d. The matrices were divided into two groups, the first group include matrices with $d > 8$, the second group include matrices with $d \leq 8$.

B. Model II

The matrix from model II was generated from a standard two-dimensional model [6], [7]. Table II shows the test matrix attributes for a 2D Markov model. This particular example has been taken from [8], [7]. The model is discussed there and it has been used to compare different solution methods in [13].

The state of the chain is described as a two-dimensional vector. In the first dimension, the state variable assumes all values from 0 through N_x; in the second dimension the state variable takes on values from 0 through N_y. The states of such a chain are described with two numbers (u, v), $u = 0, \ldots, N_x$, $y = 0, \ldots, N_y$ (here $N_x = 64$, $N_y = 16$) and transitions are only allowed from (u, v) to (u', v') if $|u' - u| \leq 1$ and $|v' - v| \leq 1$. There was assumed — as in [6] — that only some transition from each state are permitted. This two-dimensional Markov chain model allows for transitions from any non-boundary state to adjacent states in fixed directions (chosen from North, South, East, West, North-East, North-West, South-East, South-West). A sample scheme of the model (with allowed directions: South, East and North-West) is shown in Figure 5.

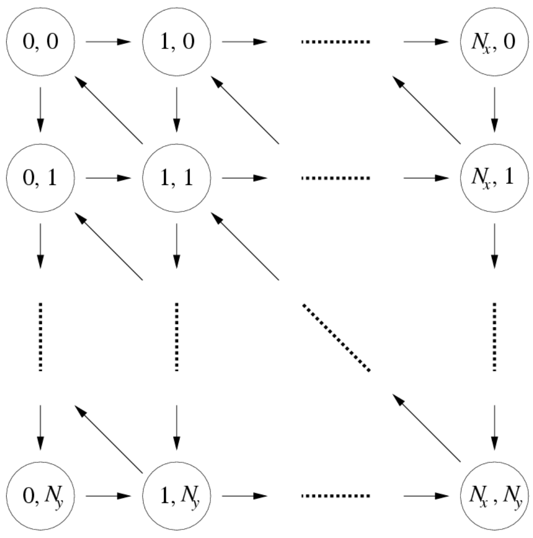


Fig. 5. A sample scheme of a two-dimensional Markov chain.

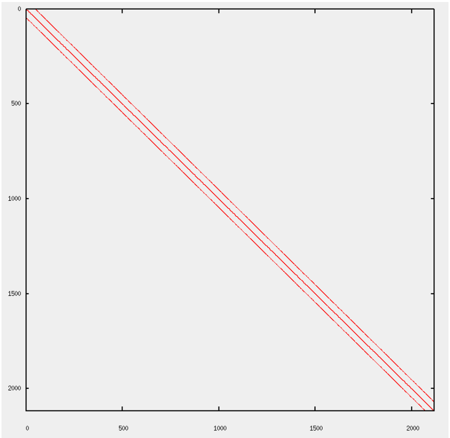


Fig. 6. The structure for the model II matrix

The matrix describing the two-dimensional Markov chain has a structure shown in Figure 6.

VI. EXPERIMENTAL RESULTS

The experiment was performed on a Pentium IV 2.8GHz computer, 1GB RAM, with Debian GNU/Linux operating system. We used high-level programming language, namely Octave.

A vector $\mathbf{x}^{(0)} = (x_i^{(0)})$ with $x_i^{(0)} = \frac{1}{i}$ was chosen as an initial vector. (We chose $x_i^{(0)} = \frac{1}{i}$ because the starting vector can be selected almost freely, but its elements should not be equal — $x_i^{(0)} = \frac{1}{i}$ fulfils this condition.) As a measure of accuracy of the solution we chose:

$$\varepsilon^{(k)} = ||\mathbf{0} - \mathbf{Q}^T \mathbf{x}^{(k)}||_2. \quad (6)$$

Accuracy has been studied experimentally for the matrix of model I and model II. We study both the number of iterations needed to achieve a given accuracy, and the rate of convergence. The stopping criterion used is that the 2-norm of the residual $||\mathbf{Q}^T \mathbf{x}^k||_2$ is less than e^{-15}

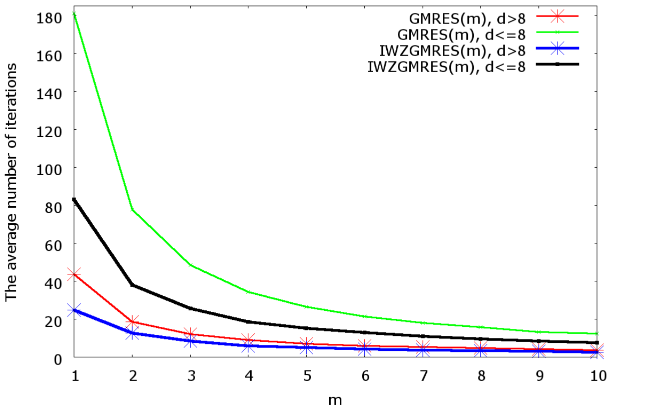


Fig. 7. Relationship between the parameter m and the average number of iterations needed to achieve accuracy e^{-16} for GMRES method and IWZGMRES method, for matrices of Group I ($d > 8$) and II ($d <= 8$).

A. Number of iterations

Table III shows numbers of iteration used to achieve a given accuracy for selected parameters m for two methods: GMRES(m) alone (denoted GMRES(m)) and GMRES(m) preconditioned with IWZ (denoted IWZGMRES(m)).

Figure 7 shows the relationship between the parameter m and the average number of iterations needed to achieve a given accuracy. The average number of iterations is counted for two groups of matrices. First for the group with $d > 8$ (matrix number 1, 3, 5) and the second group of matrices with $d \leq 8$ (matrix number 2, 4, 6).

Applications that may be present at the table III and Figure 7 are as follows:

- With the increase of parameter m the average number of iterations needed to achieve the assumed accuracy of the method GMRES(m) and IWZGMRES(m) decreases inversely.
- For each matrix from the first group number of iterations needed to achieve the assumed precision for the selected value of parameter m is almost the same. Analogous relationship can be seen for matrices of the second group. It means that regardless of the size of the matrix the number of iterations needed to achieve a given accuracy is the same and depends on the parameter d.
- In the method IWZGMRES(m) the number of iterations needed to achieve a given convergence is less than the GMRES(m) method, regardless of the parameter m and the parameter d.

Let us define the coefficient $p(m)$, which shows the relationship between the number of iterations needed to achieve (given the convergence e^{-16}) in the method of GMRES(m) and the method IWZGMRES(m).

Let $I_{IWZGMRES(m)}(m)$ mean the number of iterations needed to achieve the fixed accuracy of the method IWZGMRES(m) depending on the parameter m

Let $I_{GMRES(m)}(m)$ mean the number of iterations needed to achieve the fixed accuracy of the method GMRES(m) depending on the parameter m.

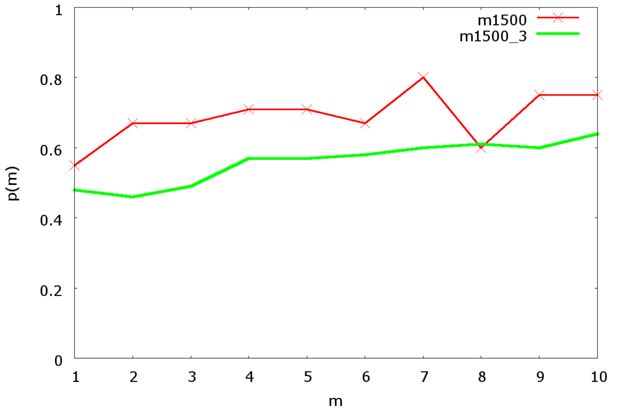


Fig. 8. Relationship between the parameter m and the value of the coefficient $p(m)$ for matrices 3 (m1500) and 4 (m1500_3)

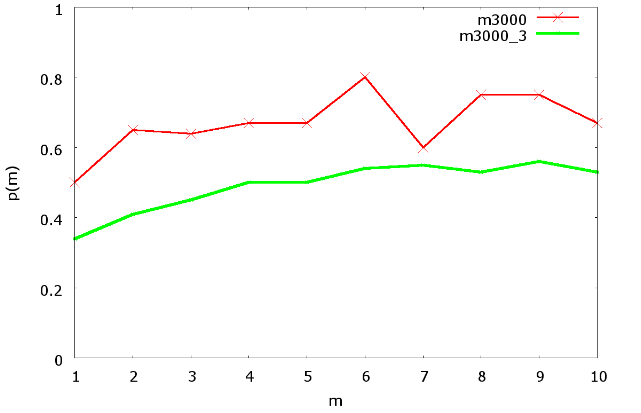


Fig. 9. Relationship between the parameter m and the value of the coefficient $p(m)$ for matrices 5(m3000) and 6 (m3000_3)

Let

$$p(m) = \frac{I_{IWZGMRES(m)}(m)}{I_{GMRES(m)}(m)}.$$

Figure 8 shows the relationship between the parameter m and the value of the coefficient $p(m)$ for the matrices 3 and 4 in Table II. Matrices have size 1500 and vary in the value of the parameter d. Figure 9 shows the relationship between the parameter m and the value of the coefficient $p(m)$ for the matrices with the numbers 5 and 6 in Table II. Matrices have size 3000 and vary in the value of the parameter d.

Figures 8 and 9 show how the value of the parameter m influence the convergence. The conclusions are:

- With the increase of parameter m (where m changes from 1 to 10) the value of the coefficient of $p(m)$ grows.
- Value of the parameter $p(m)$ for the matrices of group I is higher than for the matrices in group II.
- For the matrices of the group II growth factor $p(m)$ is more uniform than for the matrix of the group I.

Let

$$p = \max_{1<=m<=10} |p(m)|.$$

TABLE III
NUMBER OF ITERATION k NEED TO ACHIEVE A GIVEN ACCURACY $\varepsilon^{(k)} = e^{-16}$ FOR THE SELECTED OF THE VALUE PARAMETER m.

	$m = 1$		$m = 5$		$m = 10$	
ID	GMRES(m)	IWZGMRES(m)	GMRES(m)	IWZGMRES(m)	GMRES(m)	IWZGMRES(m)
1	45	29	8	6	4	3
2	87	69	17	13	8	7
3	44	24	7	5	4	3
4	185	88	30	17	14	9
5	42	21	6	4	3	2
6	271	92	32	16	15	8

TABLE IV
VALUE OF THE PARAMETER p FOR MATRICES OF THE MODEL I

ID	*p*
1	0.8
2	0.8
3	0.8
4	0.64
5	0.8
6	0.56

TABLE V
VALUE OF THE PARAMETER $p(m)$ FOR THE MATRIX OF MODEL II FOR DIFFERENT VALUES m

m	*p(m)*
5	0.86
14	1.0
25	1.33
29	1.14
33	1.0
41	1.0
49	1.0
61	0.67

Now, p can be interpreted as a number that indicates how many times GMRES(*m*) method can be faster, if we use IWZ as a preconditioning method.

Table IV shows the value of p for the matrices of model I.

From table IV it can be deduced that the matrices of group I, the coefficient p is the same regardless of the size of the matrix and the rate is 0.8. While for the matrices of group II ratio p decreases with increasing size of the matrix.

Table V provides a value for $p(m)$ for model II. The value of $p(m)$ is the highest for $m = 25$ and for $m = 33$, $m = 41$, $m = 49$ is a constant which means that preconditioning has no effect on the rate of convergence, and for example $m = 61$ this ratio is less than 1 which means that the method IWZGMRES(61) is faster convergent than the GMRES(61).

B. The convergence rate of the GMRES

Figures 10 and 11 present relationship between the number of iterations and the convergence $\log(||Q^T x_i||_2)$ for the matrices with the number 3 and 4 for methods GMRES(*m*), IWZGMRES(*m*) for a few selected values of parameter m. The plot 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 IWZGMRES(*m*) method is faster convergent.

The convergence curve $\log(||Q^T x_i||_2)$ as a function i is almost of the same shape for a particular parameter m for the GMRES(*m*) method and the IWZGMRES(*m*) method, only for the IWZGMRES(*m*) method the curve is shifted upwards. It means that the IWZGMRES(*m*) method is faster convergent than the GMRES(*m*) method.

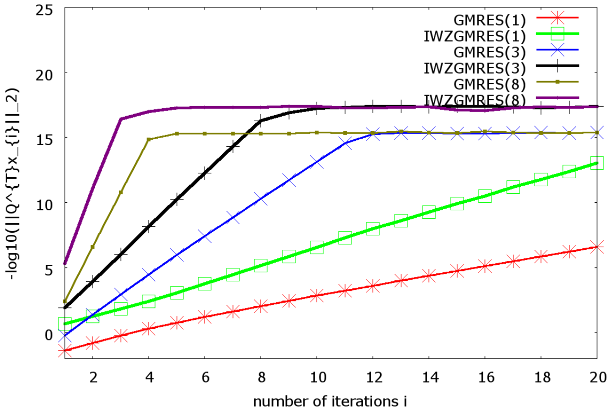


Fig. 10. Plot of the convergence curve $\log(||Q^T x_i||_2)$ as a function of i for the matrix 3

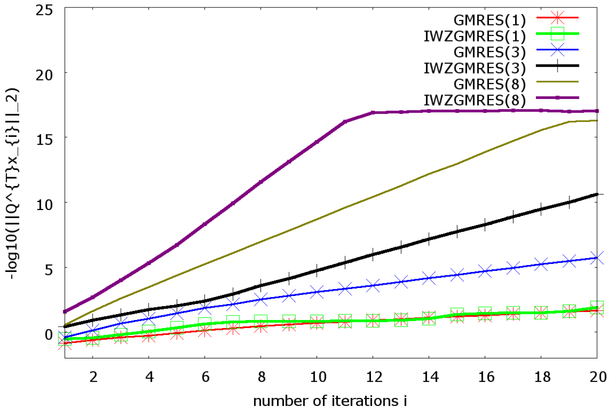


Fig. 11. Plot of the convergence curve $\log(||Q^T x_i||_2)$ as a function of i for matrix 4

Figures 12 and 13 show relationship between the number of the iterations and the convergence $\log(||Q^T x_i||_2)$ for matrices size, respectively 1500 and 3000 with different parameter d for the GMRES(*m*) methods and the IWZGMRES(*m*) methods for the parameter $m = 8$. The plots show that the GMRES(*m*) method and the IWZGMRES(*m*) method are faster convergent for matrices 3 and 5 than for matrices 4 and 6. The rate of the convergence depends on the d of the matrix and not on its

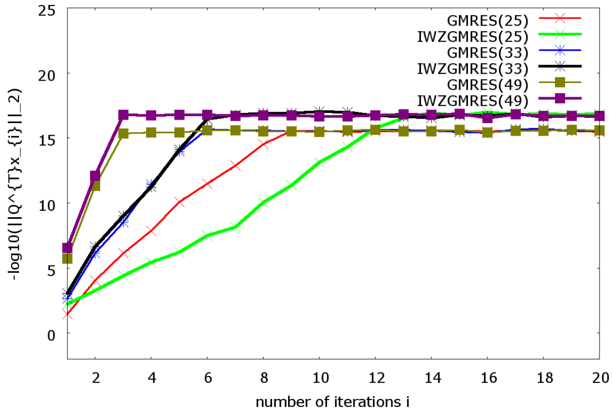


Fig. 14. Plot of the convergence curve $\log(||Q^T x_i||_2)$ as a function of i for the matrix of the model II

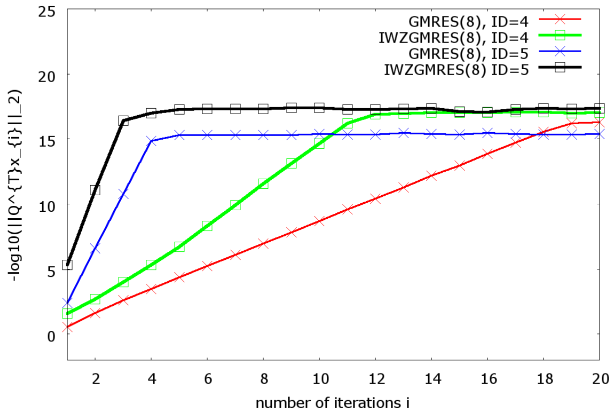


Fig. 12. Plot of the convergence curve $\log(||Q^T x_i||_2)$ as a function of i for the matrix 3 and 4

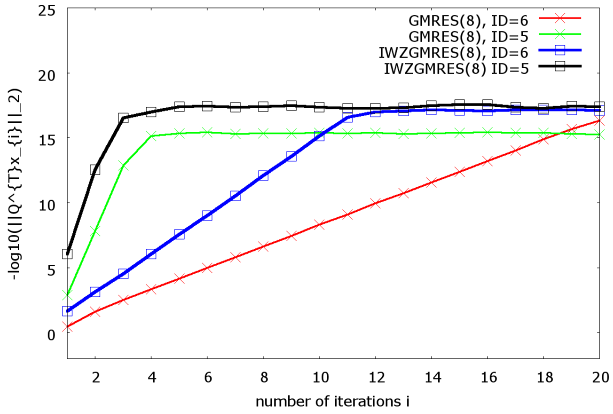


Fig. 13. Plot of the convergence curve $\log(||Q^T x_i||_2)$ as a function of i for the matrix 5 and 6

size.

Figure 14 shows relationship between number of iterations and the convergence $\log(||Q^T x_i||_2)$ for the GMRES(m) method and the IWZGMRES(m) method for parameter m with values $m = 25$, $m = 33$ i $m = 49$. Figure 14 and Table V present, that for certain values of m the method GMRES(m) had got a faster rate of convergence and for some the IWZGMRES(m) method had.

VII. Conclusion

Those numerical experiments helped us understand the effect of the nonzero structure and the size of the matrix on the convergence characteristics of preconditioned Krylov subspace methods like GMRES. The rate of convergence of the projection methods GMRES does not depend on the size of the matrix. Speed of convergence in terms of numbers of iterations of GMRES depends on the structure of the matrix. Tested matrices from two different models were characterized by the fact that the matrices of the model I have no structure, the matrix form the model II has got some structure.

Matrices from the first model was characterized by a parameter d. Namely, the set of matrices, which have a low value of the parameter d, ($d < 8$) are slowly convergent and require additional techniques to improve the rate of convergence. This technique was preconditioning.

The convergence, expressed in terms of p showed that we can identify the most optimal value of the parameter m, for which instead of the use of the GMRES(m) method we use the preconditioned GMRES(m) method, namely IWZGMRES(m).

On the basis of additional studies it may be concluded that irrespectively of the size and structure of the matrix GMRES($m + 1$) is faster convergent than the GMRES(m) for any matrix: similarly for the IWZGMRES(m) method the same dependence holds. With the increase parameter m, the rate of convergence of the method GMRES(m) increases, of course, up to some m_0, for which the rate is the largest, for all $m > m_0$ the rate is already the same.

The numerical example shows, that it is good to examine whether the matrix associated with a Markov chain is structured and has some properties, for example, the parameter d. If there is no structure you can use the preconditioning technique.

For the matrix of the model II it is not always the IWZGMRES(m) improves the convergence rate, because the matrices a structure. For this model we need to develop a separate algorithm to determine the vector of probabilities. These algorithms should take advantage of some properties of matrices associated with models.

References

[1] W. E. Arnoldi: The principle of minimized iteration in the solution of the matrix eigenvalue problem, *Quarterly for Applied Mathematics* 9, 1951, p. 17–29.

[2] M. Benzi, B. Ucar: Block Triangular preconditioners for M-matrices and Markov chains. [To appear in *Electronic Transactions on Numerical Analysis*.]

[3] B. Bylina, J. Bylina: Incomplete WZ decomposition algorithm for solving Markov chains, *Journal of Applied Mathematics*, vol. 1 (2008), n. 2, p. 147–156.

[4] J. Bylina: Distributed solving of Markov chains for computer network models, *Annales UMCS Informatica* 1 (2003), Lublin 2003, p. 15–20.

[5] J. Bylina, B. Bylina: GMRES dla rozwiazywania łańcuchów Markowa na komputerze wektorowym CRAY SV1, *Algorytmy, metody i programy naukowe*, Polskie Towarzystwo Informatyczne, Lublin 2004, p. 19–24 [in Polish].

[6] T. Dayar, W. J. Stewart.: Comparison of partitioning techniques for two-level iterative solvers on Large, Sparse Markov chins, *SIAM Journal on Scientific Computing* 21, 1691 (2000)

[7] 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.

[8] C. J. Ridler-Rowe: On a Stochastic Model of an Epidemic, *Advances in Applied Probability*, vol. 4, 1967, p. 19–33.

[9] Y. Saad: *Iterative methods for sparse linear systems*, SIAM, Philadelphia, 2003.

[10] Y. Saad: Krylov subspace methods for solving unsymmetric linear systems, *Mathematics of Computation* 37, 1981, p. 105-126.

[11] Y. Saad: Preconditioned Krylov subspace methods for the numerical solution of Markov chains, in: W. J. Stewart (Ed.), *Computations with Markov Chains*, Kluwer Academic, Dordrecht, 1995, p. 49–64.

[12] 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.

[13] W. Stewart: *Introduction to the Numerical Solution of Markov Chains*, Princeton University Press, Chichester, West Sussex 1994.

[14] P. Yalamov, D. J. Evans: The WZ matrix factorization method, *Parallel Computing* 21 (1995), p. 1111.

ISBN 978-83-60810-27-9
ISSN 1896-7094

On the Numerical Analysis of Stochastic Lotka-Volterra Models

Tuğrul Dayar[1], Linar Mikeev, and Verena Wolf
Department of Computer Science, Saarland University, Saarbrücken, Germany

***Abstract*—The stochastic Lotka-Volterra model is an infinite Markov population model that has applications in various life science domains. Its analysis is challenging since, besides an infinite state space with unbounded rates, it shows strongly fluctuating dynamics and becomes unstable in the long-run. Traditional numerical methods are therefore not appropriate to solve the system. Here, we suggest adaptations and combinations of traditional methods that yield fast and accurate solutions for certain parameter ranges of the stochastic Lotka-Volterra model. We substantiate our theoretical investigations with a comparison based on experimental results.**

I. Introduction

MORE than 80 years ago, Lotka and Volterra independently proposed the following three rules to describe the numerical evolution of two populations in interspecific competition [24, 30]:

1) The first population (prey) grows at rate αx, where x represents the population of prey.
2) The second population (predator) "eats" prey and grows at rate βxy, where y represents the population of predator.
3) The predator population decreases through natural death at rate γy.

The solution of the corresponding system of (non-linear) ordinary differential equations (ODEs)

$$\begin{aligned}\tfrac{dx}{dt} &= \alpha x - \beta xy\\ \tfrac{dy}{dt} &= \beta xy - \gamma y\end{aligned} \qquad (1)$$

shows sustained oscillations for positive constants α, β, γ along the closed curves

$$\beta x - \gamma \log x + \beta y - \alpha \log y = const$$

except if initially the system does start in the equilibrium point $x = \gamma/\beta, y = \alpha/\beta$. In Fig. 1(a), we plot the solution for $\alpha = \gamma = 1$ and $\beta = 0.01$ across time where we initially started with $x_0 = y_0 = 20$. Since its introduction, the Lotka-Volterra model became one of the most popular models in population dynamics, but it has also been successfully applied to neural networks [25] and game theoretic problems [16].

The stochastic Lotka-Volterra model assumes that x and y are represented by discrete random variables X and Y and that their evolution in time is given by a two-dimensional Markov process $\{(X(t), Y(t)), t \geq 0\}$ [11, 13, 20]. As opposed to the original Lotka-Volterra model, the stochastic variant takes into account the discreteness of the populations and their random fluctuations. In the stochastic model, extinction of species is possible and, depending on the initial conditions, the dynamics of the system can deviate drastically from the deterministic model. Recently, generalizations of the stochastic Lotka-Volterra model have been used to investigate the mechanisms that maintain biodiversity in Escherichia coli (E. coli) populations [18, 19, 26]. In such generalizations, the number of species is larger but the rules of interspecific competition remain the same.

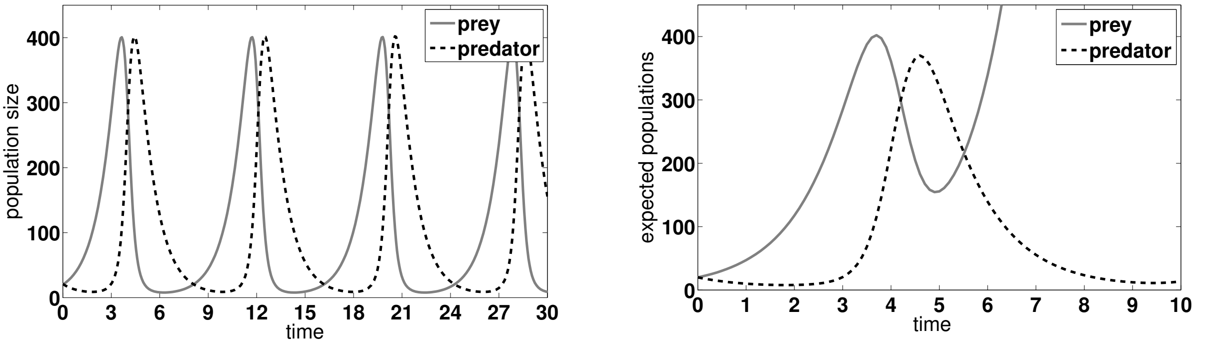
Here, we are interested in the transient probability distribution of the stochastic Lotka-Volterra model (cf. Section II). It can be used to derive measures such as the distribution of the time until extinction, the probability of extinction until a certain time instant, the expected populations, etc. For a transient solution of the stochastic Lotka-Volterra model, the underlying Markov process has to be solved; this involves the solution of a system of ordinary differential equations known as the master equation [17]. Standard methods for the transient solution of Markov processes are based on the numerical integration of the differential equations, on uniformization of the process, or on approximations in the Krylov subspace (for an overview see [28]). An alternative approach, called method of moments, has recently been proposed by Engblom and is based on a deterministic approximation of the moments of the Markov process described by the master equation [10].

Already in the case of two species, the analysis of the stochastic Lotka-Volterra model using standard methods, such as numerical integration methods or approximations of the moments become inefficient or even infeasible for several reasons. The state space is infinite in two dimensions and it is non-trivial to truncate the state space in such a way that the number of states in the finite truncation remains tractable. Before one of the species becomes extinct, the expected populations oscillate in a similar way as in the deterministic model. Therefore the dynamics of the system change drastically. If the population of at least one of the two species is high, then a large number of events occur even in small time intervals. Moreover, the rates are unbounded; that is, the system is not uniformizable. In order to describe the strong non-linear dependence between the two populations, higher-order moments are necessary and a deterministic approximation of two or more moments yields poor results.

In this paper, we suggest several adaptations and combinations of standard methods that render an efficient solution of the stochastic Lotka-Volterra model possible. We concentrate on the numerical integration of the master equation (cf. Section III) as well as the method of moments (cf. Section IV).

[1] T. Dayar is currently on sabbatical leave from Bilkent University, Turkey.

978-83-60810-27-9/09/$25.00 © 2010 IEEE



(a) Sustained oscillations of the deterministic Lotka-Volterra model.

(b) Expected populations in the stochastic Lotka-Volterra model.

Fig. 1. ODE solution and stochastic solution of the Lotka-Volterra model where $\alpha = \gamma = 1$, $\beta = 0.01$, and $x_0 = y_0 = 20$.

We extend these methods in such a way that the cheap but inaccurate method of moments becomes more accurate and the expensive but accurate numerical integration becomes faster. In Section V, we propose a stochastic hybrid approach that dynamically switches between a moment-based representation and a distribution-based representation as well as having the ability to simultaneously use a combination of both. We implemented all methods in C++ and performed extensive simulations of the stochastic Lotka-Volterra model in order to provide comparisons in terms of accuracy and run time. Numerical experiments on a number of problems indicate that the stochastic hybrid approach is overall the best method. It is at least an order faster than the numerical integration of the master equation and yields no more than a 5 % relative error in the first moment of the transient distribution.

II. Stochastic Lotka-Volterra model

We describe the transitions of the stochastic Lotka-Volterra model in a guarded command style [1, 23]. A guarded command has the form `guard |- rate -> update;` and it describes a set of transitions in the underlying Markov process. The `guard` is a Boolean predicate that determines in which states the transition is possible, the `rate` is the real-valued function that evaluated in the current state gives the positive transition rate in the Markov process, and `update` is the function that calculates the successor state of the transition. Let x and y be the positive integers that represent the populations of prey and predator, respectively. The three different possible transitions are given as follows:

$$\begin{array}{llll}
\mathtt{x} > 0 & \mathtt{|-} & \alpha \cdot \mathtt{x} & \mathtt{->} \quad \mathtt{x} := \mathtt{x} + 1; \\
\mathtt{x} > 0, \mathtt{y} > 0 & \mathtt{|-} & \beta \cdot \mathtt{x} \cdot \mathtt{y} & \mathtt{->} \quad \mathtt{x} := \mathtt{x} - 1, \mathtt{y} := \mathtt{y} + 1; \\
\mathtt{y} > 0 & \mathtt{|-} & \gamma \cdot \mathtt{y} & \mathtt{->} \quad \mathtt{y} := \mathtt{y} - 1;
\end{array}$$

The above guarded command model specifies the elements of the infinitesimal generator matrix Q of a Markov process $\{(X(t), Y(t)), t \geq 0\}$ which takes states (x, y) in $\mathbb{N}^2$. More precisely, if for a pair (x, y) the guard is true, then the element that belongs to (x, y) and the update of (x, y) is equal to the rate. E.g. the row of state $(2, 2)$ contains three positive entries. One for the transition to state $(3, 2)$ at rate 2α, one for the transition to state $(1, 3)$ at rate 4β, and one for the transition to state $(2, 1)$ at rate 2γ. The diagonal element is then given by the negative sum of the off-diagonal elements. It is easy to see that $\{(X(t), Y(t)), t \geq 0\}$ is a regular Markov process [8]. We remark that the outflow rate from state (x, y) with $x, y > 0$ is $(\alpha x + \gamma y + \beta xy)$, where $\alpha, \gamma, \beta > 0$, and this rate increases unboundedly with increasing values of x and y.

Let us fix the initial state of the system as (x_0, y_0). Then the transient probability distribution is given by the master equation

$$\tfrac{d}{dt} p^{(t)}(x, y) = \mathcal{M}(p^{(t)}(x, y)), \tag{2}$$

where $p^{(t)}(x, y) = P\{X(t) = x, Y(t) = y | X(0) = x_0, Y(0) = y_0\}$. The master operator $\mathcal{M}$ is defined for any real-valued function $g : \mathbb{N}^2 \to \mathbb{R}$ such that $\mathcal{M}(g)$ is the function that maps a state (x, y) to the value[1]

$$\begin{aligned}
\mathcal{M}(g(x, y)) = {} & \alpha(x-1)g(x-1, y) \\
& + \beta(x+1)(y-1)g(x+1, y-1) \\
& + \gamma(y+1)g(x, y+1) \\
& - (\alpha x + \beta xy + \gamma y)g(x, y),
\end{aligned} \tag{3}$$

where α, β, γ are the rate constants as defined before and $x, y > 0$. If $x = 0$ and/or $y = 0$, then the terms involving $g(x-1, y)$ and/or $g(x+1, y-1)$ are removed. The ordinary first-order differential equation in (2) is a direct consequence of the Kolmogorov forward equation and describes the change of the probability distribution as the difference between inflow of probability from direct predecessors (first three terms) and outflow of probability in state (x, y) (last term).

III. Direct numerical approximation

Since no analytical solution is known for Eq. (2) and the number of equations is infinite in two dimensions, a numerical solution is only possible if appropriate bounds for the variables x and y are found.

[1] We assume that all terms with a negative argument are zero (e.g. $\alpha(x-1)g(x-1, y) = 0$ if $x < 1$).

A. Dynamic numerical integration

Here, we suggest to construct the state space in a dynamic manner up to a certain bound. We discretize time and integrate over small time steps. In the first step, state (x_0, y_0) has probability 1 and, for a time step $h > 0$, we integrate Eq. (2) by considering only those states that have a non-zero probability during the next h time units. For the numerical integration, we use an explicit fourth-order Runge-Kutta method and thus in each step we add the states within a distance of four transitions from the current set of states.

The standard explicit fourth-order Runge-Kutta method applied to Eq. (2) yields the integration step [28]

$$\begin{aligned} p^{(t+h)}(x,y) = p^{(t)}(x,y) + \tfrac{h}{6}\Big(& k^{(1)}(x,y) \\ &+ 2k^{(2)}(x,y) + 2k^{(3)}(x,y) + k^{(4)}(x,y)\Big), \end{aligned} \tag{4}$$

where $h > 0$ is the time step of the method. For $i \in \{1, 2, 3, 4\}$ the values $k^{(i)}(x, y)$ are defined recursively as

$$\begin{aligned} k^{(1)} &= \mathcal{M}(p^{(t)}), \\ k^{(2)} &= k^{(1)} + \tfrac{h}{2}\mathcal{M}(k^{(1)}), \\ k^{(3)} &= k^{(1)} + \tfrac{h}{2}\mathcal{M}(k^{(2)}), \\ k^{(4)} &= k^{(1)} + h\mathcal{M}(k^{(3)}). \end{aligned} \tag{5}$$

Let $\tilde{p}^{(t)}$ be the approximation of $p^{(t)}$ at time t. Given $\tilde{p}^{(t)}$, we integrate Eq. (2) for h time units as follows. Let $S \subseteq \mathbb{N}^2$ be the set of states (x, y) with $\tilde{p}^{(t)}(x, y) > 0$. Each state $(x, y) \in S$ is represented as an array with entries $k^{(1)}(x, y), \ldots, k^{(4)}(x, y)$, $p(x, y)$. The former four entries are initialized with 0 while $p(x, y) = \tilde{p}^{(t)}(x, y)$ due to the previous integration step and, in the first integration step, $p(x, y) = 1$ if $x = x_0$, $y = y_0$ and $p(x, y) = 0$ otherwise. We have five substeps in which we go over all elements of S to compute for each state the four k-values as well as $\tilde{p}^{(t+h)}(x, y)$. In the first substep, each state (x, y) adds the three outflow probabilities $\alpha x p(x, y)$, $\beta x y p(x, y)$, and $\gamma y p(x, y)$ to the array element $k^{(1)}$ of the corresponding successor states $(x + 1, y)$, $(x - 1, y + 1)$, and $(x, y - 1)$ (see Eqs. (3) and (5)). Whenever a successor state is not in S, we add it to S. For $i \in \{2, 3\}$, in the i-th substep we first add $k^{(1)}(x, y)$ to the element for $k^{(i)}(x, y)$ and then add $\frac{h}{2}\alpha x k^{(i-1)}(x, y)$, $\frac{h}{2}\beta x y k^{(i-1)}(x, y)$, and $\frac{h}{2}\gamma y k^{(i-1)}(x, y)$ to the $k^{(i)}$-field of the corresponding successors $(x + 1, y)$, $(x - 1, y + 1)$, and $(x, y - 1)$ (again we add successors to S whenever they do not yet belong to S). The fourth substep is identical to the second and third except that $\frac{h}{2}$ is replaced by h (cf. Eq. (5)). In the fifth substep, we compute for each state (x, y) the probability $\tilde{p}^{(t+h)}(x, y)$ according to Eq. (4).

The dynamic numerical integration procedure described above yields accurate approximations of the transient probability distribution $p^{(t)}$ of the infinite Markov process $\{(X(t), Y(t)), t \geq 0\}$. Its drawback, however, is that the size of set S becomes very large since in each integration step states within a distance of four transitions are added. E.g. after a computer time of five hours and a time horizon of $t = 0.317$ we run out of memory on a 64-bit machine with 8 GB of main memory where we used the same parameters as in Fig. 1. At that time instant, the state space contained about 35 million states.

TABLE I
RESULTS OF AN APPROXIMATE DIRECT SOLUTION OF EQ. (2).

δ	run time	$\lvert S\rvert$	error
1e-15	51h 39min	9e5	2e-7
1e-14	40h 51min	7e5	2e-6
1e-12	22h 46min	4e5	9e-5
1e-10	10h 12min	2e5	6e-3

B. Inexact numerical integration

Similar to the approach in [9], we modify the numerical integration procedure described above as follows. During the first four substeps of the integration step, we only add new states to set S if their probability is greater than a small positive threshold δ. This has shown to lead to a significant reduction of the size of S while, for most systems, the approximation error is small. For instance, for $\delta = 10^{-15}$ the size of set S at time $t = 0.317$ is just 0.004% of the size of S for $\delta = 0$, while the total probability that got "lost", when states with probability smaller than δ were ignored, is only 7×10^{-12}.

In Table I we list our results of approximate direct solution of Eq. (2) for different values of δ. We analyzed the system for a time horizon of $t = 10$ and used the same parameters as in Fig. 1. For other parameters, such as those that we used for our experiments in Section VI, the results are similar (not shown). The column labeled $|S|$ lists the average size of set S, i.e. average number of states considered during one integration step. The last column gives the total probability that got "lost" due to the state space truncation. If the numerical integration were exact, the values in the last column would contain the total error of the approximation, i.e. the sum of all absolute errors of the state probabilities. Note that the resulting probability distributions provide full information about the system, i.e. arbitrary moments of the distribution as well as probability of certain events (such as extinction until time t) can be derived.

With the proposed numerical approach, we are able to solve the stochastic Lotka-Volterra model and accurate results are obtained. The method is, however, rather slow whenever the expected populations become large. This is because large populations are represented by a large number of discrete states, even though the relative variance is small. This problem gets worse when the populations oscillate in an even higher range as, for instance, in Fig. 2. On the other hand, a discrete stochastic representation is necessary whenever the populations become small.

Also, if the underlying system is stiff the explicit Runge-Kutta method will perform poorly and should be replaced by implicit finite difference methods. For instance, if the transition rates in the Markov process have very different orders of magnitude, the time step of explicit methods will be proportional to the fastest time scale. Besides implicit methods, it is possible to apply aggregation techniques that are designed for stiff systems [6, 7].

IV. Deterministic approximation of moments

Deterministic approximations are, besides Monte-Carlo simulation, the analysis techniques with most widespread use. The mathematical justification of the simplest deterministic approximation, mean-field analysis, has first been provided by Kurtz in the context of chemical kinetics [21] where it is applied extensively.

Mean-field analysis relies on the assumption that the expected populations can be approximated by variables that change continuously and deterministically in time. This idea can be generalized to higher moments where the accuracy of the approximation increases as higher moments are included [10].

A. Mean-field approximation

Let $Z(t) = (X(t), Y(t))$ denote the two-dimensional population vector at time t and, for $d \in \mathbb{N}$ let $f : \mathbb{N}^2 \to \mathbb{R}^d$ be a function that is independent of t. Assume that the expectation $E[f(Z(t))]$ exists. From Eq. (2), it is straightforward to derive the relationship[2]

$$\begin{aligned}\tfrac{d}{dt}E[f(Z)] &= \textstyle\sum_{(x,y)\in\mathbb{N}^2} f(x,y)\tfrac{d}{dt}p^{(t)}(x,y)\\ &= E[\alpha X(f(X+1,Y)-f(Z))\\ &\quad +\beta XY(f(X-1,Y+1)-f(Z))\\ &\quad +\gamma Y(f(X,Y+1)-f(Z))].\end{aligned} \tag{6}$$

If $f(x,y) = (x,y)$ we get as a special case that

$$\begin{aligned}\tfrac{d}{dt}E[X] &= \alpha E[X] - \beta E[XY]\\ \tfrac{d}{dt}E[Y] &= \beta E[XY] - \gamma E[Y].\end{aligned} \tag{7}$$

Thus, if $X(t)$ and $Y(t)$ were uncorrelated (which means that $E[XY] = E[X]E[Y]$), we could compute the expected number of individuals using Eq. (1). But if $X(t)$ and $Y(t)$ are highly correlated, the approximation of the mean in Eq. (1) is rather inaccurate. Fig. 1(a), for instance, shows the mean-field approximation of the Markov process whose true expectations are plotted in Fig. 1(b).

For finite time instants t, the mean-field approximation becomes exact for the scaled process $\{N^{-1}Z(t), t \geq 0\}$ as the scaling constant N approaches infinity under the assumption that the rate functions are density dependent [22]. In the context of the stochastic Lotka-Volterra model, the latter assumption means that the constant β depends on the scaling parameter N (which has a natural interpretation such as the total number of individuals) while α and γ are independent of N. Thus, the accuracy of the approximation increases if the populations become large. In Fig. 2 we plot the mean-field approximation versus the true expected populations for $\alpha = \gamma = 1$, $\beta = 0.003$ and initial state $(180, 200)$. For these parameters, both populations stay above 150 at all times in the deterministic model. Here, the mean-field approximation remains very accurate at least until time $t = 15$ (and the same holds for the expected number of preys).

B. Method of moments

Eq. (7) suggests that we would obtain a more accurate approximation of the expected number of individuals if we could accurately approximate the value $E[XY]$. The first idea, of course, is to set $f(x,y) = xy$ in Eq. (6) in order to extend the system of differential equations in Eq. (7) by additional equations for $E[XY]$. It is easy to see that with $f(x,y) = xy$ Eq. (6) yields a differential equation that involves $E[X^2Y]$, $E[X^2Y^2]$, and $E[XY^2]$. To approximate these quantities as well, additional equations are necessary that involve even higher moments. This argument repeats and yields an infinite set of differential equations. The idea of method of moments is to truncate this infinite set after a finite number of equations. Here, we improve the quality of the deterministic approximation by enriching the system in Eq. (7) with deterministic approximations of the second moments as proposed by Engblom for Markov population models [10]. Based on a Taylor series expansion of the rate function, the following deterministic approximation of second order can be shown:

$$\begin{aligned}\tfrac{d}{dt}E[X] &= \alpha E[X] - \beta E[X]E[Y] - \beta c_{XY}\\ \tfrac{d}{dt}E[Y] &= \beta E[X]E[Y] + \beta c_{XY} - \gamma E[Y]\\ \tfrac{d}{dt}c_{XX} &= 2\alpha c_{XX} + \alpha E[X] - 2\beta E[X]c_{XY}\\ &\quad -2\beta E[Y]c_{XY} + \beta E[X]E[Y] + \beta c_{XY}\\ \tfrac{d}{dt}c_{XY} &= \alpha c_{XY} - \beta E[Y]c_{XY} - \beta E[X]c_{YY}\\ &\quad +\beta E[Y]c_{XX} + \beta E[X]c_{XY}\\ &\quad -\beta E[X]E[Y] - \beta c_{XY} - \gamma c_{XY}\\ \tfrac{d}{dt}c_{YY} &= 2E[Y]c_{XY} + 2\beta E[X]c_{YY} + \beta E[X]E[Y]\\ &\quad +\beta c_{XY} - 2\gamma c_{YY} - \gamma E[Y],\end{aligned} \tag{8}$$

where c_{XX}, c_{XY}, c_{YY} approximate the covariances $COV[X,X]$, $COV[X,Y]$, $COV[Y,Y]$, respectively. Note that the first two equations are as in Eq. (7) if we use the relationship $COV[V,W] = E[VW] - E[V]E[W]$ for the two random variables V and W.

In Fig. 2 we plot the solution of Eq. (8) together with the mean-field solution as well as the true expectations. For these parameters both the mean-field approximation and the second-order method of moments become inaccurate after $t = 23$. The second-order method of moments predicts the dynamics of the expected predator population only slightly better. The situation is similar for the expected number of preys. Additional equations for higher moments give a more accurate approximation (not shown) but become very stiff once the prey population grows too fast. The main advantages of the method of moments are the low computational cost (the ODE can be solved in a few seconds) and the fact that for many models a second-order approximation is sufficient. If,

[2] To improve readability, we omit the dependence on t and write Z instead of $Z(t)$ and so on.

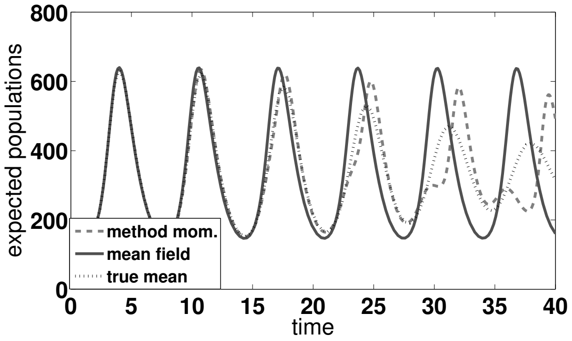


Fig. 2. The deterministic approximations of the expected number of predators remain accurate until $t = 23$. Here, we used the parameters $\alpha = \gamma = 1$, $\beta = 0.003$, and initial state $(180, 200)$.

however, besides the moments, certain probabilities are of interest (such as the probability of extinction) other solution techniques need to be chosen. Also, for the model under study higher order equations are necessary but lead to numerical instabilities. E.g. for the parameters considered in Fig. 1, the fourth-order approximation gives a matrix that is singular to working precision (while the second-order approximation yields poor accuracy). More advanced numerical techniques have to be considered in this case such as implicit finite difference methods and integration methods that are particularly designed for oscillatory systems [2–5, 29].

V. Stochastic hybrid method

In this section, we propose a hybrid solution technique for the stochastic Lotka-Volterra model that is motivated by two observations. First, the approximate direct numerical method presented in Section III-B is able to provide accurate results. However, if only one of the expected populations (say $E[Y(t)]$) is small and the other population is high, then it becomes inefficient because the number of "significant" states is large. It treats $X(t)$ and $Y(t)$ as discrete stochastic (DS) random variables and considers all possible values up to a predefined accuracy $\delta > 0$, i.e. the infinite ranges of $X(t)$ and $Y(t)$ are truncated with respect to δ. Second, the method of moments with a truncation of moments higher than order two provides a fast continuous deterministic (CD) approximation that is accurate whenever the expectations of $X(t)$ and $Y(t)$ are high.

A successful hybrid approach could therefore be based on the following two properties:

(a) It should be able to switch dynamically between a DS representation and a CD representation depending on population thresholds for $E[X(t)]$ and $E[Y(t)]$.

(b) Whenever $X(t)$ is represented as a DS variable and $Y(t)$ as a CD variable (or vice versa), a hybrid approach must be able to take into account the dependencies between $X(t)$ and $Y(t)$ in an appropriate way.

While requirement (a) is easy to realize, (b) turns out to be much more challenging. Assume that at a certain point in time, the predator population is represented by the DS variable $Y(t)$ and the prey population is represented by the CD variable $x(t) \approx E[X(t)]$; that is, we have a (truncated) probability distribution for $Y(t)$ and a single real value $x(t)$ as well as approximations of the covariances c_{XX} and c_{XY}. A straightforward idea is to consider the "global" ODE in Eq. (8) for the computation of $x(t+h) \approx E[X(t+h)]$ as well as the covariances at time $t+h$, where h is a small time step and $E[Y(t)]$ as well as c_{YY} are computed from the current approximation of the distribution of $Y(t)$. The equations for $E[Y(t)]$ and c_{YY} are removed in Eq (8) and the distribution of $Y(t+h)$ is computed by solving a "reduced" version of Eq. (2) given by

$$\begin{aligned}\tfrac{d}{dt}p^{(t)}(y) = &\ \beta(y-1)x(t)p^{(t)}(y-1)\\ &+ \gamma(y+1)p^{(t)}(y+1)\\ &- (\beta y x(t) + \gamma y)p^{(t)}(y),\end{aligned} \tag{9}$$

where $y > 0$. For state $y = 0$, we have $\frac{d}{dt}p^{(t)}(0) = \gamma p^{(t)}(1)$. Note that the transition that corresponds to the growth of prey is not included in Eq. (9) but is taken into account in Eq. (8). We can integrate Eq. (9) using the method described in Section III-B. In this way, the possible values for y and thus the number of equations would remain small (since we assumed that $E[Y(t)]$ and h are small) which makes the approach computationally cheap. For instance, the run time needed to approximate the probability distribution for the parameters used in Section I is only about 1 minute. However, it turns out that the approach yields bad accuracy. For instance, at the final time instant $t = 10$, the relative errors of the first moments of prey and predator are above 20%. In particular, the approach fails to detect the steep increase of the expected number of preys after $t = 6$ even though the results are accurate within $[0, 6]$. The reason is that when $E[Y(t)]$ is small, the (relative) variance of $Y(t)$ is important for the evolution of the CD variable $x(t)$ (and vice versa). For instance, if $Y(t) = 0$, then $X(t)$ grows exponentially at rate α. In this case, the total rate of change will therefore deviate largely from the rate $\frac{d}{dt}E[X]$ used in Eq. (8).

A more accurate approach is to consider "local" ODEs, i.e. if $X(t)$ has a CD representation and $Y(t)$ has a DS representation, then we consider the conditional expectation $E[X(t)|Y(t) = y]$ for each state y. Thus, in each integration step, we represent the current state of the system by the probabilities $p^{(t)}(y) \approx P\{Y(t) = y\}$ (but we still neglect probabilities smaller than or equal to δ) and real values $x_y(t)$ that approximate $E[X(t)|Y(t) = y]$. For a small time step h, we integrate the distribution $p^{(t)}$ and the values $x_y(t)$ in three substeps.

(1) We first integrate $p^{(t)}$ according to Eq. (9) to approximate the probabilities $P\{Y(t+h) = y\}$ by $p^{(t+h)}(y)$.

(2) For each state y with $p^{(t)}(y) > \delta$, we compute $\tilde{x}_y(t+h)$ by numerical integration of the ODE

$$\tfrac{d}{dt}\tilde{x}_y(t) = \alpha\tilde{x}_y(t) - \beta y\tilde{x}_y(t)$$

with initial condition $\tilde{x}_y(t) = x_y(t)$. Note that $\tilde{x}_y(t+h)$ is *not* an approximation of $E[X(t+h)|Y(t+h) = y]$ since the above

differential equation does not take into account that Y may leave state y within $[t, t{+}h]$. E.g. for newly discovered states with $p^{(t)}(y) < \delta$ and $p^{(t+h)}(y) > \delta$ the value $x_y(t)$ (and thus $\tilde{x}_y(t{+}h)$) does not exist. Thus, a third substep is necessary to approximate $E[X(t{+}h)|Y(t{+}h) = y]$.
(3) We compute $x_y(t{+}h)$ by "distributing" $\tilde{x}_y(t{+}h)$ according to the change in the distribution of Y as explained below. If we assume that $[t, t+h]$ is an infinitesimal time interval and, for $y \neq y'$, $q(y, y', h)$ is the probability to enter y from y' within $[t, t+h]$, then

$$\begin{aligned}P\{Y(t{+}h){=}y\} = &\sum_{y'\neq y} q(y,y',h)P\{Y(t){=}y'\}\\ &+(1-\sum_{y'\neq y} q(y',y,h))P\{Y(t){=}y\}.\end{aligned} \tag{10}$$

Thus, we approximate $E[X(t+h)|Y(t+h) = y]$ as

$$\begin{aligned}&\sum_{y'\neq y} \tilde{x}_{y'}(t{+}h)q(y,y',h)P\{Y(t){=}y'|Y(t{+}h){=}y\} \\ &+ \tilde{x}_y(t{+}h)(1{-}\sum_{y'\neq y} q(y',y,h))P\{Y(t){=}y|Y(t{+}h){=}y\}.\end{aligned} \tag{11}$$

The rationale behind this approximation is that, for $t' \in [t, t + h)$, the variance $VAR[X(t')|Y(t') = y]$ is small and the conditional distributions $P\{X(t') = \hat{x} \mid Y(t') = y\}$ can be approximated by a normal distribution with mean $E[X(t')|Y(t') = y]$.

Obviously, we make use of the current approximations $p^{(t)}$ and $p^{(t+h)}$ to compute the conditional probabilities $P\{Y(t){=} y'|Y(t{+}h) = y\}$. For a small time step h, $q(y, y+1, h) \approx h\beta y\tilde{x}_y(t)$ and $q(y, y{-}1, h) \approx h\gamma y$. Using Eq. (11), we define $x_y(t{+}h) \approx E[X(t{+}h)|Y(t{+}h) = y]$ as

$$\begin{aligned}x_y(t{+}h) = \Big(&h\beta(y{-}1)p^{(t)}(y{-}1)\tilde{x}_{y-1}(t{+}h)\tilde{x}_{y-1}(t)\\ &+h\gamma(y{+}1)p^{(t)}(y{+}1)\tilde{x}_{y+1}(t{+}h)\\ &+(1{-}hy(\beta\tilde{x}_y(t)+\gamma))p^{(t)}(y)\tilde{x}_y(t{+}h)\Big)/p^{(t+h)}(y).\end{aligned}$$

We illustrate the three substeps above by means of a simple example.

Assume that the distribution of $Y(t)$ is such that $p^{(t)}(0) = \frac{1}{3}$, $p^{(t)}(1) = \frac{2}{3}$ and all other states y have probability 0 at time t. Assume further that the CD variables $x_y(t)$ are such that $x_0(t) = 600$ and $x_1(t) = 500$. For the parameters $\alpha = \gamma = 1$, $\beta = 0.01$, $h = 0.1$ substep (1) yields $p^{(t+h)}(0) = \frac{2}{5}$, $p^{(t+h)}(1) = \frac{4}{15}$, $p^{(t)}(2) = \frac{1}{3}$ where, for simplicity, we integrate Eq. (9) with the Euler method. Substep (2) yields $\tilde{x}_0(t{+}h) = 660$ and $\tilde{x}_1(t{+}h) = 549.5$ and substep (3) gives $x_0(t{+}h) \approx 641.58$ and $x_1(t{+}h) = x_2(t{+}h) = 549.5$. Note that state $y = 2$ "inherits" the x-value of state $y = 1$ since its probability inflow of $\frac{1}{3}$ originated from $y = 2$.

The same strategy as explained above can be used for the case where $X(t)$ is a DS variable and $y(t)$ is a CD variable.

We performed experiments using different parameters to test the accuracy and run time if the representation is hybrid (coexistence of CD and DS variables) and also if switching occurs often. For the parameters used in Section I, for instance, we start with a purely stochastic representation and use a population threshold of 350 to switch the representation. At time $t \approx 3.3$ we switch the representation of $X(t)$ from DS to CD since the expectation reaches 350 (see also Fig. 1(b)). Around time $t = 4$, we switch back to a DS representation. Then, around $t = 4.3$ we switch the representation of $Y(t)$ from DS to CD because $E[Y(t)] > 350$, etc.

TABLE II
RESULTS OF THE STOCHASTIC HYBRID SOLUTION.

pop. thresh.	δ	run time	$\lvert S\rvert$	relative error of moments		
		time		1st	2nd	3rd
100	$1e$-15	12s	$9e2$	0.04	0.11	0.22
	$1e$-10	7s	$6e2$	0.04	0.11	0.22
200	$1e$-15	1h 43min	$5e4$	0.03	0.11	0.23
	$1e$-10	11min	$9e3$	0.03	0.11	0.23
400	$1e$-15	2h 35min	$1e5$	0.03	0.10	0.20
	$1e$-10	25min	$3e4$	0.03	0.11	0.21

We also performed experiments using different population thresholds and different values for δ. We summarize these results in Table II. We always compared our results with the full stochastic solution described in Section III-B to estimate the accuracy of the hybrid method. We list the relative errors of the first three moments when the hybrid method is compared to the full stochastic solution. For these parameters it turned out that the accuracy of the hybrid approach is less sensitive to population threshold than expected. In the range of $100 - 400$, the relative error of the first three moments is at most 23%. Similarly, the moments are accurately approximated even if δ is about 10^{-10}. It should, however, be noted that if a higher value is chosen for δ, then small event probabilities will be approximated as 0.

The accuracy of the hybrid approach becomes worse whenever both variables are represented deterministically. The reason is that the second-order method of moments relies heavily on an accurate prediction of the (co-)variances. But during a preceding hybrid phase where, say, X has CD representation and Y has DS representation, the variance of X is much smaller than the real variance $VAR[X(t)]$, i.e. the approximation

$$VAR[X(t)] \approx \textstyle\sum_{y:p^{(t)}(y)>\delta} p^{(t)}(y)(x_y(t) - E[X(t)])^2$$

is not accurate enough if it is used as an initial value for c_{xx} in Eq. (8) even though this has only a minor effect on accuracy during the hybrid phase. The hybrid method exploits the fact that, if X is large, then the *relative* variance is small and it is sufficient to record the evolution of $E[X(t) \mid Y(t) = y]$ for each value y that has a significant probability. This seems to contradict with the approximation in Eq. (8) that relies on a good estimate for the (co-)variances. For this reason, we suppressed the configuration CD, CD for our results in Table II and always kept at least one variable DS. We plan to extend the stochastic hybrid approach such that the distribution $P\{X(t) = x \mid Y(t) = y\}$ is approximated by a normal distribution which we represent by two values (mean and variance) instead of the single value $x_y(t)$. This, however, complicates the interdependencies between X and Y further and its study is beyond the scope of this paper.

VI. Experimental results

We performed experiments using five different parameter combinations for the stochastic Lotka-Volterra model and four different solution methods. In Table III we list our experimental results. Each row corresponds to one combination and is enumerated in the first column. We used the first and the second combination of parameters also in Section III and IV (see also Fig. 1 and Fig. 2). Note that for the latter combination, the expected populations oscillate in a much higher range. The remaining parameter combinations in rows 3 through 5 are taken from the literature [14] (third row), [15] (fourth row), [12] (fifth row). In columns two to four we list the rate constants, and the columns with labels x_0 and y_0 refer to the initial population sizes. In the column with label t_f, we list the final time instant at which we compute the transient distribution. We abbreviate the solutions methods as follows:

- IRK4(δ) refers to the inexact explicit fourth-order Runge-Kutta method as described in Subsection III-B using a significance threshold of δ. We list the run time, the average number of significant states and the error, i.e., the probability mass lost during the computation for the case $\delta = 10^{-15}$.
- MF refers to the mean field analysis discussed in Subsection IV-A. We list the run time and the relative error of the first moment (expected populations) at the final time instant where we compare to IRK4(10^{-15}). We took the average over the two species.
- MM refers to the method of moments that we discussed in Subsection IV-B. We list the run time and the relative error of the first and second moments where we again compare to IRK4(10^{-15}) and average over the two species.
- Finally, SH(δ,K) refers to the stochastic hybrid method where a significance threshold of δ and a population threshold of K is used. Here, we give results for $\delta = 10^{-15}$ and $K = 200$. The column with label $|S|$ lists the average number of significant states in each step and the last three columns list the relative error of the first, second, and third moments compared to those of IRK4(10^{-15}).

For the first combination of parameters the two deterministic approxmations MF and MM yield fast but poor results. The stochastic hybrid method gives a significant speed-up compared to IRK4(10^{-15}) (about 51 times faster) and the approximation is accurate. The parameters used in the second row yield a system where the expected populations oscillate in a high range (between 150 and 650, see also Fig. 2). Therefore, MF and MM give accurate solutions. The stochastic hybrid solution is fast since most of the time both populations are represented as continuous deterministic values. The third combination of parameters gives similar results as the first one. In the last two rows, only IRK4(10^{-15}) gives an accurate solution. The stochastic hybrid method accurately predicts the first moments of the species but for the second and third moments the approximation error is high. The reason is that, for both parameter combinations, at the final time instant $E[Y] \approx 0$ and $|S| = 1$ (fourth row) and $|S| = 3$ (fifth row). During the hybrid solution, Y is represented as a discrete stochastic variable and X becomes continuous deterministic. Only a small number of states y remain and the variance between their conditional expectations $E[X|Y = y]$ is small. Thus, the second moment $E[X^2]$ yields a relative error of around 90% whereas $E[Y^2] \approx 0$ is very accurately approximated. The average relative error is then around 50%.

VII. Conclusion

The popularity of the Lotka-Volterra model for the description of population dynamics relies on the fact that this model is able to reflect the strongly changing dynamics of interacting populations. Its stochastic variant includes the possibility of extinction, which is, for instance, important for modeling biodiversity and coevolution [18, 19, 26].

The analysis of the stochastic Lotka-Volterra model is challenging because events such as extinction of predator require a costly discrete representation of the predator population. This and other characteristics of the stochastic Lotka-Volterra model (change between small and large populations, highly correlated variables, infinite state space, unbounded rates) are prototypical for Markov population models. Therefore, its analysis is not only interesting from an application-oriented viewpoint but also from a methodological viewpoint.

We discussed different analysis approaches for the stochastic Lotka-Volterra model and modified them in order to improve accuracy and run time. It turned out that the stochastic hybrid approach with local ODEs performs best. It is at least an order faster than the numerical integration of the master equation and yields no more than a 5 % relative error in the first moment of the transient distribution. If high populations are approximated by continuous deterministic variables, then error estimates are hard to obtain. The benefits of an approximate numerical solution, however, are highly dependent on accurate error estimates. Here, we compared our hybrid solution with a full stochastic solution. In the future, we will try to estimate the accuracy of the hybrid approach without a comparison to a more accurate but costly solution. Moreover, we plan to additionally consider numerical analysis approaches based on a diffusion approximation of the stochastic Lotka-Volterra model such as the one suggested by Ferm et al. [27] and compare them with the methods discussed here.

Acknowledgment

The authors wish to thank Stefan Engblom, Royal Institute of Technology in Stockholm, Sweden, for providing an implementation for the method of moments.

References

[1] R. Alur and T. Henzinger. Reactive modules. *Formal Methods in System Design*, 15:7–48, 1999.

[2] Z. Anastassi and T. Simos. A family of two-stage two-step methods for the numerical integration of the

TABLE III
COMPARISON OF THE DIFFERENT SOLUTION METHODS.

	parameters						IRK4(10^{-15})			MF		MM			SH(10^{-15},200)				
	α	β	γ	x_0	y_0	t_f	run time	$\|S\|$	error	run time	1st	run time	1st	2nd	run time	$\|S\|$	1st	2nd	3rd
1	1	0.01	1	20	20	10	51h 39min	$9e5$	$2e$-7	$< 1s$	> 1	$< 1s$	0.54	0.91	1h 43min	$5e4$	0.03	0.11	0.23
2	1	0.003	1	180	200	10	20h 56min	$8e5$	$5e$-8	$< 1s$	0.07	$< 1s$	0.07	0.27	2min 46s	$1e4$	0.05	0.11	0.16
3	1	0.1	1	5	20	10	46h 32min	$7e4$	$3e$-9	$< 1s$	> 1	$< 1s$	> 1	> 1	1h 30min	$2e4$	0.02	0.16	0.31
4	1	1	6	10	20	10	67h 43min	$5e4$	$4e$-11	$< 1s$	> 1	$< 1s$	> 1	> 1	50min 32s	$6e3$	< 0.01	0.45	0.50
5	0.5	1.3	0.67	2	1	20	56h 15min	$6e4$	$3e$-11	$< 1s$	> 1	$< 1s$	> 1	> 1	5min 39s	$1e3$	< 0.01	0.42	0.49

Schrodinger equation and related IVPs with oscillating solution. *Journal of Mathematical Chemistry*, 45(4):1102–1129, 2008.

[3] Z. Anastassi and T. Simos. A six-step p-stable trigonometrically-fitted method for the numerical integration of the radial Schrodinger equation. *MATCH Commun. Math. Comput. Chem.*, 60(3):803–830, 2008.

[4] Z. Anastassi and T. Simos. Numerical multistep methods for the efficient solution of quantum mechanics and related problems. *Phys. Rep.*, 482-483:1–240, 2009.

[5] Z. Anastassi, T. Simos, and G. Panopoulos. Two optimized symmetric eight-step implicit methods for initial-value problems with oscillating solutions. *Journal of Mathematical Chemistry*, 46(2):604–620, 2009.

[6] A. Bobbio and K. S. Trivedi. An aggregation technique for the transient analysis of stiff Markov chains. *IEEE Transactions on Computers*, C-35(9):803–814, 1986.

[7] H. Busch, W. Sandmann, and V. Wolf. A numerical aggregation algorithm for the enzyme-catalyzed substrate conversion. In *Proc, of CMSB*, volume 4210 of *LNCS*, pages 298–311. Springer, 2006.

[8] E. Çinlar. *Introduction to Stochastic Processes*. Prentice-Hall, 1975.

[9] F. Didier, T. A. Henzinger, M. Mateescu, and V. Wolf. Fast adaptive uniformization of the chemical master equation. In *Proc. of HIBI*, pages 118–127. IEEE Computer Society, 2009.

[10] S. Engblom. Computing the moments of high dimensional solutions of the master equation. *Appl. Math. Comput.*, 180:498–515, 2006.

[11] P. Erdi and J. Toth. *Mathematical Models of Chemical Reactions: Theory and Applications of Deterministic and Stochastic Models*. Manchester University Press, 1989.

[12] C. Evans and G. Findley. A new transformation for the Lotka-Volterra problem. *J. Math. Chem.*, 25:105–110, 1999.

[13] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25):2340–2361, 1977.

[14] N. S. Goel, S. C. Maitra, and E. W. Montroll. On the Volterra and other nonlinear models of interacting populations. *Rev. Mod. Phys.*, 43(2):231–276, 1971.

[15] S. E. Hitchcock. Extinction probabilities in predator-prey models. *J. Appl. Prob.*, 23(1):1–13, 1986.

[16] J. Hofbauer and K. Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998.

[17] N. G. v. Kampen. *Stochastic Processes in Physics and Chemistry.* Elsevier, 3rd edition, 2007.

[18] B. Kerr, M. A. Riley, M. W. Feldman, and B. J. M. Bohannan. Local dispersal promotes biodiversity in a real-life game of rock-paper-scissors. *Nature*, 418:171–174, 2002.

[19] B. Kirkup and M. A. Riley. Antibiotic-mediated antagonism leads to a bacterial game of rock-paper-scissors in vitro. *Nature*, 428:412–414, 2004.

[20] F. C. Klebaner. *Introduction to stochastic calculus with applications*. Imperial College Pr., 2005.

[21] T. G. Kurtz. The relationship between stochastic and deterministic models for chemical reactions. *J. Chem. Phys.*, 57(7):2976 –2978, 1972.

[22] T. G. Kurtz. Strong approximation theorems for density dependent Markov chains. *Stochastic Processes Appl.*, 6(3):223–240, 1977/78.

[23] M. Kwiatkowska, G. Norman, and D. Parker. Prism: Probabilistic model checking for performance and reliability analysis. *ACM SIGMETRICS Performance Evaluation Review*, 36(4):40–45, 2009.

[24] A. J. Lotka. *Elements of Mathematical Biology*. Williams and Wilkins Company, 1924.

[25] M. Rabinovich, R. Huerta, and G. Laurent. Transient dynamics for neural processing. *Science*, 321(5885):48–50, 2008.

[26] T. Reichenbach, M. Mobilia, and E. Frey. Coexistence versus extinction in the stochastic cyclic Lotka-Volterra model. *Phys. Rev. E*, 74:51907–51918, 2006.

[27] P. Sjöberg, P. Lötstedt, and J. Elf. Fokker-Planck approximation of the master equation in molecular biology. Technical Report 2005-044, Department of Information Technology, Uppsala University, 2005.

[28] W. J. Stewart. *Introduction to the Numerical Solution of Markov Chains.* Princeton University Press, 1995.

[29] H. Van de Vyver. A fourth-order symplectic exponentially fitted integrator. *Comput. Phys. Commun.*, 174(4):255–262, 2006.

[30] V. Volterra. Fluctuations in the abundance of a species considered mathematically. *Nature*, 118:558–560, 1926.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Finite Element Approximate Inverse Preconditioning using POSIX threads on multicore systems

G. A. Gravvanis, P. I. Matskanidis
Department of Electrical and Computer Engineering, School of Engineering, Democritus University of Thrace,
University Campus, Kimmeria, GR 67100 Xanthi, Greece
Email: ggravvan@ee.duth.gr; pascmats@ee.duth.gr

K. M. Giannoutakis
Centre for Research and Technology Hellas, Informatics and Telematics Institute, GR 57001, Thermi, Greece
Email: kgiannou@iti.gr

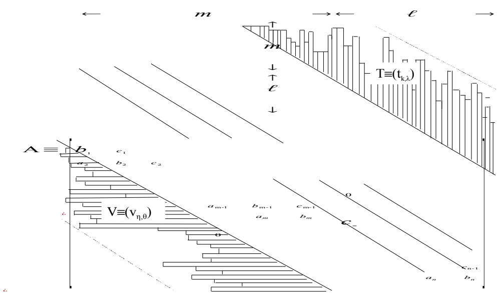
E. A. Lipitakis
Department of Informatics, Athens University of Economics and Business, 76 Patission street, GR 104 34 Athens, Greece
Email: eal@aueb.gr

Abstract—**Explicit finite element approximate inverse preconditioning methods have been extensively used for solving efficiently sparse linear systems on multiprocessor and multicomputer systems. New parallel computational techniques are proposed for the parallelization of explicit preconditioned biconjugate conjugate gradient type methods, based on Portable Operating System Interface for UniX (POSIX) Threads, for multicore systems. Parallelization is achieved by assigning every loop of the parallel explicit preconditioned bi-conjugate conjugate gradient-STAB (PEPBiCG-STAB) to the desired number of threads, thus achieving for-loop parallelization. Theoretical estimates on speedups and efficiency are also presented. Finally, numerical results for the performance of the PEPBiCG-STAB method for solving characteristic two dimensional boundary value problems on multicore computer systems are presented, which are favorably compared to corresponding results from multiprocessor systems. The implementation issues of the proposed method are also discussed using POSIX Threads on a multicore system.**

I. Introduction

LET us consider the linear system derived by the finite element (FE) method for solving boundary value problems in two dimensions, [6] ,[8], [9], [10], i.e.

$$Au = s, \quad (1)$$



(2)

where the coefficient matrix A is a non-singular large, sparse, unsymmetric, positive definite, diagonally dominant (n×n) matrix of irregular structure (where all the off-center band terms are grouped in regular bands of width ℓ at semi-bandwidth m), [2], while u is the FE solution at the nodal points and s is a vector, with components resulting from a combination of source terms and imposed boundary conditions.

During the last decades, explicit approximate inverse preconditioning methods have been extensively used for efficiently solving sparse linear systems on multiprocessor systems, [3], [4], [9], [10]. In recent years many researchers have derived preconditioners based on various techniques, which are difficult to be implemented on parallel systems, [7], [11], [12]. The effectiveness of explicit approximate inverse preconditioning schemes relies on the use of suitable preconditioners that are close approximants to the inverse of the coefficient matrix and are fast to compute in parallel, [2], [5].

The approximate inverse $M_r^{\delta l}$ represents a class of generalized approximate inverses that includes various families of approximate inverses according to the requirements of accuracy, storage and computational work, as can be seen by the following diagrammatic relation:

$$A^{-1} \equiv \overset{\underline{\text{class I}}}{M} \leftarrow \overset{\underline{\text{class II}}}{M^{\delta l}_{r=m-1}} \leftarrow \overset{\underline{\text{class III}}}{M^{\delta l}_{r}} \leftarrow M_i \quad (3)$$

where M is the exact inverse resulting in a direct method, i.e. r=m-1 and δl=n with the disadvantage of high memory requirements and computational work for large order systems. The entries of the class I inverse have been retained after the computation of the exact inverse (r=m-1, δl=n) by retaining only δl and δl-1 elements in the lower and upper part of the exact inverse. The entries of the class II inverse have been computed and retained during the computational procedure of the (approximate) inverse (r≤m-1, δl<n), while the entries of the class III of the generalized approximate inverse retains only the diagonal elements, i.e. δl=1 while requires only the the diagonal entries of the sparse lower matrix L_r, [6], [10], resulting in a fast inverse algorithm.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Furthermore, we present the **E**xplicit **P**reconditioned **BI**-conjugate **C**onjugate **G**radient-**STAB** (**EPBI-CGSTAB**) method, which can be expressed by the following compact scheme:

Let u_0 be an arbitrary initial approximation to the solution vector u. Then,

compute $$r_0 = s - Au_0, \tag{4}$$

set $$r_0^{'} = r_0,\ \rho_0 = \alpha = \omega_0 = 1 \text{ and } v_0 = p_0 = 0 \tag{5}$$

Then, for i=1,….,n (until convergence) compute the vectors u_i, r_i and the scalar quantities α, β, ω_i as follows:

calculate $$\rho_i = \left(r_0^{'}, r_{i-1}\right) \text{ and } \beta = \frac{\left(\rho_i / \rho_{i-1}\right)}{\left(\alpha / \omega_{i-1}\right)} \tag{6}$$

compute $$p_i = r_{i-1} + \beta\left(p_{i-1} - \omega_{i-1} v_{i-1}\right), \tag{7}$$

form $$y_i = \mathbf{M}_r^{\delta l} p_i \text{ and } v_i = Ay_i, \tag{8}$$

compute $$\alpha = \rho_i / \left(r_0^{'}, v_i\right) \text{ and } x_i = r_{i-1} - \alpha v_i, \tag{9}$$

form $$z_i = M_r^{\delta l} x_i \text{ and } t_i = Az_i, \tag{10}$$

set $$\omega_i = \left(\mathbf{M}_r^{\delta l} t_i,\ \mathbf{M}_r^{\delta l} x_i\right) / \left(\mathbf{M}_r^{\delta l} t_i,\ \mathbf{M}_r^{\delta l} t_i\right) \tag{11}$$

compute $$u_i = u_{i-1} + \alpha y_i + \omega_i z_i \text{ and } r_i = x_i - \omega_i t_i \tag{12}$$

Assuming that the approximate inverse $M_r^{\delta l}$ can be compactly stored in n×(2δl-1) diagonal vectors, then the computational complexity of the **EPBICG-STAB** method is ≈O[(6δl+4 ℓ +16)n mults + 6n adds]*ν* operations, where ν is the number of iterations required for convergence to a certain level of accuracy, [6], [10].

In this article, new parallel computational techniques are proposed for the parallelization of explicit preconditioned conjugate gradient type methods, based on Portable Operating System Interface for UniX (POSIX) Threads, for multicore systems. The excessive overhead produced by the template-based parallel implementations was avoided by using POSIX Threads, maximizing the overall performance of the parallel implementation of the **PEPBICG-STAB** method and throttling it close to the corresponding theoretical estimate.

Finally, numerical results for the performance of the **PEPBICG-STAB** method for solving characteristic two dimensional boundary value problems on multicore computer systems are presented, which are favorably compared to corresponding results from multiprocessor systems. The implementation issues of the proposed method are also discussed using POSIX Threads on a multicore computer system.

II. Parallel Biconjugate Conjugate Gradient-Type Method

In this section, we demonstrate the parallel implementation of the **EPBICG-STAB** method designed for multicore computer systems using POSIX threads, [1], [13].

Specifically, kernel-level threads of the POSIX 1003.1c standard are used in a thread pool pattern, [1], originally designed by Sun Microsystems for Solaris Operating System, [13], and modified for the purposes of our computations. A thread pool manages a certain number of threads that perform tasks based on a job queue. Every time a thread finishes its task it switches to idle state, waiting for another task from the queue to be assigned to it. In this way, fork and join of threads is limited to just the active threads in the pool, rather than creating and detaching every thread that performs a task. Hence, due to the reduction of latencies an improvement is expected on the parallel performance of the proposed implementation.

This parallel implementation is based on the standard for-loop parallelization model. Let *nthreads* denote the number of threads used and *thread_id* be the "id" number of each thread (from 0 to *nthreads*). Then, every thread, denoted as *localthr,* deals only with the number of allocated elements, where *localthr*=n*/nthreads.*

Hence, the **P**arallel **E**xplicit **P**reconditioned **BI**conjugate **C**onjugate **G**radient-**STAB** (**PEPBICG-STAB**) method can be expressed by the following algorithmic scheme:

Let u_0 be an arbitrary initial approximation to the solution vector u. Then,

```
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(r_0)_j = s_j - A(u_0)_j$                                  (13)
end for
thread_pool_wait
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(r_0^{'})_j = (r_0)_j$                                     (14)
    $(p_i)_j = 0.0$                                             (15)
    $(v_i)_j = 0.0$                                             (16)
end for
thread_pool_wait
    $\rho_0 = \alpha = \omega_0 = 1.0$                          (17)
    $v_0 = p_0 = 0.0$                                           (18)
```

Then, **for i=1, ..., (until convergence)** compute in parallel the vectors u_i, r_i and the scalar quantities α, β, ω_i as follows:

```
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
do reduction + $\rho_i$
    $\rho_i = (r_0^{'})_j \cdot (r_{i-1})_j$                    (19)
end for
thread_pool_wait
    $\beta = (\rho_i / \rho_{i-1}) / (\alpha / \omega_{i-1})$   (20)
thread_pool_queue
```

```
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(p_i)_j = (r_{i-1})_j + \beta((p_{i-1})_j - \omega_{i-1}(v_{i-1})_j)$   (21)
end for
thread_pool_wait
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(y_i)_j = \left(\sum_{k=\max(1,j-\delta l+1)}^{\min(n,j+\delta l-1)} \mu_{j,k}(p_i)_k\right)$   (22)
end for
thread_pool_wait
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(v_i)_j = A(y_i)_j$   (23)
end for
thread_pool_wait
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
do reduction + $d_i$
    $d_i = (r'_0)_j \cdot (v_i)_j$   (24)
end for
thread_pool_wait
    $\alpha = \rho_i / d_i$   (25)
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(x_i)_j = (r_{i-1})_j - \alpha(v_i)_j$   (26)
end for
thread_pool_wait
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(z_i)_j = \left(\sum_{k=\max(1,j-\delta l+1)}^{\min(n,j+\delta l-1)} \mu_{j,k}(x_i)_k\right)$   (27)
end for
thread_pool_wait
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(t_i)_j = A(z_i)_j$   (28)
end for
thread_pool_wait
    $a_1 = 0.0$   (29)
    $a_2 = 0.0$   (30)
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
do reduction + $a_1$ , $a_2$
    $(tt_i)_j = \left(\sum_{k=\max(1,j-\delta l+1)}^{\min(n,j+\delta l-1)} \mu_{j,k}(t_i)_k\right)$   (31)
    $a_1 = (tt_i)_j \cdot (z_i)_j$   (32)
    $a_2 = (tt_i)_j \cdot (tt_i)_j$   (33)
end for
thread_pool_wait
    $\omega_i = a_1 / a_2$   (34)
thread_pool_queue
for j=(1+thread_id*localthr) to (thread_id+1)*localthr
    $(u_i)_j = (u_{i-1})_j + \alpha(y_i)_j + \omega_i(z_i)_j$   (35)
    $(r_i)_j = (x_i)_j - \omega_i(t_i)_j$   (36)
end for
thread_pool_wait
```

The computational complexity of the **PEPBICG-STAB** method is ≈O[(6δl+4 ℓ +16)*localthr* mults + 6*localthr* adds] ν operations, where ν denotes the number of iterations required for convergence to a certain level of accuracy and *localthr* is the number of rows of the matrices distributed onto each thread, [2], [3], [5].

Thus, the theoretical speedup and efficiency of the **PEPBICG-STAB** methods are respectively:

$$S_p^{\delta l} = \frac{1}{\dfrac{1}{\text{nthreads}} + \dfrac{12t_l}{O(6\delta l+4\ell+16)n\cdot t_m}} \qquad (37)$$

and

$$E_p^{\delta l} = \frac{1}{1 + \dfrac{12t_l\cdot \text{nthreads}}{O(6\delta l+4\ell+16)n\cdot t_m}} \qquad (38)$$

where t_m denotes the computational time of a multiplication, while t_l is the threads assignment latencies. It is obvious that for δl→ ∞ , then $S_p^{\delta l}$ →nthreads and $E_p^{\delta l}$ →1, (37)-(38).

The effectiveness of the parallel explicit preconditioned conjugate gradient type method is related to the fact that the approximate inverse × vector and vector × vector can be efficiently implemented on parallel systems, [1], [2], [5], [13].

III. Numerical Results

In this section we examine the effectiveness and applicability of the new proposed parallel schemes for solving characteristic two dimensional boundary value problems, on multicore computer systems, using POSIX threads, [1], [13]. The numerical test runs were performed on a Dual 2x Intel Xeon server at 2.0 GHz with 12MB cache memory, 8GB RAM and 1333MHz bus, running Debian GNU/Linux (University College at Cork, Ireland).

Let us consider the following 2D-boundary value problem:

$$\Delta u(x,y)+u(x,y) = f(x,y), \quad (x,y) \in R \qquad (39)$$

$$u(x,y)=0, \quad (x,y) \in \partial R, \qquad (39.a)$$

Table I

The convergence behavior of the **PEPBICG-STAB** method for several values of n, m, and δl with number of threads=1.

n	m	Number of iterations of PEPBICG-STAB method				
		“Retention” parameter δl				
		δl=1	δl=m/2	δl=m	δl=2m	δl=4m
62500	251	13	11	8	5	5
122500	351	13	11	8	5	5
160000	401	13	11	8	5	5
202500	451	14	11	9	5	5

Table II

The performance of the **PEPBICG-STAB** method for several values of the number of threads, n and δl.

No of Threads	n	m	δl=1	δl=m/2	δl=m	δl=2m	δl=4m
1	62500	251	0.601	18.043	23.056	30.183	61.502
2			0.332	9.246	11.726	15.328	31.090
4			0.189	4.669	5.963	7.787	15.820
8			0.157	2.426	3.043	3.919	7.968
1	122500	351	1.184	42.828	55.145	69.803	141.597
2			0.677	21.798	27.914	35.332	71.496
4			0.333	11.042	14.164	17.915	36.294
8			0.297	5.598	7.173	9.048	18.332
1	160000	401	1.554	57.764	77.395	97.981	198.367
2			0.856	29.332	39.193	49.396	99.805
4			0.434	14.852	19.859	25.119	50.762
8			0.384	7.531	10.057	12.694	25.682
1	202500	451	2.139	79.866	118.654	132.285	267.750
2			1.000	40.398	53.243	66.560	134.712
4			0.596	20.479	27.013	33.855	68.517
8			0.488	10.357	13.662	17.117	34.647

Table III

Speedups of the **PEPBICG-STAB** method for several values of the number of threads, n, m and δl.

No of Threads	n	m	δl=1	δl=m/2	δl=m	δl=2m	δl=4m
2	62500	251	1.810	1.951	1.966	1.969	1.978
4			3.180	3.864	3.867	3.876	3.888
8			3.828	7.437	7.577	7.702	7.719
2	122500	351	1.883	1.965	1.976	1.976	1.980
4			3.282	3.879	3.893	3.896	3.901
8			3.987	7.651	7.688	7.715	7.724
2	160000	401	1.955	1.969	1.975	1.984	1.988
4			3.305	3.889	3.897	3.901	3.908
8			4.047	7.670	7.696	7.719	7.724
2	202500	451	1.986	1.977	1.981	1.987	1.988
4			3.333	3.900	3.904	3.907	3.908
8			4.070	7.711	7.720	7.728	7.728

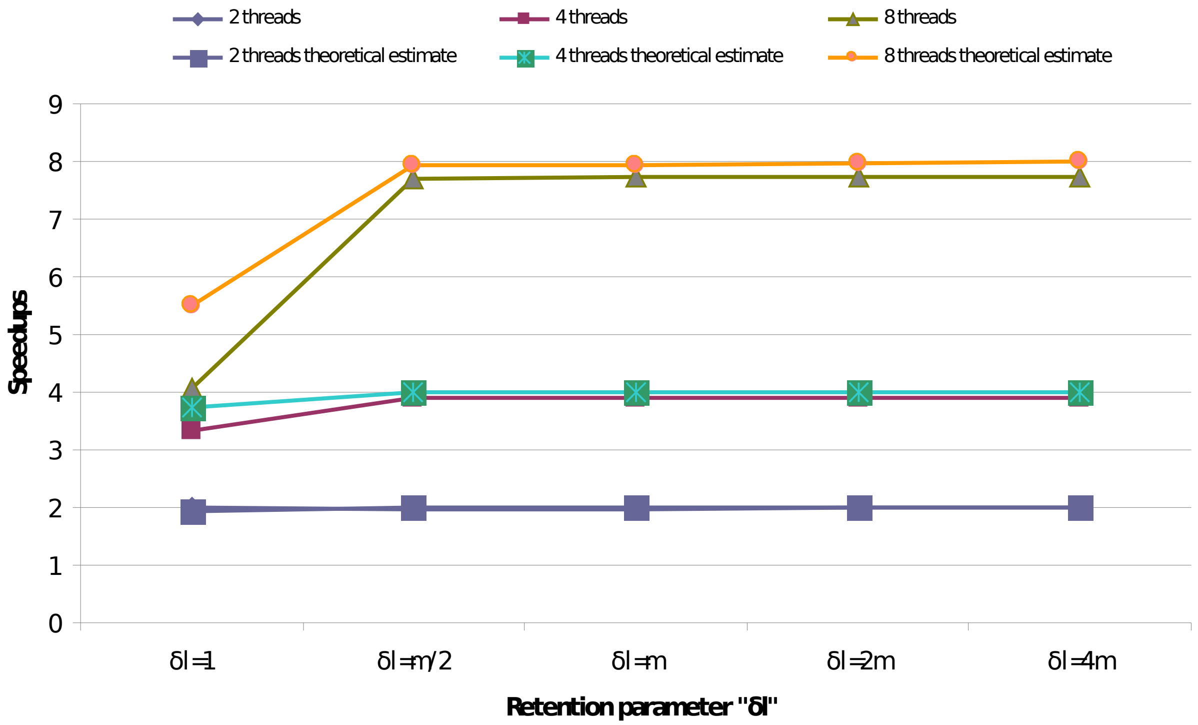


Fig 1. Speedups versus the retention parameter δl for the **PEPBICG-STAB** method, along with theoretical upper bounds, for n=202500.

where Δ is the Laplace operator, R is the unit square and ∂R denotes the boundary of R. The domain R∪∂R was covered by a non-overlapping triangular network resulting in a hexagonal mesh. The right hand side vector of the sparse linear system (1) was computed as the product of the coefficient matrix A by the solution vector, with its components equal to unity. The "width" parameter was set to $\ell=3$ and the "fill-in" parameter to r=2. The iterative process was terminated when $\|u_{i+1}-u_i\|_\infty < 10^{-5}$, [6], [8], [10].

The convergence behavior and the performance, given in "seconds.hundreds", of the **PEPBICG-STAB** method for several values of the order n, semi-bandwidth m and "retention" parameter δl is presented in Table I and Table II respectively. The speedups and number of threads allocated of the **PEPBICG-STAB** method for several values of the order n, semi-bandwidth m and "retention" parameter δl are given in Table III.

In Figure 1 the speedups and number of threads allocated for several values of δl along with theoretical estimates, are presented for the **PEPBICG-STAB** method.

It should be mentioned that the convergence behavior of the explicit preconditioned biconjugate conjugate gradient method was found to be in qualitative agreement with similar theoretical estimates obtained, [6], [10]. Additionally, when the "retention" parameter δl=1 the speedups and the efficiency are significantly improved, while for larger values of the "retention" parameter, i.e. multiples of the semi-bandwidth m, the speedups and the efficiency are slightly improved in comparison with results obtained on a multiprocessor system using OpenMP, [2].

Finally, it is pointed out that for large values of the "retention" parameter δl the speedups and the efficiency tend to the upper theoretical bound, for the parallel explicit preconditioned biconjugate conjugate gradient method, since the coarse granularity amortizes the parallelization overheads.

IV. Acknowledgment

The author would like to express his thanks to Dr. John Morrison of the Department of Computer Science, University College of Cork, for the provision of suitable computational facilities and support through the WebCom-G project funded by Science Foundation Ireland. The author wishes to acknowledge the SFI/HEA Boole Centre for Research in Informatics and the Centre for Unified Computing for the provision of computational facilities and support. Part of this work is partially funded by Science Foundation Ireland and the Higher Education Authority under the Cosmogrid project.

References

[1] D. R. Butenhof, *Programming with POSIX® Threads*, Addison-Wesley, 1997.

[2] K. M. Giannoutakis, and G. A. Gravvanis, "High performance finite element approximate inverse preconditioning," *Applied Mathematics and Computation*, vol. 201, pp. 293–304, 2008.

[3] G. A. Gravvanis, "High Performance Inverse Preconditioning," *Archives of Computational Methods in Engineering*, vol. 16, no. 1, pp. 77-108, 2009.

[4] G. A. Gravvanis, “Explicit Approximate Inverse Preconditioning Techniques,” *Archives of Computational Methods in Engineering*, vol. 9, no. 4, pp. 371-402, 2002.

[5] G. A. Gravvanis, and K. M. Giannoutakis, “Fast parallel finite element approximate inverses,” *Computer Modeling in Engineering and Sciences*, vol. 32, no. 1, pp. 35-44, 2008.

[6] G. A. Gravvanis, and E. A. Lipitakis, “An explicit sparse unsymmetric finite element solver,” *Commun. Numer. Meth. in Engin.*, vol. 12, pp. 21-29, 1996.

[7] M. J. Grote, and T. Huckle, “Parallel preconditioning with sparse approximate inverses,” *SIAM J. Sci. Computing*, vol. 18, pp. 838-853, 1997.

[8] E. A. Lipitakis, “Generalized extended to the limit sparse factorization techniques for solving large unsymmetric finite elements systems,” *Computing*, vol. 32, pp. 255-270, 1984.

[9] E. A. Lipitakis, and D. J. Evans, “Explicit semi-direct methods based on approximate inverse matrix techniques for solving boundary-value problems on parallel processors,” *Math. and Computers in Simulation*, vol. 29, pp. 1-17, 1987.

[10] E. A. Lipitakis, and G. A. Gravvanis, “Explicit preconditioned iterative methods for solving large unsymmetric FE systems,” *Computing*, vol. 54, pp. 167-183, 1995.

[11] Y. Saad, and H. A. van der Vorst, “Iterative solution of linear systems in the 20th century,” *J. Comp. Applied Math*., vol. 123, pp. 1-33, 2000.

[12] Y. Saad, *Iterative methods for sparse linear systems*, PWS Publishing, 1996.

[13] Sun Microsystems, “Multithreaded Programming Guide” (http://dlc.sun.com/pdf/816-5137/816-5137.pdf), 2008.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 303–308

ISBN 978-83-60810-27-9
ISSN 1896-7094

On the implementation of public keys algorithms based on algebraic graphs over finite commutative rings

Michał Klisowski
Maria Curie-Sklodowska University,
Institute of Mathematics,
pl. M. Curie-Skłodowskiej 1, 20-031 Lublin, Poland
Email: mklisow@hektor.umcs.lublin.pl

Vasyl Ustimenko
Maria Curie-Sklodowska University,
Institute of Mathematics,
pl. M. Curie-Skłodowskiej 1, 20-031 Lublin, Poland
Email: vasyl@hektor.umcs.lublin.pl

***Abstract*—We will consider** balanced **directed graphs, i.e., graphs of binary relations, for which the number of inputs and number of outputs are the same for each vertex. The commutative diagram is formed by two directed paths for which the same starting and ending points form the full list of common vertices. We refer to the length of the maximal path (the number of arrows) as the rank of the diagram. We will count a directed cycle of length m as a commutative diagram of rank m. We define the** girth indicator gi, gi ≥ 2 **of the directed graph as the minimal rank of its commutative diagram.**

We observe briefly the applications of finite automata related to balanced graphs of high girth in Cryptography. Finally, for each finite commutative ring K with more than two regular elements we consider the explicit construction of algebraic over K family of graphs of high girth and discuss the implementation of the public key algorithm based on finite automata corresponding to members of the family.

I. Introduction

CLASSICAL problems on Turan type problems on studies of the maximal size of simple graphs without prohibited cycles are attractive for mathematicians because they are beautiful and difficult (see [1], [9]). The concept of a family of simple graphs of large girth appears as an important tool to study such problems. Later the applications of these problems in Networking [2], Coding Theory and Cryptography were found (see [15 and further references]).

Section 2 is devoted to the concept of the girth indicator and the family of large girth for digraphs.

In Section 3 we consider the definition of a family of affine algebraic digraphs of large girth over commutative rings. Explicit constructions of such families of graphs can be used for the development of public keys and a key exchange protocol. We discuss the connection of these algorithms with the group theoretical discrete logarithm problem.

The known examples of families of simple algebraic graphs were constructed just in the case of finite fields (see [5]). In section 4 we consider an explicit construction of a family of affine algebraic digraphs of large girth over each finite commutative ring containing at least 3 regular elements. Different properties of this family are investigated in [14], [15], [17] [18] , [20], [7], [8].

Section 5 is devoted to the latest implementation of the public key algorithm based on one of the family described in section 4.

II. On the families of directed graphs of large girth

The missing theoretical definitions on directed graphs the reader can find in [6]. Let Φ be an irreflexive binary relation over the set V, i.e., $\Phi \in V \times V$ and for each v the pair (v, v) is not the element of Φ.

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 Γ of irreflexive binary relation ϕ 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 Γ is k-regular if for each vertex $v \in \Gamma$ the cardinality of $\{x|(v, x) \in \phi\}$ is k.

Let Γ be the graph of binary relation. The *path* between vertices a and b is the sequence $a = x_0 \rightarrow x_1 \rightarrow \dots x_s = b$ of length s, where x_i, $i = 0, 1, \dots s$ are distinct vertices.

We say that the pair of paths $a = x_0 \rightarrow x_1 \rightarrow \cdots \rightarrow x_s = b$, $s \geq 1$ and $a = y_0 \rightarrow y_1 \rightarrow \cdots \rightarrow y_t = b$, $t \geq 1$ 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 ≥ 2, because the graph does not contain multiple edges.

Notice that the graph of antireflexive binary relation may have a directed cycle $O_s = O_{s,0}$: $v_0 \rightarrow v_1 \rightarrow \dots v_{s-1} \rightarrow v_0$, 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 *girth indicator* gi, which is the minimal value for $\max(s, t)$ for parameters s, t of a ommutative 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 $<$ gi are connected by the unique path from u to v of length $<$ gi.

We assume that the *girth* $g(\Gamma)$ of a directed graph Γ 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 $\mathrm{gi} = d$ implies that the girth of the graph is $2d$ or $2d-1$. It does not contain an even cycle $2d-2$. In general case $\mathrm{gi} = d$ 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 $\mathrm{gi} \geq g/2$.

In the case of symmetric irreflexive relations the above mentioned general definition of the girth agrees with the standard definition of the girth of 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 Γ_i of degree k_i and order v_i such that $\mathrm{gi}(\Gamma_i)$ is $\geq c\log_{k_i} v_i$, where c' is a constant independent of i.

As it follows from the definition $g(\Gamma_i) \geq c'\log_{k_i}(v_i)$ 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 [6] is the minimal length d of the shortest directed path $a = x_0 \to x_1 \to x_2 \cdots \to x_d$ between two vertices a and b. Recall that a graph is k-regular, if each vertex of G has exactly k outputs. 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 $d_i \leq C\log_{k_i}(v_i)$, $i = 1, \ldots$ for some constant C independent on i. The definition of small world simple graphs and related explicit constructions the reader can find in [3]. For the studies of small world simple graphs without small cycles see [9], [20] and [33].

III. On the K-theory of affine graphs of high girth and its cryptographical motivations

Let K be a commutative ring. A *directed algebraic graph* ϕ 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 ϕ in $Q \times Q$. We assume that ($x\phi y$ means $(x, y) \in \phi$).

The graph ϕ is *balanced* if for each vertex $v \in Q$ the sets $\mathrm{Im}(v) = \{x \mid v\phi x\}$ and $\mathrm{Out}(v) = \{x \mid x\phi v\}$ are quasiprojective varieties over K of the same dimension.

The graph ϕ is *homogeneous* (or (r, s)-homogeneous) if for each vertex $v \in Q$ the sets $\mathrm{Im}(v) = \{x|v\phi x\}$ and $\mathrm{Out}(v) = \{x|x\phi v\}$ 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, *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 $\mathrm{Reg}(K)$ 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 the $\mathrm{Reg}(K)$ 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 Zarisski topology.

Let G be r-regular affine graph with the vertex $V(G)$, such that 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 ρ from $E(G)$ onto the set $R(K)$ (the set of colours) if for each vertex v different output arrows $e_1 \in \mathrm{Out}(v)$ and $e_2 \in \mathrm{Out}(v)$ have distinct colours $\rho(e_1)$ and $\rho(e_2)$ and the operator $N_\alpha(v)$ of taking the neighbour u of vertex v ($v \to u$) 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 $\mathrm{dirg}(G)$ be a directed girth of the graph G, i.e., the minimal length of a directed cycle in the graph. Obviously $\mathrm{gi}(G) \leq \mathrm{dirg}(G)$.

Studies of infinite families of directed affine algebraic digraphs over commutative rings K of large girth with the rainbow-like colouring is a nice and a difficult mathematical problem. Good news is that such families do exist. In the next section we consider the example of such a family for each commutative ring with more than 2 regular elements.

Here, 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 group G is about finding a solution for the equation $g^x = b$ where x is unknown positive number. If the order $|g| = n$ is known we can replace G on a cyclic group C_n. So we may assume that the order of g is sufficiently large to make unfeasible the computation of n. 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 automorphism 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 $G_i(K)$, $i = 1, 2, \ldots$ 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 α are elements of $C_i = C(V_i(K))$. We can chose a relatively small number k to generate $h = h_i = N_{\alpha_1} N_{\alpha_2} \ldots N_{\alpha_k}$ in each group C_i, $i = 1, 2, \ldots$

Let us assume that the family of graphs $G_i(K)$ is the family of graphs of large girth. It means that the girth indicator $\mathrm{gi}_i = \mathrm{gi}(G_i(K))$ and the parameter $\mathrm{dirg}_i = \mathrm{dirg}(G_i(K))$ are growing with the growth of i. Notice that $|h_i|$ is bounded below by dirg_i/k. So there is j such that for $i \geq j$ the computation of $|h_i|$ is impossible. Finally we can take the base $g = u^{-1}h_j u$ where u 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

$x_1 \rightarrow g_1(x_1, x_2, \ldots, x_n)$, $x_2 \rightarrow g_2(x_1, x_2, \ldots, x_n)$, ..., $x_n \rightarrow g_n(x_1, x_2, \ldots, x_n)$.

The symbolic map g can be used for Diffie - Hellman *key exchange protocol* (see [3] for the details). Let Alice and Bob be correspondents. Alice computes the symbolic map g and send it to Bob via 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 were cooked) , so he and Cezar are making computations much slower than Alice who has the decomposition $g = u^{-1} N_{\alpha_1} N_{\alpha_2} \ldots N_{\alpha_k} u$.

We may modify slightly the Diffie - Hellman protocol using the action of the group on the variety. Alice chooses a rather short password $\alpha_1, \alpha_2, \ldots, \alpha_k$, 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 is getting $(g^{n_B})(v_A)$.

The common information is the vertex $g^{n_A \times n_B}(v)$.

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 $\alpha_1, \alpha_2, \ldots, \alpha_t$ 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 $x_1, x_2, \ldots, x_n$. Alice presents the map in the form of public rules, such as

$x_1 \rightarrow f_1(x_1, x_2, \ldots, x_n)$, $x_2 \rightarrow f_2(x_1, x_2, \ldots, x_n)$, ..., $x_n \rightarrow f_n(x_1, x_2, \ldots, x_n)$.

We can assume (at least theoretically) that the public rule depending on parameter j is applicable to encryption of potentially infinite text (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 u^{-1}, maps of kind N_α and u 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 of taking the neighbour of vertex x in $G_j(K)^{-1}$ of colour α. Then Alice can decrypt applying consequently $u^{-1}, N'_{\alpha_t}, N'_{\alpha_{t-1}}, \ldots, N_{\alpha_1}$ and u 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 $\alpha_1, \alpha_2, \ldots, \alpha_{t_j}$ has the period k and the transformation $h = u^{-1} N_{\alpha_1} N_{\alpha_2} \ldots N_{\alpha_k} u$ 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 related Cremona group of all polynomial bijective transformations.

Of course for further cryptoanalysis we need to study the information on possible divisors of order of the base of 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 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 that case different keys produce distinct ciphertexts from the chosen plaintext. In that case we prove that if the adversary has no access to plaintexts then he can break the encryption via the brut-force search via all keys from the key space. The encryption map has no fixed points.

IV. On the family of affine digraph of large girth over commutative rings

E. Moore used 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 ϕ 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 parenthesis 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

$(l_1, p_1) \rightarrow [l_2, p_2]$ if and only if $p_1 = p_2$ and $l_1 \neq l_2$,

$[l_2, p_2] \rightarrow (l_1, p_1)$ if and only if $l_1 = l_2$ and $p_1 \neq p_2$.

Below we consider the family of graphs $D(k, K)$, 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 [5]).

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 [16] for the case of general commutative ring K:

$$\begin{aligned}(p) &= (p_{0,1}, p_{1,1}, p_{1,2}, p_{2,1}, p_{2,2}, p'_{2,2}, p_{2,3}, \ldots, \\ &\quad p_{i,i}, p'_{i,i}, p_{i,i+1}, p_{i+1,i}, \ldots), \\ [l] &= [l_{1,0}, l_{1,1}, l_{1,2}, l_{2,1}, l_{2,2}, l'_{2,2}, l_{2,3}, \ldots, \\ &\quad l_{i,i}, l'_{i,i}, l_{i,i+1}, l_{i+1,i}, \ldots].\end{aligned}$$

The elements of P and L can be thought 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 coordinates hold:

$$\begin{aligned}l_{i,i} - p_{i,i} &= l_{1,0}p_{i-1,i} \\ l'_{i,i} - p'_{i,i} &= l_{i,i-1}p_{0,1} \\ l_{i,i+1} - p_{i,i+1} &= l_{i,i}p_{0,1} \\ l_{i+1,i} - p_{i+1,i} &= l_{1,0}p'_{i,i}\end{aligned}$$

(These four relations are defined for $i \geq 1$, $p'_{1,1} = p_{1,1}$, $l'_{1,1} = l_{1,1}$). This incidence structure (P, L, I) we denote as $D(K)$. 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 $D(k, K)$.

For each positive integer $k \geq 2$ we consider the *standard* graph homomorphism ϕ_k of (P_k, L_k, I_k) onto $(P_{k-1}, L_{k-1}, I_{k-1})$ defined 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 $DE_n(K)$ $(DE(K))$ be the double directed graph of the bipartite graph $D(n, K)$ $(D(K)$, 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 $l^1_{1,0} - l^2_{1,0}$.

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 $p^1_{1,0} - p^2_{1,0}$. It is easy to see that ρ is a perfect algebraic colouring.

If K is finite, then the cardinality of the colour set is $(|K| - 1)$. Let Reg$K$ be the totality of regular elements, i.e., not zero divisors. Let us delete all arrows with colour, which is a zero divisor. We will show that a new graph $RE_n(K)$ $(RE(K))$ with the induced colouring into colours from the alphabet $\mathrm{Reg}(K)$ is vertex transitive. Really, according to [31] graph $D(n, K)$ is an edge transitive. This fact had been established via the description of regular on the edge set subgroup $U(n, K)$ of the automorphisms group $\mathrm{Aut}(G)$. The vertex set for the graph $DE_n(K)$ consists of two copies F_1 and F_2 of the edge set for $D(n, K)$. It means that Group $U(n, K)$ acts regularly on each set F_i, $i = 1, 2$. An explicit description of generators for $U(n, K)$ implicates that this group is a colour preserving group for the graph $DE_n(K)$ with the above colouring.

If K is finite, then the cardinality of the colour set is $(|K| - 1)$. Let Reg$K$ be the totality of regular elements, i.e., nonzero divisors. Let us delete all arrows with colour, which is a zero divisor. We can show that a new affine graph $RE_n(K)$ $(RE(K))$ with the induced colouring into colours from the alphabet $\mathrm{Reg}(K)$ is vertex transitive (see [18]).

V. On the implementation of the public key algorithm based on $RE(t, K)$

The graphs $CRE_n(K)$ have the best known speed of growth of the girth indicator evaluated in the previous section. It turns out that for the computer implementation of the public key algorithm described in the section 4 the family $RE_n(K)$ of "enveloping" for $CRE_n(K)$ graphs were chosen first. It is also a family of digraphs of large girth but the speed of the growth of girth indicator for the family is less of those for $RE_n(K)$. Graphs $RE_n(K)$ were defined via the family of graphs $D(n, K)$ in the way described in the previous section. So, in some publications the description of the algorithm was done in terms of $D(n, K)$. We introduced here a speed evaluation of the algorithm for its latest implementation.

The set of vertices of the graph $RE_n(K)$ is a union of two copies 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 π. 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 $\mathrm{x} \rightarrow \mathrm{x}A + \mathrm{b}$, where $\mathrm{x} = (x_1, x_2, \ldots, x_{n+1}) \in C(K^{n+1})$, $\mathrm{b} = (b_1, b_2, \ldots, b_{n+1})$ is a chosen vector from $C(K^{n+1})$ and A is a matrix of a linear invertible transformation of K^{n+1}.

Graph $RE_n(K)$ is a bipartite directed graph. We assume that the plaintext K^{n+1} is a point $(p_1, p_2, \ldots, p_{n+1})$. We choose two affine transformations T_1 and T_2 and a linear transformation u will be of kind $p_1 \rightarrow p_1 + a_1p_2 + a_3p_3 + \cdots + a_{n+1}$. We slightly modify a general scheme, so Alice computes symbolically of chosen T_1 and T_2, chooses a string $(\beta_1, \beta_2, \ldots, \beta_l)$ of colours for $RE_n(K)$, such that $\beta_i \neq -\beta_{i+1}$ for $i = 1, 2, \ldots, l-1$. She computes $N_l = N_{\beta_1} \times N_{\beta_2} \cdots \times N_{\beta_l}$.

Recall that N_α, $\alpha \in \mathrm{Reg}(K)$ is the operator of taking the neighbour of the vertex v alongside the arrow with the colour α in the graph $RE_n(K)$.

Alice keeps chosen parameters secret and computes the public rule g as the symbolic composition of T_1, N and T_2. The case $uT_2 = {T_1}^{-1}$ is a special form of the general algorithm considered in chapter 4.

In case $K = F_q$, $q = 2^n$ this public key rule has a certain similarity to the Imai-Matsomoto public rule, which is computed as a composition T_1ET_2 of two linear transformations T_1 and T_2 of the vector space $F_{2^n F_{2^s}}$, where F_{2^s} is a special subfield, and E is a special Frobenius automorphism of F_{2^n}. The public rule corresponding to T_1ET_2 is a quadratic polynomial map (see [3] for the detailed description of the algorithm, its cryptoanalisis and generalisations by J. Patarin)

In the case of $RE_n(K)$ the degree of transformation N_l is 3, independently on the choice of length l [21]. So the public rule is a cubical polynomial map of the free module K^{n+1} onto itself. In case of a finite field the algorithm is equivalent to the public rule considered in [19.]

In our computer implementations we used T_1 and T_2 of kind $p_1 \to p_1 + a_1p_2 + a_3p_3 + \cdots + a_{n+1}p_{n+1}$.

A. Time evaluation of the private key algorithm for Alice

Alice can use numerical algorithms for the encryption. The decryption ${T_2}^{-1}N_{-\alpha_l}N_{-\alpha_{l-1}}\dots N_{-\alpha_1}{T_1}^{-1}$ takes the same time as the encryption.

In [4] we have implemented a computer application, which uses the family of graphs $RDE(n, K)$ for *private key* cryptography. To achieve high speed, we will use commutative rings $K = Z_{2^k}$, $k \in \{8, 16, 32\}$, with operations $+, \times$ modulo 2^k. The parameter n stands for the length of a plaintext (input data) and the length of a ciphertext. Later on we use a loaded multiplication tables for finite fields and implement cases of finite fields $K = F_{2^k}$, $k \in \{8, 16, 32\}$. We denote by $G1$ the algorithm with $k = 8$, by $G2$ the algorithm with $k = 16$, and by $G4$ the algorithm with $k = 32$. So $Gi, i \in 1, 2, 4$ denotes the number of bytes used in the alphabet (and the size of 1 character in the string).

All the tests were run on a computer with parameters:

- AMD Athlon 1.46 GHz processor
- 1 GB RAM memory
- Windows XP operating system.

The program is written in Java language. Well known algorithms RC4 which is used for comparison is taken from Java standard library for cryptography purposes - *javax.crypto*.

RC4 is a well known and widely used stream cipher algorithm. Protocols SSL (to protect Internet traffic) and WEP (to secure wireless networks) use RC4 as an option. Nowadays RC4 is not secure enough. Anyway we chose it for comparison, because of its popularity and high speed.

RC4 is not dependent on the password length in terms of complexity, and our algorithm is. Longer password makes us do more steps between vertices of graph. So for fair comparison we have used a fixed password length equal suggested upper bound for RC4 (16 Bytes).

TABLE II
TIME OF ENCRYPTION

	$w = 1$ ($\mathbb{Z}_{2^8}$)	$w = 2$ ($\mathbb{Z}_{2^{16}}$)	$w = 4$ ($\mathbb{Z}_{2^{32}}$)
$n = 20$	16	0	0
$n = 40$	265	47	15
$n = 60$	1375	188	15
$n = 80$	3985	578	47
$n = 100$	10078	1360	125

The speed of execution of the above implementation compares well with implementations of other graph based symmetric encryption algorithms [10], [12], [16].

B. On the time evaluation for the public rule

Recall, that we combine a graph transformation N_l with two affine transformation T_1 and T_2. Alice can use $T_1N_lT_2$ for the construction of the following public map of

$$y = (F_1(x_1, \dots, x_n), \dots, F_n(x_1, \dots, x_n))$$

$F_i(x_1, \dots, x_n)$ are polynomials of n variables written as the sums of monomials of kind $x_{i+1} \dots x_{i_3}$, where $i_1, i_2, i_3 \in 1, 2, \dots, n_1$ with the coefficients from $K = F_q$. As we mention before the polynomial equations $y_i = F_i(x_1, x_2, \dots, x_n)$, which are made public, have the degree 3. Hence the process of an encryption and a decryption can be done in polynomial time $O(n^4)$ (in one y_i, $i = 1, 2 \dots, n$ there are $2(n^3 - 1)$ additions and multiplications). But the cryptoanalyst Cezar, having only a formula for y, has a very hard task to solve the system of n equations of n variables of degree 3. It is solvable in exponential time $O(3^{n^4})$ by the general algorithm based on Gröbner basis method. Anyway studies of specific features of our polynomials could lead to effective cryptoanalysis. This is an open problem for specialists.

We have written a program for generating a public key and for encrypting text using the generated public key. The program is written in C++ and compiled with the Borland bcc 5.5.1 compiler.

We use a matrix in which all diagonal elements equal 1, elements in the first row are non-zero and all other elements are zero as A, identity matrix as B and null vectors as c and d. In such a case the cost of executing affine transformations is linear.

The table **??** presents the time (in milliseconds) of the generation of the public key depending on the number of variables (n) and the password length (p).

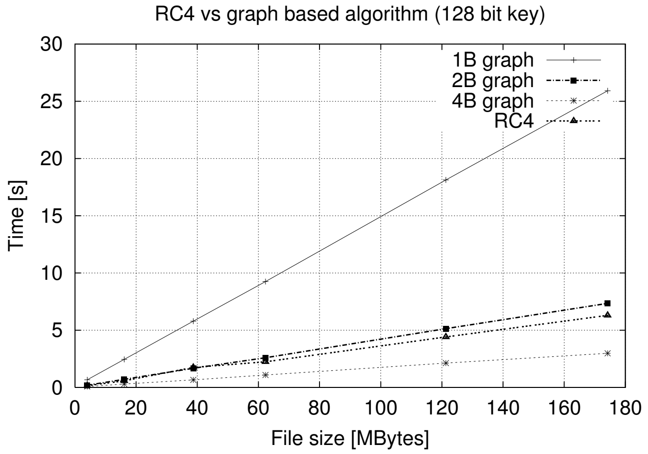
The table **??** presents the time (in milliseconds) of encryption process depending on the number of bytes in plaintext (n) and the number of bytes in a character (w).

References

[1] B. Bollobás, *Extremal Graph Theory*, Academic Press, London, 1978.

[2] F. Bien, *Constructions of telephone networks by group representations*, Notices Amer. Mah. Soc., 36 (1989), 5-22.

[3] N. Koblitz, *Algebraic aspects of Cryptography*, in Algorithms and Computations in Mathematics, v. 3, Springer, 1998.



File [MB]	RC4 [s]	G1 [s]	G2 [s]	G4 [s]
4.0	0.15	0.67	0.19	0.08
16.1	0.58	2.45	0.71	0.30
38.7	1.75	5.79	1.68	0.66
62.3	2.24	9.25	2.60	1.09
121.3	4.41	18.13	5.14	2.13
174.2	6.30	25.92	7.35	2.98

Fig. 1. RC4 vs high girth graph based algorithm (128 bit password)

TABLE I
TIME OF PUBLIC KEY GENERATION

	$p = 10$	$p = 20$	$p = 30$	$p = 40$	$p = 50$	$p = 60$
$n = 10$	15	15	16	32	31	32
$n = 20$	109	250	391	531	687	843
$n = 30$	609	1484	2468	3406	4469	5610
$n = 40$	2219	7391	12828	18219	24484	29625
$n = 50$	5500	17874	34078	49952	66749	82328
$n = 60$	12203	42625	87922	138906	192843	242734
$n = 70$	22734	81453	169250	286188	405500	536641
$n = 80$	46015	165875	350641	619921	911781	1202375
$n = 90$	92125	332641	708859	1262938	1894657	2525360
$n = 100$	159250	587282	1282610	2220610	3505532	4899657

[4] S. Kotorowicz, V. Ustimenko, *On the implementation of cryptoalgorithms based on algebraic graphs over some commutative rings*, Condensed Matter Physics, 2008, vol. 11, No. 2(54), (2008) 347–360.

[5] F. Lazebnik, V. A. Ustimenko and A. J. Woldar, *A New Series of Dense Graphs of High Girth*, Bull (New Series) of AMS, v.32, N1, (1995), 73-79.

[6] R. Ore, *Graph Theory*, Wiley, London, 1971.

[7] T. Shaska, V. Ustimenko, *On the homogeneous algebraic graphs of large girth and their applications*, Linear Algebra and its Applications Article, Volume 430, Issue 7, 1 April 2009, Special Issue in Honor of Thomas J. Laffey.

[8] T. Shaska and V. Ustimenko, *On some applications of graph theory to cryptography and turbocoding*, Special issue of Albanian Journal of Mathematics:Proceedings of the NATO Advanced Studies Institute "New challenges in digital communications", May 2008, University of Vlora, 2008, v.2, issue 3, 249-255.

[9] M. Simonovits *Extremal Graph Theory*, Selected Topics in Graph Theory 2 (L.W. Beineke and R.J. Wilson, eds), Academic Press, London, 1983, 161-200.

[10] A. Touzene, V. Ustimenko, *Private and Public Key Systems Using Graphs of High Girth*, in Roland E. Chen (editor), Cryptography Research Perspective, Nova Science Publishers, April, 2009.

[11] V. Ustimenko, *Graphs with Special Arcs and Cryptography*, Acta Applicandae Mathematicae, 2002, vol. 74, N2, 117-153.

[12] V. Ustimenko, *CRYPTIM: Graphs as tools for symmetric encryption*, In Lecture Notes in Comput. Sci., 2227, Springer, New York, 2001.

[13] V. A. Ustimenko, *Linguistic Dynamical Systems, Graphs of Large Girth and Cryptography*, Journal of Mathematical Sciences, Springer, vol.140, N3 (2007) pp. 412-434.

[14] V. Ustimenko, *On the extremal regular directed graphs without commutative diagrams and their applications in Coding Theory and Cryptography*, Special Issue of Albanian J. Math. on Algebra and Computational Algebraic Geometry, Vol.1, No 4 (2007).

[15] V. A. Ustimenko, *On the extremal graph theory for directed graphs and its cryptographical applications*, in T. Shaska, W. C. Huffman, D. Joener and V. Ustimenko (editors), Advances in Coding Theory and Cryptography. Series on Coding Theory and Cryptology, World Scientific, vol. 3, (2007).

[16] V. Ustimenko and J. Kotorowicz, *On the Properties of Stream Ciphers Based on Extremal Directed Graphs*, in Roland E. Chen (editor), Cryptography Research Perspective, Nova Science Publishers, April 2009.

[17] V. Ustimenko, *Algebraic groups and small world graphs of high girth*, Albanian J. Math., Vol. 3, No 1 (2009), 25-33.

[18] V. A. Ustimenko, *On the cryptographical properties of extremal algebraic graphs*, in Algebraic Aspects of Digital Communications, IOS Press (Lectures of Advanced NATO Institute, NATO Science for Peace and Security Series - D: Information and Communication Security, Volume 24, July 2009, 296 pp.

[19] V. Ustimenko, *Maximality of affine group and hidden graph cryptsystems*, Journal of Algebra and Discrete Mathematics, October, 2004, v.10, pp. 51-65.

[20] V. Ustimenko,*On the extremal regular directed graphs without commutative diagrams and their applications in Coding Theory and Cryptography*, 2007, Albanian Journal of Mathematics, 2007 (Special Issue on Algebra and Computational Algebraic Geometry), Vol 1, N4 (2007).

[21] A. Wróblewska, *On some properties of graph based public key*, Albanian J. Math., vol. 2, No 3 (2008), Special Issue "New Challenges of Digital Communications", Proc. of NATO Advanced Studies Institute, Vlora, 2008, pp. 229-234.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 309–315

ISBN 978-83-60810-27-9
ISSN 1896-7094

Analysis of Pseudo-Random Properties of Nonlinear Congruential Generators with Power of Two Modulus by Numerical Computing of the b-adic Diaphony

Ivan Lirkov
Institute of Information and Communication Technologies
Bulgarian Academy of Sciences
Acad G. Bonchev, bl. 25A, 1113 Sofia, Bulgaria
E-mail: ivan@parallel.bas.bg

Stanislava Stoilova
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev, bl. 8, 1113 Sofia, Bulgaria
E-mail: stoilova@math.bas.bg

***Abstract*—We consider two nonlinear methods for generating uniform pseudo-random numbers in [0, 1), namely quadratic congruential generator and inversive congruential generator. The combinations of the Van der Corput sequence with the considered nonlinear generators are proposed. We simplify the mixed sequences by a restriction of the b-adic representation of the points.**

We study numerically the b-adic diaphony of the nets obtained through quadratic congruential generator, inversive congruential generator, their combinations with the Van der Corput sequence, and the simplification of the mixed sequences. The value of the b-adic diaphony decreases with the increase of the number of the points of the simplified sequences which proves that the points of the simplified sequences are pseudo-random numbers. The analysis of the results shows that the combinations of the Van der Corput sequence with these nonlinear generators have good pseudo-random properties as well as the generators.

I. Introduction

Many branches of the contemporary science research as stochastic simulation, stochastic optimization techniques, computational statistics, Monte Carlo simulation, molecular dynamics, cryptography, computer graphics, etc., depend on the random numbers [3], [4], [16], [19]–[21], [29]. In practice, random numbers are generated by deterministic recursive rules, formulated in terms of simple arithmetic operations. Obviously the emerging numbers can at best be pseudo-random. To design random number generators that approximate "true randomness" as closely as possible, is a great challenge. Except this obvious requirement, pseudo-random number generators should possess reproducible results, they should be portable between different computer architectures, and since in most applications millions of random numbers are needed, generators should be as efficient as possible.

The fields of probability and statistics are built over the abstract concepts of probability space and random variable. This has given rise to elegant and powerful mathematical theory, but exact implementation of these concepts on conventional computers seems impossible. Random variables and other random objects are simulated by deterministic algorithms. The purpose of these algorithms is to produce sequences of numbers or objects whose behavior is almost undistinguishable from that of their "truly random" counterparts, at least for the application of interest. Depending on the context, pseudo-random number generators must satisfy different requirements.

For Monte Carlo methods, the main aim is to reproduce the statistical properties on which these methods are based, so that the Monte Carlo estimators have a behavior as expected. On the other hand, for gambling machines and cryptology, observing the sequence of output values for some time should provide no practical advantage for predicting the forthcoming numbers better than by just guessing at random. In computational statistics, random variate generation is usually made in two steps. The first step is generating imitate ions of independent and identically distributed (i.i.d.) random variables having the uniform distribution over the interval (0, 1). And the second step is applying transformations to these i.i.d. $U(0,1)$ random variates in order to generate (or imitate) random variates and random vectors from arbitrary distributions. The expression (pseudo-)random number generator (RNG) usually refers to an algorithm used for first step. In principle, the simplest way of generating a random variate X with distribution function F from a $U(0,1)$ random variate U is to apply the inverse of F to $U : X = F^{-1}(U) \overset{\mathrm{def}}{=} \min\{x|F(x) \geq U\}$. This is the inversion method. It is easily seen that X has the desired distribution: $P[X \leq x] = P[F^{-1}(U) \leq x] = P[U \leq F(x)] = F(x)$. Other methods are sometimes preferable when F^{-1} is too difficult or expensive to compute.

The basic ingredients of any stochastic simulation are uniform pseudo-random numbers in the interval $[0,1)$. The outcome of a typical stochastic simulation strongly depends on the structural and statistical properties of the underlying pseudo-random number generators. That is why their quality is of fundamental importance for the success of the simulation. The classical and most frequently used method for the generation of pseudo-random numbers is still the linear congruential

978-83-60810-27-9/09/$25.00 © 2010 IEEE

method. However, its simple linear nature implies several undesirable regularities. Mainly for this reason, a variety of nonlinear methods for the generation of pseudo-random numbers has been introduced and studied as alternatives to the linear congruential method. These nonlinear congruential generators provide a very attractive alternative to linear congruential generators and have been extensively studied in the literature [1], [5], [7]–[10], [15], [16], [23], [25]–[28]. Two special cases: the quadratic congruential generator and the inversive congruential generator, are of special interest. A good survey of this important research area is given in [4], [17], [19]–[21], [29]. Many authors study pseudo-random properties of the sequences, generated by quadratic and inversive congruential generators by using the bounds of the discrepancy [2], [6], [11]–[13], [22], [24].

We use the $b-$adic diaphony to study pseudo-random numbers, generated by quadratic and inversive congruential generators. We will recall some known notions and definitions.

Let $\xi = (\mathbf{x}_j)_{j\geq 0}$ be a sequence in $[0,1)^s$ and $J = [\boldsymbol{\alpha}, \boldsymbol{\beta}) \subseteq [0,1)^s$, where $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_s)$ and $\boldsymbol{\beta} = (\beta_1, \dots, \beta_s)$. For arbitrary integer M $A_M(\xi; J)$ is the number of belonging to J points of ξ. The sequence $\xi = (\mathbf{x}_j)_{j\geq 0}$ is uniformly distributed mod 1 in $[0,1)^s$ if for every $J = [\boldsymbol{\alpha}, \boldsymbol{\beta}) \subseteq [0,1)^s$ the equality

$$\lim_{M\to\infty} \frac{A_M(\xi; J)}{M} = \prod_{i=1}^{s} (\beta_i - \alpha_i)$$

is hold, see [30].

The above equality shows when a sequence of points in $[0,1)^s$ is uniformly distributed but it does not allow to compare the distributions of two sequences. For that purpose various measures of the distribution of the sequences are used. Such measure is the $b-$adic diaphony [14].

Let $b \geq 2$ be a fixed integer and b denotes the base of the number system everywhere in this paper. Also, let an arbitrary $x \in [0,1)$ have a $b-$adic representation $x = \sum_{l=0}^{\infty} x_l b^{-l-1}$, where for $l \geq 0$, $x_l \in \{0, 1. \dots, b-1\}$ and for infinitely many values of l, $x_l \neq b-1$. The integer part of $b-$adic logarithm of x is $\lfloor \log_b x \rfloor = -g-1$, if $x_l = 0$ for $l < g$ and $x_g \neq 0$. We denote the operation $x \dot{-} y = \sum_{l=0}^{\infty} [(x_l - y_l) (\mathrm{mod}\ b)] b^{-1-l}$ and for vectors $\mathbf{x}, \mathbf{y} \in [0,1)^s$ we note $\mathbf{x} \dot{-} \mathbf{y} = (x_1 \dot{-} y_1, x_2 \dot{-} y_2, \dots, x_s \dot{-} y_s)$, where $\mathbf{x} = (x_1, x_2, \dots, x_s), \mathbf{y} = (y_1, y_2, \dots, y_s)$.

We define the functions $\gamma : [0,1) \to \mathbb{R}$ and $\Gamma : [0,1)^s \to \mathbb{R}$ as

$$\gamma(x) = \begin{cases} b+1-(b+1)b^{1+\lfloor \log_b x \rfloor}, & \text{if } x \in (0,1) \\ b+1, & \text{if } x = 0 \end{cases}$$

and

$$\Gamma(\mathbf{x}) = -1 + \prod_{d=1}^{s} \gamma(x_d), \mathbf{x} = (x_1, x_2, \dots, x_s).$$

The $b-$adic diaphony $F_N(\xi)$ of the first N elements of the sequence $\xi = (\mathbf{x}_i)_{i\geq 0}$ in $[0,1)^s$ is defined as

$$F_N(\xi) = \left(\frac{1}{(b+1)^s - 1} \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \Gamma(\mathbf{x}_i \dot{-} \mathbf{x}_j) \right)^{\frac{1}{2}},$$

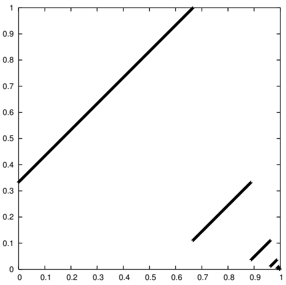


Fig. 1. The distribution of points of the Van der Corput sequence with $N = 1024$, $b = 3$.

where the coordinates of all points of the sequence ξ are $b-$adic rational.

Let $i = \sum_{l=0}^{\infty} i_l b^l$ be the $b-$adic representation of the non-negative integer i. Then the i-th element of the Van der Corput sequence is defined as

$$\zeta_b(i) = \sum_{l=0}^{\infty} i_l b^{-l-1}.$$

The Van der Corput sequence is deterministic sequence and does not have pseudo-random properties. The distribution of 1024 points of the net

$$(\zeta_b(i), \zeta_b(i+1))$$

is seen on Fig. 1.

II. Nonlinear Congruential Generators

Let $M \geq 2$ be modulus of a nonlinear congruential generator. The sequence of pseudo-random numbers $\left\{ x_i = \frac{y_i}{M} \right\}$ is produced by nonlinear congruential generator $y_{i+1} = f_M(y_i), \quad y_i \in [0, M), \quad \forall i = 0, 1, \dots$, where $y_0 \in [0, M)$ is the initial starting point. The study of the pseudo-randomness of the sequence $x_i, i = 0, 1, \dots$ is connected with the estimation of the distribution of the two-dimensional net

$$(x_i, x_{i+1}) = \left(\frac{y_i}{M}, \frac{y_{i+1}}{M} \right). \qquad (1)$$

Our aim is a numerical computing of the $b-$adic diaphony F_N of these two-dimensional nets. For the function f_M we use two nonlinear congruential generators: quadratic generator and inversive generator.

The quadratic congruential generator (QCG) is introduced by Knuth [17] and studied by [2], [11], [12]. This generator is defined as

$$f_M^{QCG}(y_i) \equiv q_2 y_i^2 + q_1 y_i + q_0 (\mathrm{mod}\ M),$$
$$y_{i+1} = f_M^{QCG}(y_i), \quad y_i \in [0, M), \qquad (2)$$

where q_2, q_1, q_0 are three integer parameters.

The inversive congruential generator (ICG) is defined as

$$f_M^{ICG}(y_i) \equiv \begin{cases} r_{-1} y_i^{-1} + r_0 (\mathrm{mod}\ M), & \text{if } y_i \geq 1, \\ r_0, & \text{if } y_i = 0, \end{cases}$$

TABLE I
THE DIAPHONY F_N OF QUADRATIC AND INVERSIVE GENERATORS, $b = 3$.

M	QCG	ICG
16	0.175682	0.310316
32	0.122138	0.175682
64	0.074016	0.198416
128	0.057257	0.159625
256	0.039630	0.123801
512	0.028726	0.053107
1024	0.020017	0.046659
2048	0.014486	0.029463
4096	0.011113	0.023956
8192	0.007010	0.015353
16384	0.005087	0.011706
32768	0.003595	0.007119
65536	0.002502	0.006440

where r_{-1}, r_0 are integer parameters and y_i^{-1} denotes the inversive element of y_i, i.e. $y_i^{-1} y_i \equiv 1 (\bmod\ M)$,

$$y_{i+1} = f_M^{ICG}(y_i), \quad y_i \in [0, M). \tag{3}$$

In this paper we consider generators with modulus $M = 2^\mu$, $\mu \geq 4$. In this case the quadratic congruential generator (2) is purely periodic with maximum possible period length $M = 2^\mu$ if and only if

$$q_0 \equiv 1 (\bmod\ 2), \quad q_2 \equiv 0 (\bmod\ 2), \quad q_2 \equiv q_1 - 1 (\bmod\ 4).$$

In [10] it is proved that for $M = 2^\mu$ the inversive congruential generator (3) has maximal period length $2^{\mu-1}$ if and only if

$$r_{-1} \equiv 1 (\bmod\ 4) \quad \text{and} \quad r_0 \equiv 2 (\bmod\ 4).$$

Further in this paper we will denote the period of the generators by N.

III. PSEUDO-RANDOMNESS OF THE SEQUENCES PRODUCED BY QUADRATIC AND INVERSIVE CONGRUENTIAL GENERATORS

In this section the $b-$adic diaphony of two-dimensional nets (1), obtained by quadratic and inversive congruential generators, is numerically computed. We used PRNG library [31] created by Otmar Lendl to obtain the sequences. The parameters used in the numerical experiments in this work are: $q_2 = 8, q_1 = 5, q_0 = 3, r_{-1} = 9, r_0 = 6, y_0 = 1$, i. e.

$$y_{i+1} = f_M^{QCG}(y_i) \equiv 8y_i^2 + 5y_i + 3 (\bmod\ M),$$

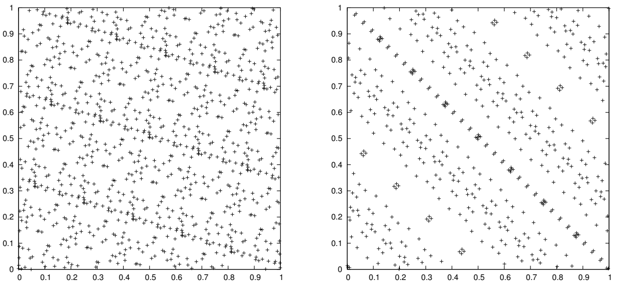
$$y_{i+1} = f_M^{ICG}(y_i) \equiv 9y_i^{-1} + 6 (\bmod\ M).$$

The distribution of the points of two-dimensional nets (1) with $M = 1024$ can be seen at Fig. 2. Table I contains the results for the $b-$adic diaphony of such two-dimensional nets.

IV. PSEUDO-RANDOMNESS OF THE COMBINATION OF THE VAN DER CORPUT SEQUENCE WITH QUADRATIC AND INVERSIVE CONGRUENTIAL GENERATORS

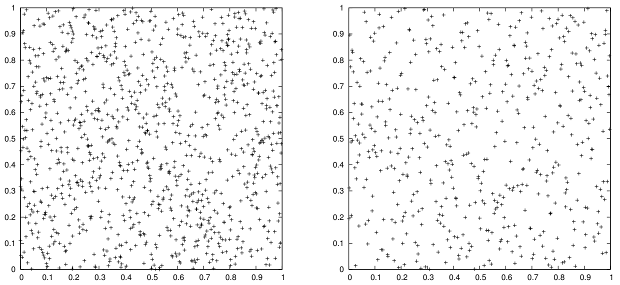
We consider the net

$$(\zeta_b(y_i), \zeta_b(y_{i+1})), i = 0, 1, \ldots, N-1. \tag{4}$$



QCG, $N = 1024$ ICG, $N = 512$

Fig. 2. The distribution of points of quadratic and inversive generators, $M = 1024$.



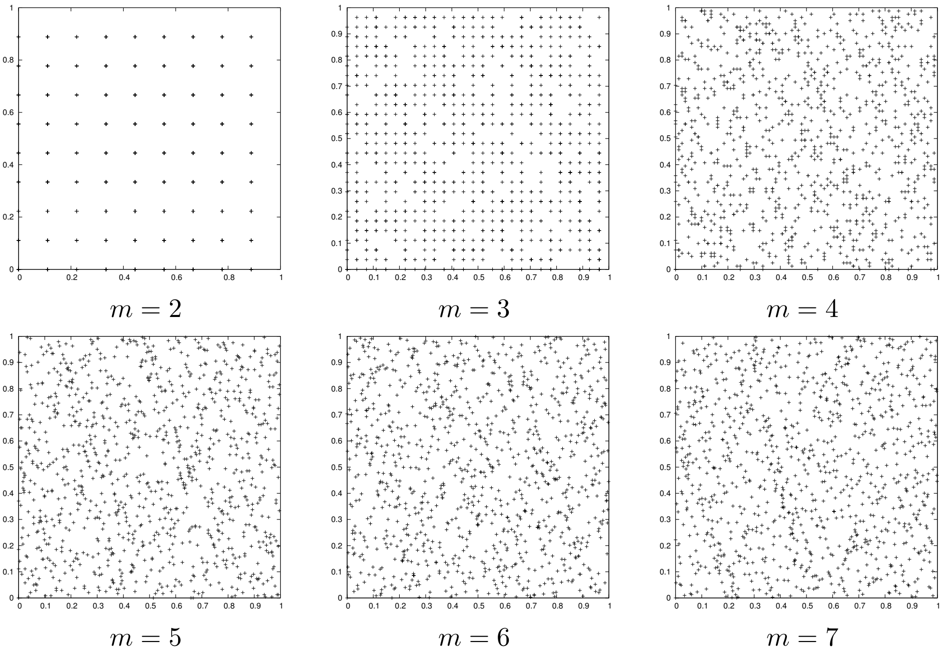
QCG, $N = 1024$ ICG, $N = 512$

Fig. 3. The distribution of points of the combination $\zeta_b(y_i)$ of quadratic and inversive generators with Van der Corput sequence with $M = 1024$, $b = 3$.

We compute the $b-$adic diaphony of the combination of the Van der Corput sequence ζ_b with quadratic and inversive generators, i.e. $y_{i+1} = f_M^{QCG}(y_i)$ and $y_{i+1} = f_M^{ICG}(y_i)$. The combination of the Van der Corput sequence with quadratic generator is proposed by Oto Strauch. We consider the combination of the Van der Corput sequence with inversive generator, too. In such way, the obtained nets have a better pseudo-random property than original sequences. The distribution of such nets for $M = 1024$ can be seen at Fig. 3. The obtained results for the $b-$adic diaphony of these combinations are presented in Table II. Fig. 4 shows a comparison between the computed $b-$adic diaphony of the nets (1) and (4) using two nonlinear generators.

TABLE II
THE DIAPHONY F_N OF THE COMBINATION OF THE VAN DER CORPUT SEQUENCE WITH QUADRATIC AND INVERSIVE GENERATORS, $b = 3$.

M	QCG	ICG
16	0.189215	0.310316
32	0.118721	0.161015
64	0.084152	0.155085
128	0.059790	0.084913
256	0.045328	0.075950
512	0.029762	0.063425
1024	0.022144	0.054596
2048	0.015634	0.036257
4096	0.010938	0.033067
8192	0.008214	0.019782
16384	0.005680	0.016421
32768	0.003888	0.010836
65536	0.002777	0.007971



$m = 2$ $m = 3$ $m = 4$

$m = 5$ $m = 6$ $m = 7$

Fig. 5. The distribution of the simplification of the quadratic generator with $M = 1024$, $b = 3$.

TABLE III
THE DIAPHONY F_N OF THE NETS (6) WITH A QUADRATIC GENERATOR, $b = 3, M = 2^{\mu}, \mu = 4, \dots, 16$.

M	F_N									
	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11
16	0.20184	0.19245								
32	0.14672	0.12360	0.13709							
64	0.11102	0.09234	0.08944							
128	0.09456	0.06650	0.06661	0.06359						
256	0.08285	0.04829	0.04651	0.04376	0.04438					
512	0.07757	0.03722	0.03151	0.03000	0.03060					
1024	0.07392	0.03195	0.02310	0.02256	0.02162	0.02022				
2048	0.07248	0.02785	0.01767	0.01518	0.01543	0.01550				
4096	0.07163	0.02573	0.01368	0.01117	0.01094	0.01079	0.01157			
8192	0.07131	0.02478	0.01087	0.00824	0.00734	0.00836	0.00743	0.00728		
16384	0.07112	0.02408	0.00940	0.00620	0.00530	0.00559	0.00534	0.00560		
32768	0.07102	0.02380	0.00878	0.00479	0.00411	0.00387	0.00412	0.00407	0.00388	
65536	0.07097	0.02359	0.00823	0.00383	0.00293	0.00281	0.00275	0.00273	0.00287	0.00274

More detailed analysis of the results is presented in section VI.

V. SIMPLIFICATION

Let $m \geq 1$ be an arbitrary integer and $x \in [0,1)$ then for the $b-$adic expression

$$x = 0.x_1 x_2 \dots x_{m-1} x_m x_{m+1} \dots$$

we define

$$\zeta^*_{b^m}(x) = 0.x_m x_{m-1} \dots x_2 x_1.$$

In fact we truncate the $b-$adic expression of the number x to the m digits and we reverse the digits. O. Strauch proposed the net

$$\zeta^*_{b^m}\left(\frac{y_i}{M}\right), i = 0, 1, \dots, N-1. \tag{5}$$

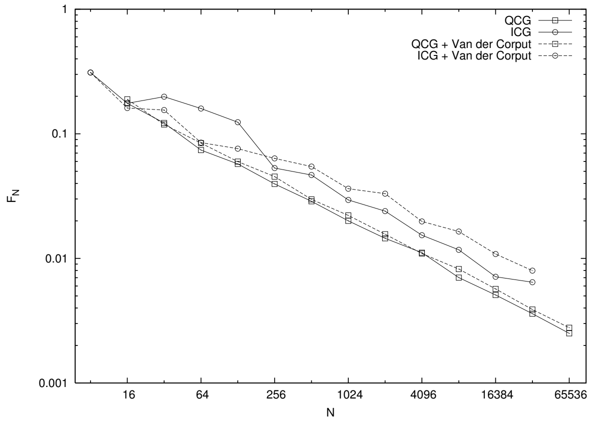
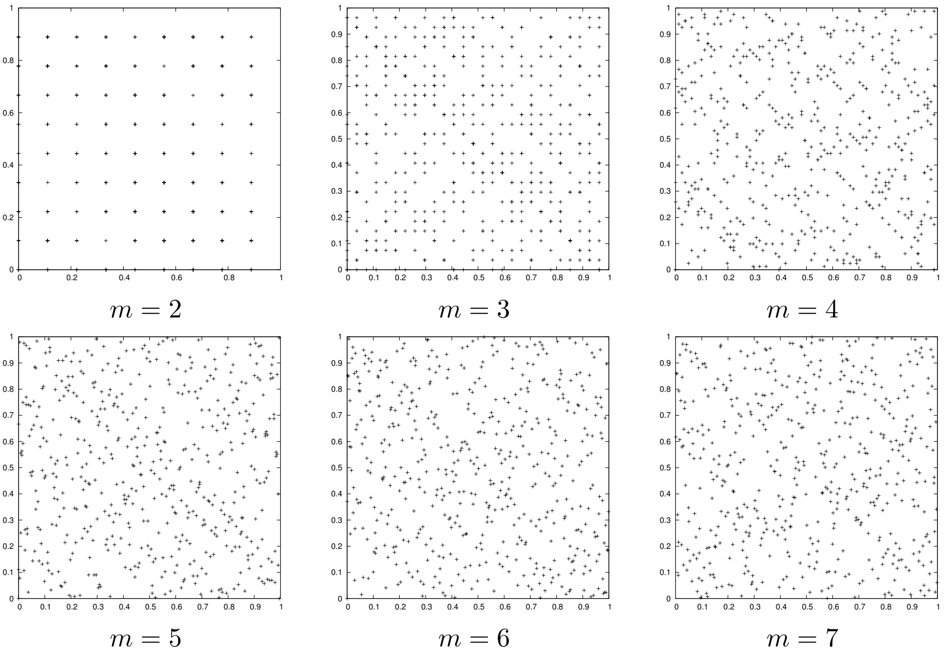


Fig. 4. The Diaphony F_N of the nets (1) and (4) with quadratic and inversive generators.



$m = 2$ $m = 3$ $m = 4$

$m = 5$ $m = 6$ $m = 7$

Fig. 6. The distribution of the simplification of the inversive generator with $M = 1024$, $b = 3$.

TABLE IV
THE DIAPHONY F_N OF THE NETS (6) WITH AN INVERSIVE GENERATOR, $b = 3, M = 2^{\mu}, \mu = 4, \ldots, 16$.

M	F_N									
	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11
16	0.31032	0.31032								
32	0.16102	0.20387	0.16480							
64	0.19425	0.11520	0.14865							
128	0.11662	0.07514	0.11154	0.07343						
256	0.09570	0.10746	0.05317	0.05888	0.08200					
512	0.09331	0.07922	0.05841	0.04082	0.05530					
1024	0.08536	0.05432	0.02671	0.03312	0.05335	0.03935				
2048	0.07907	0.03706	0.02297	0.02626	0.02243	0.04587				
4096	0.07568	0.03023	0.01977	0.02686	0.02510	0.02545	0.01777			
8192	0.07359	0.02582	0.01783	0.01611	0.01029	0.01503	0.01139	0.01639		
16384	0.07252	0.02576	0.01542	0.01038	0.00661	0.01147	0.01323	0.00816		
32768	0.07146	0.02398	0.01180	0.00633	0.00556	0.01108	0.00838	0.00887	0.00451	
65536	0.07133	0.02411	0.01127	0.00576	0.00522	0.00577	0.00410	0.00723	0.00635	0.00369

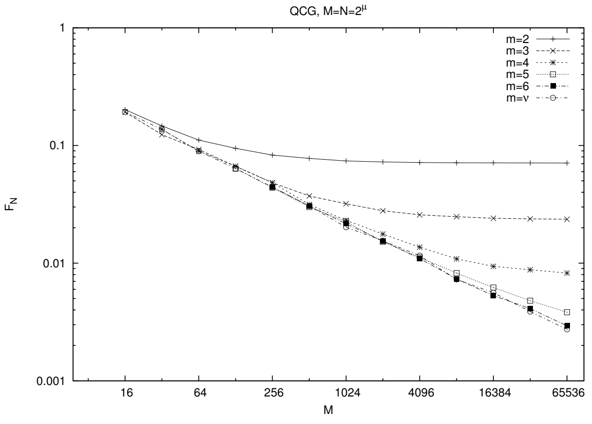


Fig. 7. The Diaphony F_N of the nets (6) with a quadratic generator.

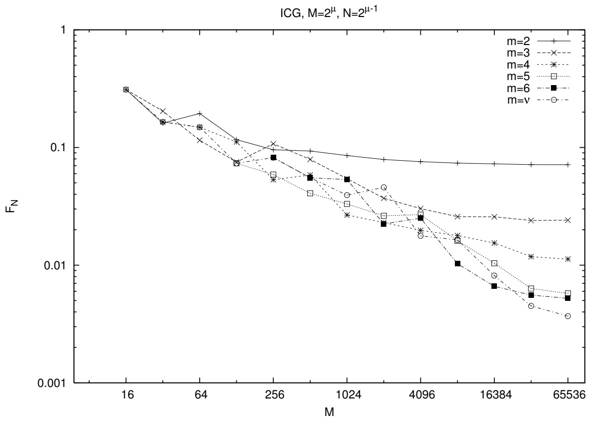


Fig. 8. The Diaphony F_N of the nets (6) with an inversive generator.

For pseudo-randomness of (5) we study the $b-$adic diaphony F_N of the two-dimensional net

$$\left(\zeta^*_{b^m}\left(\frac{y_i}{M}\right), \zeta^*_{b^m}\left(\frac{y_{i+1}}{M}\right)\right), i = 0, 1, \ldots, N-1. \qquad (6)$$

Let ν be an integer such that $b^{\nu-1} < N \leq b^{\nu}$. For two-dimensional net with N points we choose $m = 2, \ldots, \nu$. The distribution of the points of the nets (6) for $M = 1024$ for six values of the number m is shown in Figs. 5 and 6. Tables III and IV as well as Figs. 7 and 8 show the computed $b-$adic diaphony of such nets using two nonlinear generators.

VI. Analysis of the Results

The results in Table I show that the $b-$adic diaphony of the two-dimensional net (1) tends to zero with the increase of the number of the points for both considered nonlinear generators. The $b-$adic diaphony has the same order for both generators. We will note a fact that for one and the same modulus the number of the points produced by an inversive generator is twice less than the number of the points from a quadratic generator. The fast convergence of the $b-$adic diaphony of the quadratic generator shows that it has better pseudo-random properties. In other words, this deterministic algorithm simulates better random number. The same conclusion can be made for the $b-$adic diaphony of the combination of the Van der Corput sequence with considered nonlinear generators. The comparison between Fig. 1 and Fig. 3 shows that the combination of the deterministic Van der Corput sequence with the nonlinear generators improves the distribution of two-dimensional net. Hence, the pseudo-random properties of the sequence $\zeta_b(y_i)$ are similar to the properties of the generators. Although the values of the $b-$adic diaphony of the combination are greater than the values of the $b-$adic diaphony of the generators, from the comparison between Tables I and II it is seen that the $b-$adic diaphony of the combination has a faster convergence to zero. The two-dimensional net (4) with the used nonlinear generators have better uniform distribution than (1), therefore, the combination $\zeta_b(y_i)$ has better pseudo-random properties than original sequences. Fig. 4 illustrates this fact.

The usage of the proposed simplification leads to a restriction of the number of the sequence points. This simplification maps the points produced by the generators to uniformly distributed nets with b^{2m} points in $[0, 1)^2$. Under too strong restriction, i.e. $m = 2$ or $m = 3$, the value of the $b-$adic diaphony of the net (6) for quadratic generator stays big independently of the increase of points number. It is true for inversive generator, too. However, for big m the behavior of the $b-$adic diaphony using both generators is different. This behavior for different restrictions $m = 2, 3, \ldots, 11$ can be seen in the last two tables and the last two figures. Comparing Fig. 7 and 8 we conclude that F_N of the simplification of the quadratic generator has faster convergence to zero. The diaphony F_N of the simplification of the inversive generator also tends to zero but for $N < b^{2m}$ there are some intervals where the distribution worsen and F_N increases.

VII. Conclusions and Future Work

In practice, different pseudo-random number generators are used depending on the application. A generator is good, if it has some properties as large period length, good uniform distribution qualities, lattice structure, efficiency, fast generation algorithm, repeatability, portability, unpredictability.

The combination of the Van der Corput sequence with nonlinear generators, in fact, is a new generator. This combination saves the properties of the generators. The obtained generators have the same period length as the original generators. The good uniform distribution qualities are shown on Fig. 3. Obviously, the new generators have lattice structure, efficiency, fast generation algorithm, reproduce exactly the same sequence on different computers and we can not predict the next generated value by the algorithm from the previous ones. In this way the combination of the Van der Corput sequence with quadratic and inversive generators is a good pseudo-random number generator.

The comparison between the combinations of the Van der Corput sequences with quadratic generator and inversive generator shows that the $b-$adic diaphony of the first combination tends to zero faster than $b-$adic diaphony of the second combination. It means that the first combination has better simulation of the pseudo-random numbers. On the other hand, the second combination has less points than the first combination, but the behavior of the $b-$adic diaphony for this combination is similar. The proposed simplification of quadratic and inversive generators can also be considered as a generator. Depending on the purpose of the application, the combination of the Van der Corput sequence with quadratic and inversive generators or proposed simplification can be used. The results show that the $b-$adic diaphony is a good tool to study pseudo-randomness of the sequences.

These research can be continued in several directions: from theoretical point of view — to find estimations of the $b-$adic diaphony of all considered generators; from the viewpoint of Monte Carlo and quasi-Monte Carlo applications — to find connection between the error of the numerical integration and the $b-$adic diaphony and to study applications of the considered generators in the area of the computer graphics for uniform sampling.

Acknowledgments

We would like to thank Prof. Oto Strauch for the wonderful ideas about the combination of the Van der Corput sequence with quadratic generator and the simplification of this combination. The study of pseudo-randomness of the proposed by Prof. Oto Strauch sequences is very interesting and useful for us. This work is supported by the project Bg-Sk-207, Bulgarian NSF.

References

[1] Blackburn, S. R., Gomes-Perez, D., Gutierrez, J., Shparlinski, I. E., Predicting Nonlinear Pseudorandom Number Generators, *Mathematics of Computation*, **74**, No 251, pp. 1471–1494, (2004).

[2] Blažeková, O., Strauch, O., Pseudo-randomness of quadratic generators, *Uniform distribution theory*, **2**, No 2, pp. 105–120, (2007).

[3] Dimov, I. T., Penzov, A. A., Stoilova, S. S., Parallel Monte Carlo, Sampling Scheme for Sphere and Hemisphere, *Lecture Notes in Computer Science*, **4310**, Springer, Berlin, Heidelberg, pp. 148–155, (2007).

[4] Drmota, M., Tichy, R. F., Sequences, Discrepancies and Applications, *Lecture Notes in Mathematics*, **1651**, Springer, Berlin, Heidelberg, (1997).

[5] Eichenauer-Herrmann, J., Inversive congruential pseudorandom numbers avoid the planes, *Mathematics of Computation*, **56**, No 193, pp. 297–301, (1991).

[6] Eichenauer-Herrmann, J., On the discrepancy of inversive congruential pseudorandom numbers with prime power modulus, *Manuscripta Mathematica*, **71**, pp. 153–161, (1991).

[7] Eichenauer-Herrmann, J., Emmerich, F., Compound Inversive Congruential Pseudorandom Numbers: An Average-Case Analysis, *Mathematics of Computation*, **65**, No 213, pp. 215–225, (1996).

[8] Eichenauer, J., Lehn, J., A non-linear congruential pseudorandom number generator, *Statist. Helfe*, **27**, pp. 315–326, (1986).

[9] Eichenauer, J., Grothe, H., Lehn, J., On the Period Length of Pseudorandom Vector Sequences Generated by Matrix Generators, *Mathematics of Computation*, **52**, No 185, pp. 145–148, (1989).

[10] Eichenauer, J., Lehn, J., Topuzoğlu, A., A Nonlinear Congruential Pseudorandom Number Generator with Power of Two Modulus, *Mathematics of Computation*, **51**, No 184, pp. 757–759, (1988).

[11] Eichenauer-Herrmann, J., Niederreiter, H., On the discrepancy of quadratic congruential pseudorandom numbers, *J. Comput. Appl. Math.*, **34**, No 2, pp. 243–249, (1991).

[12] Eichenauer-Herrmann, J., Niederreiter, H., An improved upper bound for the discrepancy of quadratic congruential pseudorandom numbers, *Acta Arithmetica*, **69**, No 2, pp. 193–198, (1995).

[13] Eichenauer-Herrmann, J., Niederreiter, H., Lower Bounds for the Discrepancy of Triples of Inversive Congruential Pseudorandom Numbers with Power of Two Modulus, *Monatshefte für Mathematik*, **125**, pp. 211–217, (1998).

[14] Grozdanov, V., Stoilova, S. The $b-$adic diaphony, *Rendiconti di Matematica*, **22**, pp. 203–221, (2002).

[15] Gutierrez, J., Niederreiter, H., Shparlinski, I. E., On the Multidimensional Distribution of Inversive Congruential Pseudorandom Numbers in Parts of the Period, *Monatshefte für Mathematik*, **129**, pp. 31–36, (2000).

[16] Hellekalek, P., Inversive Pseudorandom Generators: Concepts, Results and Links, *Proceedings of the* 27^{th} *conference on Winter simulation*, Arlington, Virginia, US, IEEE CS, Washington, DC, USA, pp. 255–262, (1995).

[17] Knuth, D. E., Seminumerical algorithms. The art of computer programming, **2**, 2nd edition, Addison Wesley, Reading, MA, (1981).

[18] Kuipers, L., Niederreiter, H., Uniform distribution of sequences, John Wiley, New York, (1974).

[19] L'Ecuyer, P., Uniform Random Number Generation, *Annals of Operations Research*, **53**, No 1, Springer-Verlag, pp. 77–120, (1994).

[20] Niederreiter, H., Quasi-Monte Carlo Methods and Pseudo-Random Numbers, *Buletin of the American Mathematical Society*, **84**, No 6, pp. 957–1041, (1978).

[21] Niederreiter, H., Random number generation and quasi-Monte Carlo methods, *CBMS-NSF Regional Conference Series in Applied Mathematics*, **63**, SIAM, Philadelphia, PA, (1992).

[22] Niederreiter, H., A Discrepancy Bound for the Hybrid Sequences Involving Digital Explicit Inversive Pseudorandom Numbers, *Uniform Distribution Theory*, **5**, No 1, pp. 53–63, (2010).

[23] Niederreiter, H., The Serial Test for Congruential Pseudorandom Numbers Generated by Inversions, *Mathematics of Computation*, **52**, No 185, pp. 135–144, (1989).

[24] Niederreiter, H., Lower Bounds for the Discrepancy of Inversive Congruential Pseudorandom Numbers, *Mathematics of Computation*, **55**, No 191, pp. 277–287, (1990).

[25] Niederreiter, H., Shparlinski, I. E., On the distribution of inversive congruential pseudorandom numbers in parts of the period, *Mathematics of Computation*, **70**, No 236, pp. 1569–1574, (2000).

[26] Niederreiter, H., Shparlinski, I. E., Exponential sums and the distribution of inversive congruential pseudorandom numbers with prime-power modulus, *Acta Arithmetica*, **XCII**, No 1, pp. 89–98, (2000).

[27] Niederreiter, H., Shparlinski, I. E., On the Distribution and Lattice Structure of Nonlinear Congruential Pseudorandom Numbers, *Finite Fields and Their Applications*, **5**, pp. 246–253, (1999).

[28] Niederreiter, H., Shparlinski, I. E., On the Distribution of Pseudorandom Numbers and Vectors Generated by Inversive Methods, *Applicable Algebra in Engineering, Communication and Computing*, **10**, Springer-Verlag, pp. 189–202, (2000).

[29] Strauch, O., Porubský, Š., Distribution of Sequences: A Sampler, Peter Lang, Frankfurt am Main, (2005).

[30] Weil, H., Über die Gleichverteilung von Zahlen mod. Eins., *Mathematische Annalen*, **77**, No 3, Springer, pp. 313–352, (1916).

[31] Pseudo-Random Number Generator, http://statmath.wu.ac.at/prng/

ISBN 978-83-60810-27-9
ISSN 1896-7094

Assembling Recursively Stored Sparse Matrices

Michele Martone, Salvatore Filippone, Salvatore Tucci
University of Rome "Tor Vergata", Via del Politecnico 1, 00133 Rome, Italy
Email: {michele.martone,salvatore.filippone,tucci}@uniroma2.it
Marcin Paprzycki
Warsaw Management Academy, Poland Email: marcin.paprzycki@ibspan.waw.pl

***Abstract*—Recently, we have introduced an approach to multi-core computations on sparse matrices using recursive partitioning, called *Recursive Sparse Blocks* (*RSB*). In this document, we discuss issues involved in assembling matrices in the RSB format. Since the main expected application area is iterative methods, we compare the performance of matrix assembly to that of matrix-vector multiply (*SpMV*), outlining both scalability of the method and execution times ratio.**

I. Introduction

IN RECENT papers, we have introduced a recursive data structure for performing Sparse BLAS ([1]) operations on cache based multi-core architectures. Initial performance results for the proposed format and its modifications/improvements, for both matrix-vector multiplication and triangular solve have been very encouraging (see, [2], [3], [4], [5]). We call our hybrid format *Recursive Sparse Blocks* (*RSB*), as it features the recursive subdivision of a matrix, with sparse submatrices in the leaves of a quad-tree.

Our efforts are targeted primarily towards *iterative methods*, appearing in the solution of either linear or eigenvalue problems ([6]). Here, the usual case for the user, is to select a specific iterative method (and preconditioner) for a given computational problem. For instance, the choice of a method over another could be influenced by the available computer hardware. With different sparse matrix storage formats, different operations could have different performance patterns on variously configured core/processor hardware. An important motivation for our work (as well as that of others, e.g.: CSB [7]) is the need for a format capable of performing thread-parallel Sparse Matrix-Vector Multiply (*SpMV*) with the same ease and comparable performance to the *Transposed SpMV* (*SpMV_T*). Since many iterative methods require both *SpMV* and *SpMV_T* (e.g. CGS or BiCGSTAB; see [8, section 2.4]), the development of a unified parallel approach to both operations is needed. Here, the common matrix formats, as *Compressed Sparse Rows* (*CSR*), and the plain *COOrdinate* (*COO*) formats do not support efficiently parallel *SpMV_T*.

At the problem level, the use of iterative methods often occurs in the solution of partial differential equations (PDEs), e.g. arising from simulating some physical phenomenon. Here, different usage patterns could arise; for instance: i) a sparse matrix could be instantiated and solved once; ii) a sparse matrix could be instantiated once and solved against a number of (updated/not known *a priori*) right hand sides. In choosing the best solution method for a given problem, the developer should take into account the overall (wall-clock) computation time; that is, the time to assembly a sparse matrix (and for updating it, if needed), as well as solving the problem. Additionally, the time for computing a preconditioner (See [6, Ch.10]) should be taken into account, but this issue is outside the scope of this paper.

The assembly of a sparse matrix could proceed in a number of ways. The author of [9, Ch.4] presents some common cases for the assembly of symmetric and unsymmetric matrices—from *usual* Finite Element Methods (FEM) data arrays as input, into COO and CSR formats. Overall, the most general routine for assembling a sparse matrix should be able to process as input the full list of coordinates of the matrix elements (*structural nonzeroes*), defining the *nonzeroes pattern* of the matrix. The numerical values of the matrix elements could be specified or updated in a later moment. The RSB matrix "*constructor*" we discuss here, works by converting the original array specifying the matrix as row-major sorted COO (now on, we will refer to COO assuming it as row-major sorted) into the RSB layout. Note that, from the assembly routines we present, it would be easy to derive routines for the extraction/conversion of rows or block-rows of RSB matrices.

We start by reviewing related works, in Sec. II. Next, we outline selected properties of RSB's quad-trees (in Sec. III). We follow by a presentation of our algorithm for converting a row-major sorted COO matrix to RSB, in Sec. IV. Then, in Sec. VI we report, for a representative set of matrices, how many *SpMV*'s are time-equivalent to a single matrix assembly. While we do not claim optimality of either of our techniques (the assembly and the *SpMV*), presented results illustrate the relevant relation between performance of these two operations. We have selected this particular comparison since, as stated above, *SpMV* is the basic operation for most standard iterative methods. Experiments were performed on machines and matrices described in Sec. V.

II. Related Work

The use of recursion in numerical linear algebra is a *recurring* theme (e.g., see [10]). We found proposals for *hypermatrix* (multilevel indexed matrices) based approaches to the solution of linear systems dating back to 1972 [11] (usage of hypermatrices with dense submatrices), and 1969 [12] (out-of-core dense matrix computations). Further, with regards to sparse matrices, Herrero and Navarro [13] investigated a hypermatrix-based Cholesky solver, but without discussing

978-83-60810-27-9/09/$25.00 © 2010 IEEE

performance of hypermatrix assembly. In [14] authors used asymmetrical *recursive bipartitioning* of sparse matrices in a distributed computing context. Sparse hypermatrix techniques were reportedly used for distributed-memory operations in the PERMAS proprietary package for FEM analysis [15]. The techniques most similar to the ones investigated here, reported in [16], deal with the optimal balancing of sparse matrix computations across distributed processors. Interestingly enough, while hypermatrix-based approaches have been applied to sparse matrix computations, almost no research has been reported as to what concerns assembly of such matrices. However, we found research in a similar spirit to ours in [17]. It documents algorithms and discusses patterns of usage in the assembly of sparse matrices in the context of their *serial* "MTL" package. Our discussion is more limited but in-depth than that, as we are concerned with a single pattern of construction: the conversion of COO input arrays to RSB, to be used on multi-core computers.

III. Some Properties of the Quad Trees Used in RSB Matrices

We described our rules for the construction of a quad tree-based recursive matrix representation in [2]. Given an $m \times k$ matrix A, we build a graph structure (*quad-tree*) q with nodes corresponding to *quadrant submatrices*. The four quadrants are sized respectively (in clockwise order, from the upper left) $\lceil \frac{m}{2} \rceil \times \lceil \frac{k}{2} \rceil$, $\lceil \frac{m}{2} \rceil \times \lfloor \frac{k}{2} \rfloor$, $\lfloor \frac{m}{2} \rfloor \times \lceil \frac{k}{2} \rceil$, and $\lfloor \frac{m}{2} \rfloor \times \lfloor \frac{k}{2} \rfloor$. This subdivision (or *bipartition*) is applied recursively to the quadrants; quadrants with no nonzero are not represented. Only leaf nodes are associated with actual data arrays, while inner ones contain only pointers. A simple *cutoff* function is used to balance the tree in order to obtain *leaf submatrices* with neither too many, nor too few nonzeroes. Fig. 23 depicts a matrix subdivided into RSB.

Let us now review some properties of our quad-trees, which will be useful during the discussion of matrix assembly.

Let us call q_h the *complete* quad-tree of height h; that is, the quad-tree having $N_i(q_h) \stackrel{def}{=} \sum_{i=0}^{h-1} 4^i$ intermediate nodes and $N_l(q_h) \stackrel{def}{=} 4^h$ leaf nodes, We indicate with $H(q)$ the height of quad-tree q. We assume that any quad-tree could be constructed by adding nodes to the singleton quad-tree q_0 (the one which is associated to the entire matrix). Let Q be the set of quad-trees with height ≥ 1. We call q' a $k-$*derivation* (or *derivation*, for short, if we ignore k) of quad-tree q, if q' can be built from q, by making one leaf an intermediate node, and adding $1 \leq k \leq 4$ leaves. We call q' an *indirect derivation* of quad-tree q, if q' can be built from q after a sequence of derivations. Observe that if q' is a $k-$derivation of q, then $N_i(q') = N_i(q) + 1$, and $N_l(q') = N_l(q) + k - 1$.

Property 1. *For any q among the possible quad-trees with height 1, we have $\frac{N_i(q)}{N_l(q)} \geq \frac{1}{4}$, and $\frac{N_i(q_1)}{N_l(q_1)} = \frac{1}{4}$.*

Proof: By explicit enumeration of possible cases. ■

Property 2. *For any $q \in Q$ with $H(q) > 1$,we have $\frac{N_i(q)}{N_l(q)} \geq \frac{1}{4}$*

```
/*Matrix $A$ is expressed using arrays $I, J, V$*/
Instantiate the root matrix node $s_A$, marked RSB and "open"
$[P, s_A] \leftarrow COO\_to\_RSB\_s(s_A, I, J)$/*symbolic subdivision*/
/*Now $s_A$ is the root of a quad-tree for A, with empty leaves*/
/*P is a rows pointer array for $I, J, V$*/
$COO\_to\_RSB\_V(s_A, P, V)$   /*numerical arrays shuffling*/
$COO\_to\_RSB\_J(s_A, P, J)$ /*indices shuffling/displacement*/
/*P is no longer needed and $I, J, V$ are in RSB order*/
$RSB\_Leaf\_Switch(s_A)$    /*indices switch*/
/*A number of leaf matrices has halfword indices, now.*/
return $s_A$ /*return $s_A$, now quad-tree for $A$*/
```

Fig. 1. $COO\_to\_RSB(I, J, V)$

Proof: Let q be a quad-tree having $\frac{N_i(q)}{N_l(q)} < \frac{1}{4}$, necessarily a derivation of a quad-tree q' having $\frac{N_i(q')}{N_l(q')} \geq \frac{1}{4}$. In the case q is a $k-$derivation of q', indicating $i = N_i(q'), l = N_l(q')$, we have $\frac{i}{l} \geq \frac{1}{4}$ and $\frac{i+1}{l-1+k} < \frac{1}{4}$. But this implies $4i - l \geq 0$ and $4i - l < k - 5$, which is impossible, for $1 \leq k \leq 4$. In the case q is an indirect derivation of q', it must be a derivation of some quad-tree q" having $\frac{N_i(q")}{N_l(q")} < \frac{1}{4} \leq \frac{N_i(q')}{N_l(q')}$, but existence of such q" is impossible, as we have seen. ■

If some internal node of q has one child only, we call q *degenerate* (with some terminology abuse;see [18, Sec. 2.3.4.5]).

Property 3. *For any sparse matrix M with no empty rows, if its corresponding quad-tree q is not degenerate, we have $\frac{N_i(q)}{N_l(q)} \leq 1$.*

Proof: Since M has no missing rows, it has some leaf node of q covering each row interval. Since q is not degenerate, at each level > 1, there are at least two nodes, or no node at all. Therefore, quad-tree q can be built by inserting additional $k \geq 0$ leaves to some binary tree q'. A non degenerate binary tree q' has $N_l(q') = N_i(q') + 1$, So we have $\frac{N_i(q)}{N_l(q)} = \frac{N_i(q')}{N_i(q')+k+1}$, whose upper limit is 1, for $k = 0$, and $N_i(q) \to \infty$. ■

Property 3 guarantees that for *non degenerate* quad-trees, there will be no more internal nodes than leaves. Please note that, for the time being, for simplicity of implementation, degenerate quad-trees are allowed in RSB.

IV. Building RSB From COO

Let us now describe in detail our approach for the conversion of an $m \times k$ matrix A with nnz nonzeros, expressed in (row-major sorted) COO (I, J coordinate arrays and the V numerical values array) into RSB order. The proposed procedure builds a quad-tree structure for A, allocating a *small* number of auxiliary structures (see the previous section) for the submatrix nodes, and reusing I, J, V. Unless otherwise stated, in the following, by *matrix* we will refer to A only, and denote as a *submatrix* any of the *quadrant submatrices* obtained by recursive bipartitioning (defined in Sec. III). Here, we assume no duplicates in the input (which happen in publicly available matrices; e.g.: ones from [19]).

There are three stages of assembly: first the *subdivision* of A in *COO_to_RSB_s*, where the input is repeatedly scanned, and a quad-tree structure is built; then the *shuffling* of rows laid in COO order to the rows of RSB sub-

```
1  $N \leftarrow [0,0,0,0]$ /*nonzeroes count for quadrants */
2  Allocate four $(s.m+1)$-sized arrays $L, M, R, P$
3  /*CS: Cache(s) Size, ES(= 8 for double): Element Size*/
4  $s_A.N_S \leftarrow 0$; $s_A.MAX_S \leftarrow (s_A.nnz \cdot ES)/(CS/N_{threads})$
5  $COO\_RowP(I, J, P, s_A.nnz, s_A.m)$ /*fill row pointers in P*/
6  $s_A.L \overset{p}{\leftarrow} P$; $s_A.R \overset{p}{\leftarrow} P+1$
7  /*$s_A.L$ points to row beginnings, $s_A.R$ points to row endings
   (aliasing the second element of P)*/
8  while Some leaf submatrix is still "open" do
9      $s_A.N_S \leftarrow s_A.N_s + 1$;
10     $s \leftarrow$ "largest" open submatrix
11     if $\delta_r(s.m, s.k, s.nnz, CS, ES, WS)$ then
12         /*copy subrow pointers stored in $s.I, s.J$ */
13         $L \leftarrow s.I$; $R \leftarrow s.J$;
14         /*get quadrants info $N$, fill middle pointers array $M$*/
15         $N \leftarrow Subrow\_Split(s, L, R, M, J)$
16         /*split s, appending up to four quadrant submatrices*/
17         $RSB\_Split\_Node(s, N, L, M, R, I, J)$
18     else
19         /*closing (marking as terminal)*/
20         if $s$ is $s_A$ && $s.nnz \geq s.m+1$ then $s.I \leftarrow L$
21         /*For $s_A$, a copy is necessary.*/
22         if $s.nnz \geq s.m+1$ then Mark as CSR
23         else Mark as COO
24     end
25 end
26 return $[P, s_A]$/*Arrays $L, M, R$ can be freed.*/
```

Fig. 2. $COO\_to\_RSB\_s(s_A, I, J)$

matrices (Fig. 8,9), and finally *compression of indices* in RSB_Leaf_Switch (Fig.10). Accordingly, we break down the RSB assembly pseudo code into three listings, called from procedure COO_to_RSB, in Fig. 1.

Procedure $COO\_to\_RSB\_s$ (Fig. 2), performs a cycle, identifying bounds for candidate submatrices. This information is stored in auxiliary arrays L, M, R. A *row pointers* array P is constructed (line 5), kept and returned for later usage. At each iteration, the *largest open submatrix s* (in terms of number of nonzeroes) is selected; then it is analyzed, and either subdivided in quadrants (and marked as *closed node*) or marked as a *closed leaf*. In either case, each cycle *closes* submatrix s and *opens* up to four submatrices. Therefore, the loop iterates a number of times equal to the number of the nodes (both inner and leaf) in the produced quad-tree. In Fig. 3 we present the cutoff function δ_r which decides if subdivision of s should proceed.

Since the input COO arrays are row-major sorted, in order to identify quadrants of s in them, we need to mark, for each row, indices for: the leftmost element of the two left quadrants, the leftmost of the two right quadrants, and the first one after the rightmost of the two right quadrants; that is, pin-point *subrows* in each quadrant. To this end, I and J are scanned in $Subrow\_Split$, and subrows information is stored in the three *row pointers arrays* L, M, R. Row pointers data will be reused when assembling submatrices in CSR. The first invocation of $Subrow\_Split$ requires L, R for the whole A in order to compute the first *middle row pointers* array M. Notice that for any row i of A, $L[i+1] \equiv R[i]$. For this reason, before entering the loop, we pre-compute a single row pointers array P, and set the initial L, R as *pointer aliases* of P. That is, P can serve as L, and aliased after its first element, as R does; in Fig. 2 and 7, we have used "$\overset{p}{\leftarrow}$" to signify pointer aliasing. P is computed by COO_RowP, listed in Fig. 4.

```
/*$WS(=4)$: Word Size of index element, $\mu = 3$ */
if $s_A.N_S \geq s_A.MAX_S$ then return False
if $n \cdot ES \ > 2 \cdot CS$ && $m < 2^{16}$ && $k < 2^{16}$ then return
True
if $(ES\ (2 \cdot n + m) + WS \cdot (m+n)) > \alpha\ CS$ &&$n/m > \mu$
then return True
return False
```

Fig. 3. $\delta_r(m, k, n, CS, ES, WS)$

```
$P[:] \leftarrow$ 0/* fill with zeros*/
for $n \leftarrow 0$ to $nnz-1$ do  $P[I[n]+1] \leftarrow P[I[n]+1]+1$
for $i \leftarrow 0$ to $m-1$ do  $P[i+1] \leftarrow P[i+1]+P[i]$
/*for each $i$, $P[i]$ now has the offset of row $i$ in $I, J$*/
```

Fig. 4. $COO\_RowP(I, J, P, nnz, m)$

After boundaries are identified, and nonzeroes counts are known for each quadrant, at line 17, we invoke the RSB_Split_Node. It will add an *open* leaf submatrix for each non-empty quadrant, and copy the L, M, R arrays in appropriate offsets of the I array. In this way, I is used for storing submatrices rows information, and subsequent invocations of $Subrow\_Split$ will use the L, R arrays recovered from there.

In the case the δ_r does not make s a candidate for subdivision, s gets *closed* as a leaf matrix, and marked to contain data in the CSR or COO format (depending on the available index space; lines 18-23). In the case s is the root node for A (s_A), and fitting CSR arrays ($nnz > m$), L (aliasing P) is copied at the appropriate offset of I, overwriting original row indices (not needed anymore).

After assembling the quad-tree for the s_A, the original J, V arrays storing column indices and values of the matrix coefficients are still unmodified, and ready for being displaced

```
$n_{00} \leftarrow 0$; $n_{01} \leftarrow 0$; $n_{10} \leftarrow 0$; $n_{11} \leftarrow 0$;
for $i \leftarrow$ 0 to $\lfloor (s.m+1)/2 \rfloor$ do
    $M[i] \leftarrow Search(J, L[i], R[i], s.j_0 + \lceil s.k/2 \rceil)$
    $n_{00} \leftarrow n_{00} + (M[i] - L[i])$; $n_{01} \leftarrow n_{01} + (R[i] - M[i])$
end
for $i \leftarrow \lceil (s.m+1)/2 \rceil$ to $s.m-1$ do
    $M[i] \leftarrow Search(J, L[i], R[i], s.j_0 + \lceil s.k/2 \rceil)$
    $n_{10} \leftarrow n_{10} + (M[i] - L[i])$; $n_{11} \leftarrow n_{11} + (R[i] - M[i])$
end
return $[n_{00}, n_{01}, n_{10}, n_{11}]$
```

Fig. 5. $Subrow\_Split(s, L, R, M, J)$

```
Binary search for the smallest $m$ such that $J[m] \geq h$ and
$l \leq m \leq r$
return $m$
```

Fig. 6. $Search(J, l, r, h)$

```
$Q \leftarrow [...]$ /*allocate a submatrix structure for each nonempty
quadrant of s; then for each quadrant $q_{ij}$, set info for
nonzeroes, dimensions, and row,column,nonzeroes offsets
relative to the whole matrix; then copy portions from the
subrow pointer arrays from $L, M, R$*/
if $n_{00} > 0$ then
    $q_{00}.m \leftarrow \lceil s.m/2 \rceil$; $q_{00}.k \leftarrow \lceil s.k/2 \rceil$;
    $q_{00}.moff \leftarrow s.moff + 0$; $q_{00}.koff \leftarrow s.koff + 0$;
    $q_{00}.nzoff \leftarrow s.nzoff + 0$; $q_{00}.nnz \leftarrow n_{00}$;
    $q_{00}.I \overset{p}{\leftarrow} I + q_{00}.nzoff$; $q_{00}.J \overset{p}{\leftarrow} J + q_{00}.nzoff$
    if $q_{00}.nnz > 2 \cdot q_{00}.m + 2$ then
        $q_{00}.I \leftarrow IL[1 : q_{00}.m]$; $q_{00}.J \leftarrow IM[1 : q_{00}.m]$;
    end
end
if $n_{01} > 0$ then
    $q_{01}.m \leftarrow \lceil s.m/2 \rceil$; $q_{01}.k \leftarrow \lfloor s.k/2 \rfloor$;
    $q_{01}.moff \leftarrow s.moff + 0$; $q_{01}.koff \leftarrow s.koff + q_{00}.k$;
    $q_{01}.nzoff \leftarrow s.nzoff + n_{00}$; $q_{01}.nnz \leftarrow n_{01}$;
    $q_{01}.A \overset{p}{\leftarrow} I + q_{01}.nzoff$; $q_{01}.J \overset{p}{\leftarrow} J + q_{01}.nzoff$
    if $q_{01}.nnz > 2 \cdot q_{01}.m + 2$ then
        $q_{01}.I \leftarrow IM[1 : q_{01}.m]$; $q_{01}.J \leftarrow IR[1 : q_{01}.m]$;
    end
end
.../*And so on for $q_{10}, q_{11}$.*/ ...
```

Fig. 7. $RSB\_Split\_Node(s, N, L, M, R, I, J)$

to their destination location. The I array, instead, has been overwritten. For submatrices marked for CSR storage, I already stores a *row pointers array*, which a CSR representation requires. For submatrices marked for COO storage, the relevant subarrays for I could have been overwritten during parent node subdivision, and therefore they should be reinitialized to their original values. Actually, each submatrix node has information on the count of nonzero elements in its own quadrants. Recall, that in RSB_Split_Node, the *nonzero offset* of each submatrix in the quad-tree representation was computed. Now, each submatrix s could be extracted to a temporary storage, row by row, from the original matrix specified in subsequent rows, at the submatrix offset $s.nzoff$ (computed by RSB_Split_Node, in Fig. 7). To keep the shuffling algorithm simple, we have chosen to allocate two temporary J_t and V_t arrays; gather there the displaced rows for coefficients and indices, and copy back to J, V. Since different submatrices should be laid in separate intervals of J and V, the shuffling algorithm can be parallelized on a submatrix basis in a parallel cycle. Once shuffled, the temporary arrays are copied back using a simple OpenMP-parallel wrapper around the standard memcpy ([20]) function .

The shuffling procedures for J (Fig.9) and V (Fig.8) are similar. For V ($COO\_to\_RSB\_V$), only rows shuffling is needed, but for J ($COO\_to\_RSB\_J$), besides shuffling, we need also to adjust indices relative to the submatrix location, and restore indices of I. After the shuffling phase, submatrices are either stored as *fullword* (by default, 32 bit) COO or CSR. RSB (see [5]) allows smaller leaves to have 16 bit coordinate (for COO) or column (for CSR) indices. For this, we use a separate procedure, RSB_Leaf_Switch, operating an *in place* conversion on the arrays of the candidate submatrices.

```
Allocate a temporary vector $V_t$, fitting $V$.
parallel foreach $s \in S$ do
    $V_s \overset{p}{\leftarrow} V_t[s.nzoff]$
    if $s.nnz \geq 2 \cdot s.m + 2$ then
        for $i \leftarrow 0$ to $s.m - 1$ do
            Append subrow $V[s.L[i] : s.R[i]]$ to $V_s$
        end
    else
        for $i \leftarrow 0$ to $s.m - 1$ do
            $l \leftarrow P[s.moff + i]$;$r \leftarrow P[s.moff + i + 1]$
            $l \leftarrow Search(J, l, r, s.koff)$
            $r \leftarrow Search(J, l, r, s.koff + s.k)$
            Append subrow $V[l : r]$ to $V_s$
        end
    end
end
$MEMCPY\_Parallel(V, V_t)$/*$V \leftarrow V_t$*/
```

Fig. 8. $COO\_to\_RSB\_V(s_A, P, V)$

```
Allocate a temporary vector $J_t$, fitting $J$.
parallel foreach $s \in S$ do
    $J_s \overset{p}{\leftarrow} J_t[s.nzoff]$
    if $s.nnz > 2 \cdot s.m + 2$ then
        for $i \leftarrow 0$ to $s.m - 1$ do
            Append subrow $J[s.L[i] : s.R[i]]$ to $J_s$
            Make a CSR row pointer in I, using L, R
        end
    else
        for $i \leftarrow 0$ to $s.m - 1$ do
            $l \leftarrow P[i]$;$r \leftarrow P[i + 1]$
            $l \leftarrow Search(J, l, r, s.koff)$
            $r \leftarrow Search(J, l, r, s.koff + s.k)$
            Append subrow $J[l : r]$ to $J_s$
            if $s.nnz < s.m + 1$/*COO case*/ then
                Set array $s.I$ with value $i$
            else
                Make a CSR row pointer in I, using L, R
            end
        end
    end
    Adjust $s.J$ indices, by subtracting the offset $s.koff$.
end
$MEMCPY\_Parallel(J, J_t)$/*$J \leftarrow J_t$*/
```

Fig. 9. $COO\_to\_RSB\_J(s_A, P, J)$

Note that interleaving shuffling and conversion could save a substantial fraction of memory accesses; however the constructor logic would be much more involved. After this (last) step, the matrix is assembled as RSB and ready for use.

The presented assembly procedure consists of a *serial* stage (*subdivision*), followed by two stages exploiting parallelism (*shuffling* and *conversion*). Initially, we considered to propose a parallel subdivision step. However, we observed that this would require us to use more complicated techniques, and would also entail differences in the computed partitions. For instance, we could have let threads subdivide the matrix concurrently, but non-determinism in the order of subdivision could lead to non-deterministic quad-tree shape/matrix partitioning. In such case, we would have either to accept the

```
parallel foreach leaf node $s$ of quad-tree $s_A$ do
    if Marked for halfword indices then
        if CSR format then
            Convert J to use 16 bit indices, in place
        end
        if COO format then
            Convert I, J to use 16 bit indices, in place
        end
    end
end
```

Fig. 10. $RSB\_Leaf\_Switch(s_A)$

algorithm as non-deterministic (which we did not want), or use complicated *backtracking* techniques to revert unnecessarily subdivided submatrices and an equivalent tree. On the other hand, we have found strategies for the parallelization of the current subdivision algorithm routines (based on fine-grained parallelism) to be problematic regarding synchronization, and therefore shortsighted, in the perspective of many-core computations, expected in the forthcoming computers. Therefore, for the time being, we have chosen a simple serial strategy, and left other enhancements for future developments. Indeed, besides being serial, the subdivision stage faces a growing amount of work, as more subdivisions are performed on a matrix; and thus, it will slow down further, the more threads will participate in the *SpMV* computation (recall line 1 in Fig. 2). Each subdivision of a submatrix s requires (a) the copy of two arrays, (b) $s.m$ binary searches during split, and (c) one array write per search. In the worst case, this involves about $s.m$ random accesses in the binary searches, (which perform non-linear accesses), but the remaining accesses are linear, and could be performed taking advantage of the available prefetching engine on the CPU. Analysis of the complexity of subdivision is beyond the scope of this paper; a gross, pessimistic estimate we could provide for the memory traffic would be up to $o(h \cdot nnz)$ memory writes (where h is the height of the quad-tree). This would be the case where all of the submatrices would fit exactly as CSR: if some were COO, binary searches would be performed on their parent matrices, but with no subsequent row pointers copy (matrices are assigned as COO if they don't fit CSR, with no further subdivision). If some submatrices had rows denser ($s.nnz > s.m + 1$), it would mean that only $O(s.m + 1)$ elements would be moved (out of $s.nnz$).

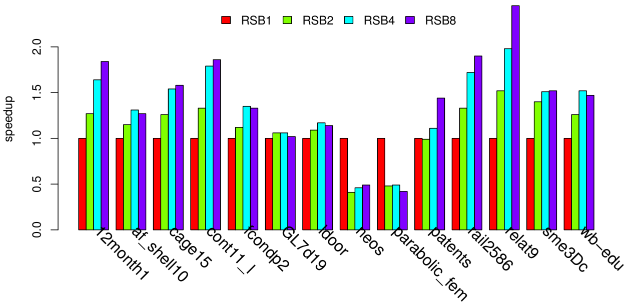


Fig. 11. RSB matrix assembly scaling on **M1**.

The shuffle stage is different: it involves two transfers of contents of arrays V and J; and between m and nnz element moves for I. If not coupled to the copy operation, the index adjustment for J accounts for further, up to $O(nnz)$, accesses; similarly for restoring the I arrays of COO leaves. Similarly, the complexity of the compression stage involves modifications of up to $2nnz$ memory locations (once). Besides the `memcpy`-like operations, when shuffling the COO submatrices, the J array would be *binary-searched* repeatedly for the identification of subrows bounds (after determining bounds for search using P). The same binary-search based algorithm is needed for the CSR submatrices having $s.nnz \le m$ (since the corresponding I subarray would not contain both right and left subrows pointer arrays). For CSR leaves having $s.nnz > s.m$, right and left subrow pointers are recovered from I, subrows in J and V are located, and no search is needed at all. Notice the independence from the quad-tree height (and thus, from the matrix size).

V. Experimental Setup and Methodology

TABLE I
Matrices test-set, obtained from [19].

matrix	symm	r	c	nnz	nnz/r
12month1	G	12471	872622	22624727	1814.19
af_shell10	S	1508065	1508065	27090195	17.96
cage15	G	5154859	5154859	99199551	19.24
cont11_l	G	1468599	1961394	5382999	3.67
fcondp2	S	201822	201822	5748069	28.48
GL7d19	G	1911130	1955309	37322725	19.53
ldoor	S	952203	952203	23737339	24.93
neos	G	479119	515905	1526794	3.19
patents	G	3774768	3774768	14970767	3.97
rail2586	G	2586	923269	8011362	3097.97
relat9	G	12360060	549336	38955420	3.15
sme3Dc	G	42930	42930	3148656	73.34
wb-edu	G	9845725	9845725	57156537	5.81

Our experimental setup is similar to that of [5]: same machines, same compilers, same methodology, but for space reasons, we selected only an *essential* subset of the matrices used there (see Table I). We compiled and ran our codes on machines **M1** (AMD Opteron 2354; 2×4-core CPU; caches: 2×2MB L3, 4× 512KB L2 and 64KB L1) and **M2** (Intel Xeon 5670; 2×6-core CPU; caches: 2×12MB L3, 4× 256KB L2 and 32KB L1), using `-O3` as the only optimization flag, with `icc` v.11 on **M1**, and `gcc` v.4.3 on **M2**. The time samples employed are the *best ones*, after 100 runs for the *SpMV* operation, and 10 runs for the constructor. **M2** is a lightly loaded network server.

VI. Results

For space reasons, we won't be able to present a comprehensive analysis of the constructor performance, and thus we will focus on the most important topics. Our exposition is

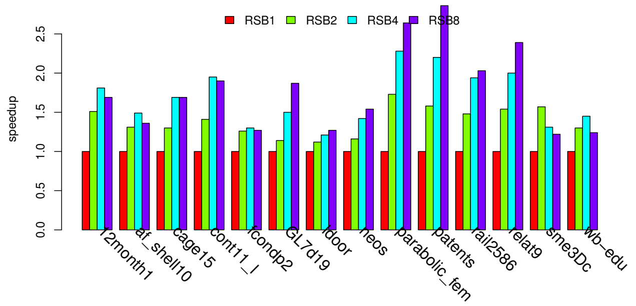


Fig. 12. RSB matrix assembly scaling on **M2**.

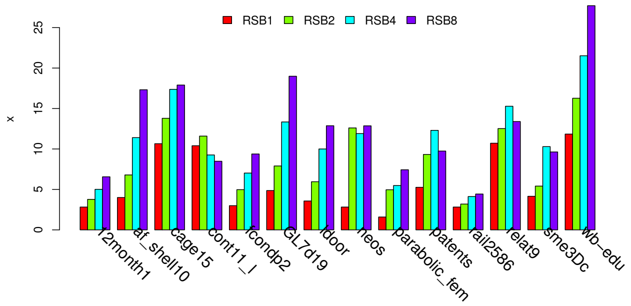


Fig. 13. RSB matrix assembly to *SpMV* time ratio on **M1**.

geared towards iterative methods; here, the affordability of the constructor code is inversely proportional to the number of *SpMV*'s that are expected to be performed after matrix instantiation. Thus, performance profiles for both *SpMV* and construction operations are needed. We will thus present the constructor performance considering two metrics: the number of *SpMV* that are time-equivalent to a constructor run on the given matrix, and the scalability of the constructor with respect to the single core case.

In our previous work (See [5],[4]), using 8 cores on **M1** and **M2**, we have encountered a *SpMV* speedup of up to 5×. In Sec. IV, we have motivated the reasons for keeping a part of our constructor code serial. Therefore, the observed scalability is indeed weak, as depicted in Fig. 11,12. We see that the maximum speedup on both machines is 2.45× on **M1** and 2.86× on **M2**; this is approximately half than observed for the *SpMV*. We notice the best speedup for matrices *relat9* and *rail2586* on **M1**; *patents* and *parabolic_fem* on **M2**. In two cases (*neos* and *parabolic_fem* on **M1**) we notice a slow-down. Due to the increasingly loaded serial stage; in both cases, this happens after a no-subdivisions instantiation, for 1-core (for space reasons, we omit graphs with submatrix counts).

Relating constructor and *SpMV* times, we notice the constructor dominating the *SpMV*, in Fig. 13,14. We observe the maximal ratio for matrix *wb-edu* (up to 52.8× on **M1**, up to 27.7× on **M2**); a minimal one for matrix *rail2586* (from 2.8× to 4.4×, on **M1**). In two cases (matrices *cont11_l*, *patents*), it happens that the constructor and *SpMV* times keep a similar pace (around 10×, on both machines). Indeed, the *SpMV* performance of matrix *cont11_l* does not increase with more cores, and matrix *patents* gets partitioned in the same number of leaf matrices, regardless the cores count. We notice worse ratios for bigger matrices: *cage15*, *wb-edu*, and *GL7d19*. Here, *patents* is big, but it performs *SpMV* exceptionally slow (see [5]).

Let us break down the constructor performance in the serial (*subdivision*) and parallel (*shuffle* and *conversion*–we will include this last one in the shuffle results, for convenience) stages. As discussed in Sec.IV, the subdivision code is expected to perform a number of passes on the input growing with the number of threads available for *SpMV*. In Fig. 15 and 16, we see the *scaling-down* of subdivision performance; we encounter a near-to 5-fold slow-down for matrices *parabolic_fem* and *neos*. It is due to no subdivision being performed in the 1-core case, on them. In the remaining cases, we do not notice more than a 2-fold slow-down.

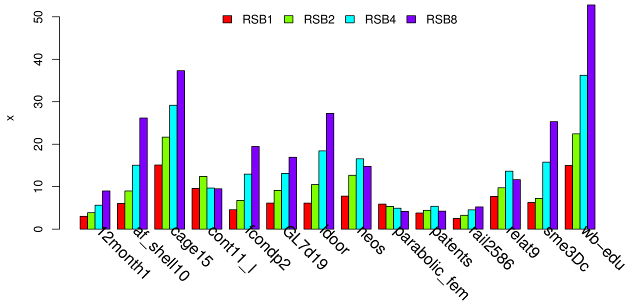


Fig. 14. RSB matrix assembly to *SpMV* time ratio on **M2**.

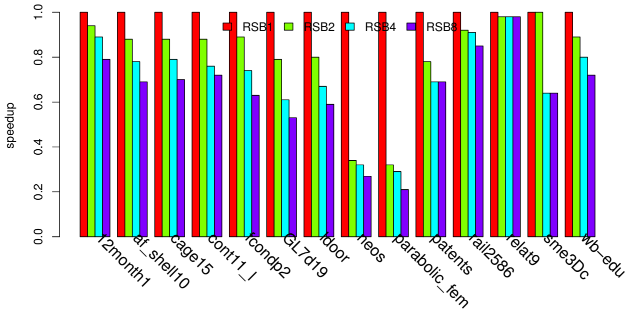


Fig. 15. Subdivision scaling on **M1**.

In Fig. 17,18, we can see the growing gap between the subdivision and *SpMV*. For 1 or 2 cores, this ratio is always lower than 7.0, but for more, it can grow much: for matrix *wb-edu* on **M2**, the subdivision takes 5.1× for 1 core, and up to 42.4 times *SpMV* time, for 8 cores.

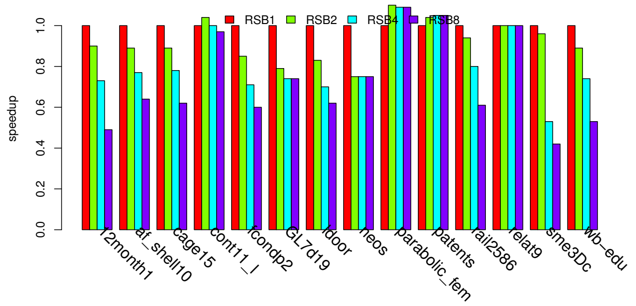


Fig. 16. Subdivision scaling on **M2**.

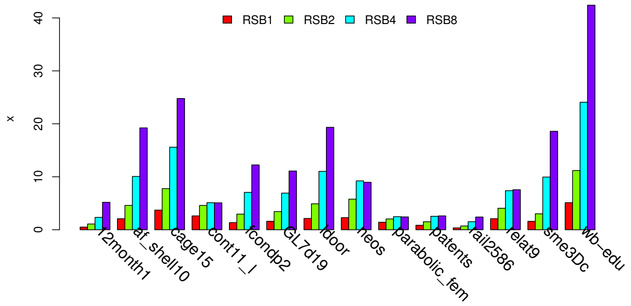


Fig. 18. Subdivision to *SpMV* time ratio on **M2**.

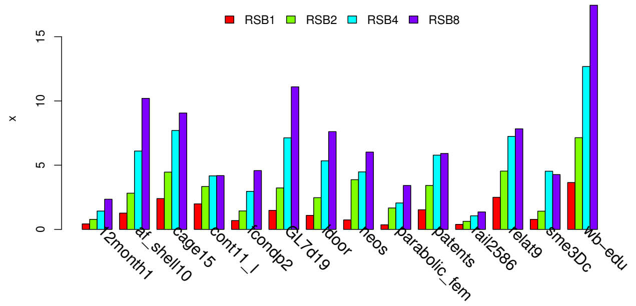


Fig. 17. Subdivision to *SpMV* time ratio on **M1**.

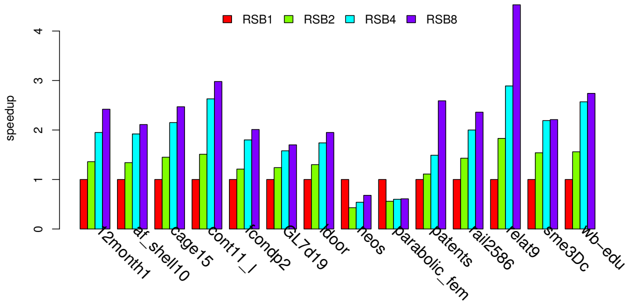


Fig. 19. Shuffle scaling on **M1**.

On the other hand, the *shuffle* stage scales quite regularly on all matrices in the test set; see Fig. 19, 20. Recall that, with higher cores counts, notwithstanding the growing number of submatrices to handle, the shuffle operation moves approximately the same amount of memory locations (see [5] for a discussion on indexing space). As noted in Sec. IV, in the shuffle (comprehensive of index compression) phase, the amount of involved traffic depends on the leaves format; prevalence of COO leaves will trigger more traffic; since compression happens after the copy operations, it contributes to additional traffic.

We also notice that the ratio of shuffle-to-*SpMV* times remains very close, regardless active cores count. This is satisfactory, because it indicates that both the operations scale similarly: see Fig. 21,22. Indeed, both operations seems to be memory bound; shuffle more than *SpMV*, as it doesn't involve floating point operations, which could be slower than integer operations. During stand-alone benchmarking our naive parallel `memcpy` wrapper (MEMCPY_Parallel, used in Sec. IV), we experienced at most $8.4GB/s$ on **M1**, $6.4GB/s$ on **M2**, and speedups respectively up to 3.1 and 2.2. We expect this limit to contribute with a relevant fraction to the shuffle stage.

By comparing, respectively, Fig. 17 to Fig. 21 and Fig. 18 to Fig. 22, we notice that on both machines, the subdivision (serial) stage becomes dominant over the shuffle (parallel) at around 4 active threads. Clearly, this situation is not desirable in the perspective of more computing cores, so we recognize the need for a scaling parallel subdivision stage. Also, by allowing degenerate subtrees (see Sec. IV) input could be scanned repeatedly and generating no new subdivision; this case should also be dealt with.

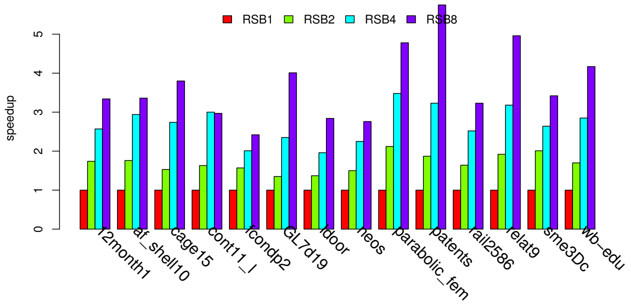


Fig. 20. Shuffle scaling on **M2**.

VII. Concluding Remarks

We have shown a multi-threaded algorithm for the instantiation of RSB matrices out of row major sorted COO arrays.

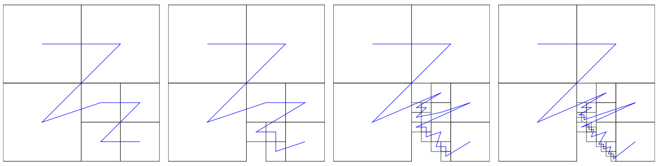


Fig. 23. Recursive subdivisions of matrix *cont11_l* for respectively 1,2,4,8 threads on **M1**. Notice the blue line joining (nonempty) leaf submatrices in the order they are stored in the RSB arrays. Notice that the more threads are active, the finer is the partitioning.

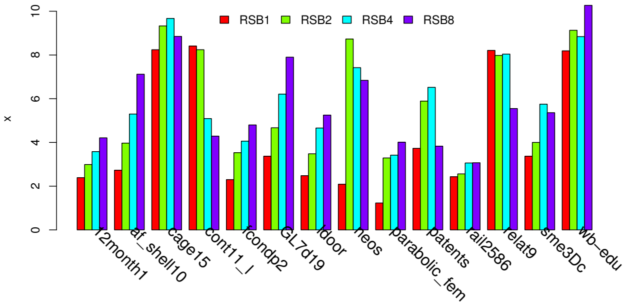


Fig. 21. Shuffle to *SpMV* time ratio on **M1**.

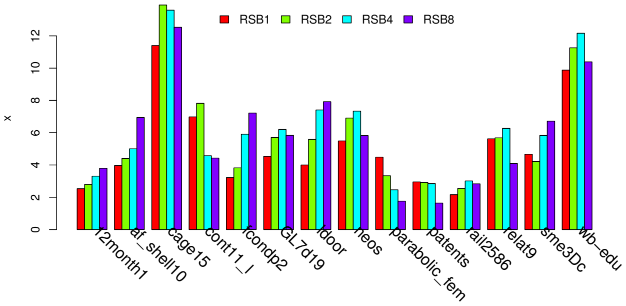


Fig. 22. Shuffle to *SpMV* time ratio on **M2**.

Experimentally, we established that its execution speed seems tightly bound to the peak memory bandwidth; even more than *SpMV*. Our procedure features a serial *subdivision* stage, where binary search and arrays copy operations are predominant, followed by a parallel *shuffle* stage, where arrays are displaced and indices adjusted. The shuffle stage scales smoothly; its performance seems strictly memory bandwidth-bound. While shaping the subdivision stage, we observed that an efficient parallel reformulation of it would require us to modify the definition of our format. We did not want to proceed this way as we wanted it to remain comparable with our earlier work, and so we have decided to leave the subdivision serial, for now. In practice, we observed the constructor-to-*SpMV* time ratio to be 1.6..15.1 times for 1 core, 2.5..22.4/2 cores, and 4.2..52.8/8 cores. Indeed, we have observed that the serial phase begins to dominate the constructor time as soon as at about 4 threads. For this reason, we recognize need of further research to develop a scalable, parallel algorithm to perform the initial subdivision, as this is the key to a scalable RSB matrix constructor. We deem also interesting to study parallel conversion/extraction mechanisms for interfacing to other formats, and consider the performance impact of building preconditioners, while solving linear systems. Of course, a number of trivial but effective optimizations (see Sec. IV) may also be applied.

References

[1] I. S. Duff, M. A. Heroux, and R. Pozo, "The sparse BLAS," Tech. Rep., 2001.

[2] M. Martone, S. Filippone, S. Tucci, M. Paprzycki, and M. Ganzha, "Utilizing recursive storage in sparse matrix-vector multiplication - preliminary considerations," in *CATA*, T. Philips, Ed. ISCA, 2010, pp. 300–305.

[3] M. Martone, S. Filippone, M. Paprzycki, and S. Tucci, "On BLAS operations with recursively stored sparse matrices," in *Proceedings of the International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, September 2010.

[4] ——, "On the usage of 16 bit indices in recursively stored sparse matrices," in *Proceedings of the International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, September 2010.

[5] ——, "Use of hybrid recursive CSR/COO data structures in sparse matrices-vector multiplication," in *Proceedings of the International Multiconference on Computer Science and Information Technology*, October 2010.

[6] Y. Saad, *Iterative Methods for Sparse Linear Systems, 2nd edition*. Philadelphia, PA: SIAM, 2003.

[7] A. Buluc, J. T. Fineman, M. Frigo, J. R. Gilbert, and C. E. Leiserson, "Parallel sparse matrix-vector and matrix-transpose-vector multiplication using compressed sparse blocks," in *SPAA*, F. M. auf der Heide and M. A. Bender, Eds. ACM, 2009, pp. 233–244.

[8] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. Philadelphia, PA: SIAM, 1994.

[9] D. T. Nguyen, *Finite Element Methods: Parallel-Sparse Statics and Eigen-Solutions*. Springer US, 2006.

[10] V. Strassen, "Gaussian elimination is not optimal," *Numerische Mathematik*, vol. 13, no. 4, pp. 354– 356, 1969.

[11] G. von Fuchs, J. R. Roy, and E. Schrem, "Hypermatrix solution of large sets of symmetric positive-definite linear equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 1, no. 2, pp. 197 – 216, 1972.

[12] A. C. McKellar and E. G. Coffman, Jr., "Organizing matrices and matrix operations for paged memory systems," *Commun. ACM*, vol. 12, no. 3, pp. 153–165, 1969.

[13] J. R. Herrero and J. J. Navarro, "Hypermatrix oriented supernode amalgamation," *The Journal of Supercomputing*, vol. 46, no. 1, pp. 84–104, Oct. 2008.

[14] B. Vastenhouw and R. H. Bisseling, "A two-dimensional data distribution method for parallel sparse matrix-vector multiplication," *SIAM Review*, no. 47, pp. 47–95, 2005.

[15] R. Fischer, M. Ast, J. Labarta, and H. Manz, "A dynamic task graph parallelization approach," in *Proceedings of IASS-IACM-2000: Fourth International Colloquium on Computation of Shell and Spatial Structures, June 4-7, 2000 in Chania-Crete, Greece*.

[16] A. Pinar and C. Aykanat, "Sparse matrix decomposition with optimal load balancing," in *High-Performance Computing, 1997. Proceedings. Fourth International Conference on*, 1997, pp. 224 – 229.

[17] P. Gottschling and D. Lindbo, "Generic compressed sparse matrix insertion: algorithms and implementations in MTL4 and FEniCS," in *POOSC '09: Proceedings of the 8th workshop on Parallel/High-Performance Object-Oriented Scientific Computing*. ACM, 2009, pp. 1–8.

[18] D. E. Knuth, *The art of computer programming, volume 1 (3rd ed.): fundamental algorithms*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1997.

[19] T. Davis, "University of Florida sparse matrix collection," *ACM Transactions on Mathematical Software, to appear*, 2010.

[20] "Standard for information technology— portable operating system interface (POSIX) (IEEE std 1003.1)," 2008.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 327–335

ISBN 978-83-60810-27-9
ISSN 1896-7094

Use of Hybrid Recursive CSR/COO Data Structures in Sparse Matrix-Vector Multiplication

Michele Martone, Salvatore Filippone,
Salvatore Tucci
University of Rome
"Tor Vergata", Italy

Paweł Gepner
Intel Corporation

Marcin Paprzycki
University of Rome
"Tor Vergata", Italy
Polish Academy of Sciences, Poland

***Abstract*—Recently, we have introduced an approach to basic sparse matrix computations on multicore cache based machines using recursive partitioning. Here, the memory representation of a sparse matrix consists of a set of *submatrices*, which are used as leaves of a *quad-tree* structure. In this paper, we evaluate the performance impact, on the Sparse Matrix-Vector Multiplication (*SpMV*), of a modification to our *Recursive CSR* implementation, allowing the use of multiple data structures in leaf matrices (CSR/COO, with either 16/32 bit indices).**

I. Introduction

It IS known that computations with sparse matrices incur very poor memory performance: indirect addressing causes unpredictable run-time dependencies in memory read/write access; memory access has poor data locality (just to name a two key aspects; see also [1], [2], [3]). To address these issues, recently, we have proposed a *recursive* approach to sparse matrix representation. In [4] we have outlined the proposed method and reported initial experiments with the *SpMV* operation. In the follow-up [5], we have evaluated its performance for the triangular solve and *SpMV* for symmetric matrices. Experimental results lead us to modify the storage scheme in order to reduce the indexing overhead. Encouraging results of an approach employing 16-bit indices have been reported in [6]. Here, we continue investigations leading to the development of methods that can reduce impact of indirect addressing, by reducing the memory traffic incurred in accessing index data. Specifically, we employ *index compression*, and *diversify* the representation of *leaf submatrices* with the intent of raising the floating point performance of the *SpMV* by saving memory bandwidth.

Proceeding, we outline the RCSR storage format with index compression in Section II. Next, we describe modifications to the sparse matrix representations in Section III. Setup for performed experiments can be found in Section IV, while in Section V we analyze the obtained results.

II. The recursive storage format and index compression

We (logically) organize a sparse matrix as a *quad-tree* structure, with nodes consisting of submatrices arising from a recursive partitioning into quadrants. While intermediate nodes are used only as a pointer structure, leaf nodes hold actual subarrays with index and numerical values. The *SpMV* algorithm described in [5] is independent from the actual format of leaf matrices. It only assumes a *coarse* recursive partitioning in leaf submatrices. Similarly to blocking used in dense matrix computations, submatrices at leaf level should be *sized* (in terms of their *memory footprint* during the *SpMV*) in relation to the *cache sizes of the machine*.

In this context, we have investigated a variation to the leaf matrices format, obtained by converting some of the *Compressed Sparse Rows* (CSR) leaves of a matrix to use 16 bit column indices (and thus, reducing the memory traffic). As motivated in Section I (and in the literature; e.g.: [1]), index compression techniques are particularly effective with many active cores. Here, techniques which may not be optimal on a single core (because of a slight memory-bandwidth-to-computation trade-off, in the form of pointer arithmetics), may show their potential when working with multiple cores (where the memory traffic is heavier). As a motivation of our "16-bit" approach, we observe that after partitioning a large sparse matrix (in the RCSR format), it is likely to have many of the leaf submatrices *dimensioned* less than 2^{16}. Thus, using a 16 bit (*halfword*) index type in their CSR *column indices* arrays is possible, and could lead to savings in memory traffic. We name this variant RCSRH. Obviously, for matrices dimensioned less than 2^{16}, the conversion to RCSRH is possible for all submatrices. The outcome of our experiments (documented in [6]) was encouraging: using halfword indices by itself yielded up to a 25% floating point speedup (with a saving in memory usage up to a 16%) on unsymmetric matrices, and 30% on symmetric ones. However, in a number of cases, the RCSRH variant was not helpful. One of the perceived reasons was that CSR itself does not always fit into leaf submatrices, and thus we have decided to convert some leaf matrices to the *COOrdinate* (COO) format. Let us discuss this change with more detail.

III. Recursive storage format with CSR and COO leaves

In this section, we motivate quantitatively why and when storing some submatrices as COO instead of CSR could reduce index overhead, and the way we have chosen to use COO to enhance RCSR.

A matrix is stored in the RCSR format as a quad-tree structure with CSR ([7, Section. 4.3.1]) submatrices at the leaf level of a *recursive bipartitioning* (see [4]). To store an $r \times c$ matrix with n nonzeroes in CSR, we use an array JA (of size n) with *column indices*, and a *row pointers* array PA (of size $r+1$), referencing *rows* in the JA array. Array JA stores column indices for nonzeroes in a *row-major* order. The array of coefficients (VA) is laid in the same order as JA. To store a matrix in a plain COO format, two n-sized arrays for (row,column) indices (IA,JA) are required. By denoting as $I(r,n)$ the index space requirements for an $r \times c$ matrix (with n nonzeroes) instance we have $I_{CSR}(r,n) \doteq 4(r+1)+4n$ and $I_{COO}(r,n) \doteq 4n+4n$ bytes. Let us call CSRH the CSR format implementation with 16 bit JA indices, and COOH, a COO format implementation with 16 bit IA and JA indices. For these variants, we have $I_{CSRH}(r,n) \doteq 4(r+1)+2n$ and $I_{COOH}(r,n) \doteq 2n+2n$ bytes. This means that for some values of (r,n), COO/COOH would use less indexing space than CSR/CSRH; specifically, $I_{COO}(r,n) < I_{CSR}(r,n)$ when $n < r+1$, and $I_{COOH}(r,n) < I_{CSRH}(r,n)$ when $n < 2r+2$. For this paper, we modified the matrix constructor code to use CSRH whenever a CSR submatrix is dimensioned less than 2^{16}. Similarly, we use COOH whenever a COO submatrix is dimensioned less than 2^{16}; we choose to use COO when $n < r+1$. We adopt COO/COOH as row-major sorted (so we have the same memory access pattern of CSR for JA and VA arrays). In [4] and [6], we have described the *cutoff* function δ as our heuristic regulating subdivision into submatrices; in this paper, we use slightly differing matrix assembly criteria. While we still use the δ_h function from [6], we limit subdivisions by forcing each submatrix not to use more indexing space than a *fullword* COO storage of it would require. The other rules for subdivision are still the same as imposed by δ_h. Please refer to [8] for a full discussion on the new constructor layout. We call the hybrid format resulting from these modifications *Recursive Sparse Blocks* (*RSB*).

IV. Experimental Setup and Methodology

In order to compare the new approach with previously documented experiments using RCSR format (see [6]), we measured performance on the same test set of 36 matrices: 12 of them are symmetric (See Table I), 12 are square unsymmetric (See Table II), and 12 are non square (See Table III). For readability reasons, in Sec. V we left matrices with less significant results (marked with an asterisk (*), in the tables) out of the plots; so the commentary of them is indirect. Furthermore, we have used the same two machines (summarized in table IV). Recall, that **M2** is a lightly loaded network server, while **M1** is a dedicated machine.

For each *matrix/cores* sample, we ran our RSB code, performing 100 times the *SpMV* operation and report the best result. However, timing variation was below 5%, so our results were consistent. We measured timings using the POSIX ([9]) `gettimeofday()` function. Figures in section V depict results, expressed in MFlops (millions of floating point operations per second). Conventionally, we counted 2 Flops per nonzero element for non-symmetric matrices, and 4 for symmetric. We use double precision arithmetic (C's `double` type). Our measurements were performed with *hot caches*; that is, we did not flush deliberately cache contents between subsequent *SpMV*'s; therefore, to avoid artificially high results, all measurements were performed on matrices not fitting entirely in the caches.

TABLE I
SYMMETRIC MATRICES

matrix	r	c	nnz	nnz/r
af_shell10	1508065	1508065	27090195	17.96
BenElechi1	245874	245874	6698185	27.24
bone010	986703	986703	36326514	36.82
crankseg_1	52804	52804	5333507	101.01
ct20stif	52329	52329	1375396	26.28
F1	343791	343791	13590452	39.53
fcondp2	201822	201822	5748069	28.48
kkt_power	2063494	2063494	8130343	3.94
ldoor	952203	952203	23737339	24.93
mip1*	66463	66463	5209641	78.38
nd24k	72000	72000	14393817	199.91
s3dkq4m2	90449	90449	2455670	27.15

TABLE II
GENERAL SQUARE MATRICES

matrix	r	c	nnz	nnz/r
atmosmodl	1489752	1489752	10319760	6.93
av41092	41092	41092	1683902	40.98
cage15	5154859	5154859	99199551	19.24
lhr71	70304	70304	1528092	21.74
patents	3774768	3774768	14970767	3.97
raefsky3	21200	21200	1488768	70.22
rajat31	4690002	4690002	20316253	4.33
rma10*	46835	46835	2374001	50.69
sme3Dc	42930	42930	3148656	73.34
torso1	116158	116158	8516500	73.32
venkat01	62424	62424	1717792	27.52
wb-edu	9845725	9845725	57156537	5.81

TABLE III
GENERAL NON SQUARE MATRICES

matrix	r	c	nnz	nnz/r
12month1	12471	872622	22624727	1814.19
c8_mat11_I	4562	5761	2462970	539.89
cont11_l	1468599	1961394	5382999	3.67
diego-MM-573x230k	573286	230401	41694697	72.73
GL7d19	1911130	1955309	37322725	19.53
neos*	479119	515905	1526794	3.19
rail2586	2586	923269	8011362	3097.97
rel9	9888048	274669	23667183	2.39
relat9	12360060	549336	38955420	3.15
Rucci1	1977885	109900	7791168	3.94
spal_004	10203	321696	46168124	4524.96
tp-6	142752	1014301	11537419	80.82

Our codes were compiled with the Intel `icc` version 11 on **M1**, and `gcc`, version 4.3 on **M2**. In Section V-C we compare our results to that obtained with a publicly available CSB prototype ([2]). On both machines we compiled it using the Cilk++ compiler; version (“`Cilk Arts build 8503`”), based on the `gcc` (*GNU C Compiler*), v.4.2.4. To unify the

TABLE IV
TEST MACHINES.

machine	model	cpus× cores	data caches
M1	Intel Xeon 5670 6-Core 2.93GHz	2×6	2xL3,2x6xL2,2x6xL1: L3:12MB/16-w/64B L2:256KB/8-w/64B L1:32KB/8-w/64B
M2	AMD Opteron 2354 Quad-Core 2.2GHz	2× 4	2xL3,2x4xL2,2x4xL1: L3:2MB/32-w/64B L2:512KB/16-w/64B L1:64KB/2-w/64B

test environment, all codes were compiled using the `-O3` flag only (besides the OpenMP enabling flags).

V. RESULTS

We structure the analysis of results as in [6]. Note that, for brevity, we sometimes reference as RSB-k the k-threaded RSB. In most cases we start by commenting the 8 threaded performance, and proceed from discussing the particularly problematic cases to the best performing ones.

A. Results, Unsymmetric Matrices

For the unsymmetric matrices on **M1**, we observe an improvement when switching from RCSR to RSB in nearly all of the test set matrices; up to 67% on square ones, and up to 33% on non square ones (Fig. 1,2).

The only matrices "suffering" from the switch are: square *av41092* and *raefsky3* (Fig. 1), non square *c8_mat11_I* and *diego-smtxMM-573x230k*, and two borderline cases: *rail2586* and *sme3Dc*.

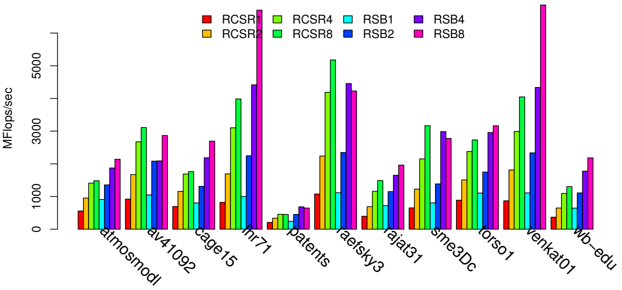


Fig. 1. Unsymmetric *SpMV* on **M1**, square matrices.

On machine **M2** (Fig. 3,4), we see improvements up to 128% for square matrices, and 65% for non square ones, and a single case of a performance drop: a 3% fall for the non square matrix *cont11_l*.

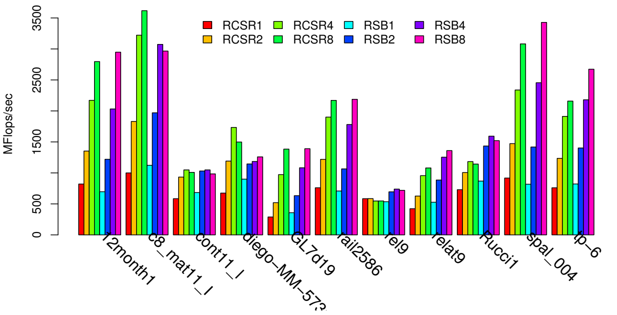


Fig. 2. Unsymmetric *SpMV* on **M1**, non square matrices.

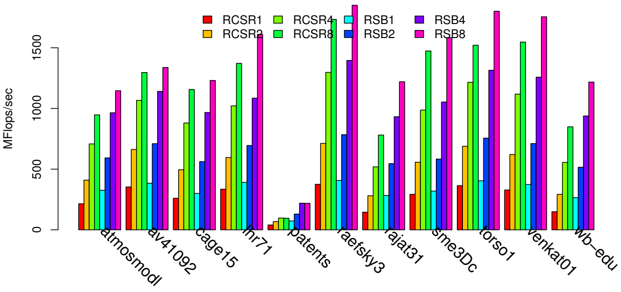


Fig. 3. Unsymmetric *SpMV* on **M2**, square matrices.

In Fig. 6,7,8,9 we observe index usage saving almost always. Out of 24 non-symmetric matrices, we experience three cases where index usage raises: square matrix *patents* (Fig. 6,8) and non square matrices, *rel9*, *relat9* (Fig. 7,9). We note, however, that the effect of RSB is actually an improvement of the performance on these matrices, notwithstanding the increased index usage. Among these matrices, problematic cases remain: *patents* performs better, but continues scaling poorly, (remaining the "slowest" of our entire test set); *relat9* suffers from poor scaling, too (especially on 8 cores **M2**); *rel9* continue not scaling at all.

These matrices have a feature in common: a very low nonzeroes/row elements ratio: 2.39 for *rel9*, 3.15 for *relat9* (see Table III) 3.97 for *patents* (see Table II). Although for such matrices one cannot expect high efficiency for either CSR or COO formats, we have realized why this is also the case for our recursive format (see [6], [5]), so now we present only the particular case for RSB.

Although very poorly performing, *patents*, actually scales up to 4 threads. In facts, *patents* is assembled in 37 COO leaves, regardless the thread count. When working with 8 threads, we observe that scaling is inhibited: this means that particular partitioning leaves a number of threads starving, while most of row intervals are *locked* by other threads. This is a situation occurring when the thread count approaches the number of submatrices in disjoint row intervals (see Fig. 5); and thus threads contend for available row intervals to operate on. In the current formulation of RSB, further partitioning of this matrix is not allowed, for it does not have enough

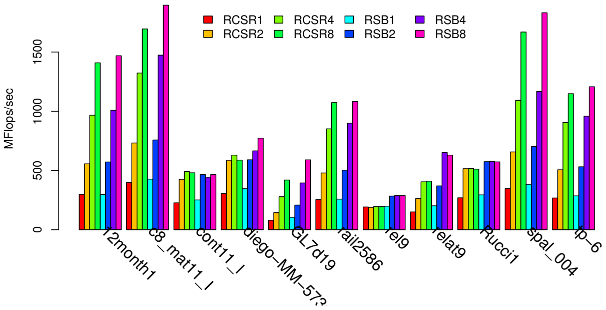


Fig. 4. Unsymmetric *SpMV* on **M2**, non square matrices.

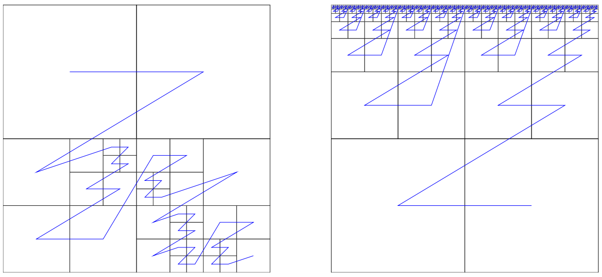


Fig. 5. On the left, matrix *patents* as partitioned on **M1**. On the right (widened, for viewing convenience) *diego-smtxMM-573x230k* on **M1**. Both in RSB format.

nonzeroes per row. On **M2**, the case for *patents* is similar: while on 1,2,4,8 threads, the matrix is partitioned respectively into 13,25,37,37 COO leaves.

The cases of *rel9* and *relat9* (Fig. 2,4) are similar. Since *relat9* has a little higher nonzeroes/row count than *rel9*, it succeeds in scaling in a limited way (up to 30 COO leaves, on both machines), but *rel9* gets partitioned in 7 leaves only, in all cases. Therefore, for *rel9*, more than 2 threads contend for row locking on 7 submatrices, with no possible scaling. Notice, however, that RSB is capable of allowing dual threaded parallelism in these *very sparse* cases, whereas RCSR was not.

The cases we have just discussed are worst/limit cases, and as such are not the primary target of our modifications, so we tolerate them here, and use them for comparison means.

Although quite different, two matrices (*sme3Dc*, *raefsky3*) suffer similar problems, when instantiated as RSB on **M1**. That is, while they are well-performing on RCSR and loosing index overhead from the RSB switch, they also get partitioned into less leaves, giving rise to the same *SpMV* scalability problem. In facts, while RCSR-8 partitions these matrices respectively into 115 (113 CSRH, 2 COOH) and 94 (CSRH) leaves, RSB-8 produces 16 (all CSRH) and 13 (11 CSRH, 2 COOH) leaves.

Given the lock-based nature of our *SpMV* algorithm, and the distribution of submatrices, RSB-8 suffers from contention problems on both matrices. It is interesting to note that on **M2**, these matrices get subdivided respectively in 115 and 94 leaves, and we observe in Fig. 3 that this suffices to scale and experience, respectively, a 7% and a 6.7% improvement. Index overhead shifts from 4.44. to 2.55 bytes/nonzero for *sme3Dc*, and from 4.28 to 2.34 bytes/nonzero for *raefsky3*.

Matrix *av41092* on **M1** experiences the same problem *sme3Dc* and *raefsky3* did: insufficient partitioning. While **M1** partitions this matrix in 10 (9 CSRH, 1 COOH) submatrices only, **M2**, due to its smaller caches, partitions it in 72 leaves (64 CSRH, 8 COOH). So, the halving in index overhead experienced on **M1** (from 4.65 to 2.27 bytes/nonzero) could not bring advantage to RSB-8, while on **M2**, the 42% index saving (from 4.5 to 2.61 bytes/nnz) allows for scaling and a modest 3% performance increase.

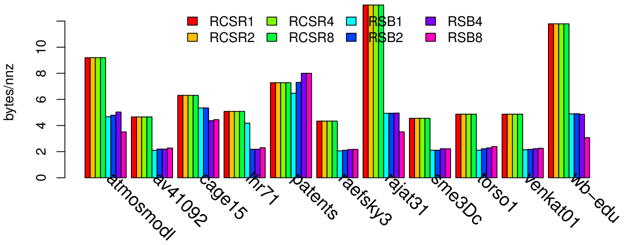


Fig. 6. Index storage requirement (in bytes) per nonzero on **M1** (square matrices).

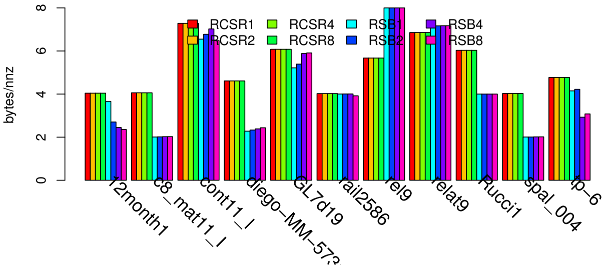


Fig. 7. Index storage requirement (in bytes) per nonzero on **M1** (non square matrices).

The remaining three cases with a missing improvement are non square matrices *c8_mat11_I*, *diego-smtxMM-573x230k*, and *rail2586* (Fig. 2). Matrix *c8_mat11_I*, alike to the matrices we have seen before on **M1**, suffers from poor partitioning, here: RSB partitions it in respectively 1,4,10,13 leaves for 1,2,4,8 threads. On 8 threads, the 13 leaves are not enough to ensure the parallel operation of all the threads, thus leaving some of them *starving*. Similarly to the previous cases, **M2**

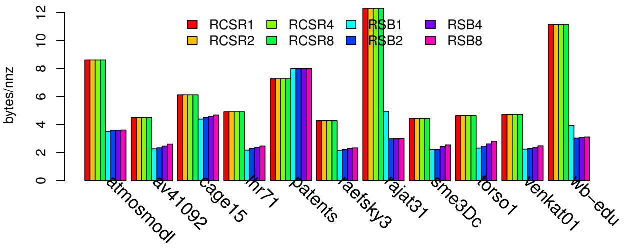


Fig. 8. Index storage requirement (in bytes) per nonzero on **M2** (square matrices).

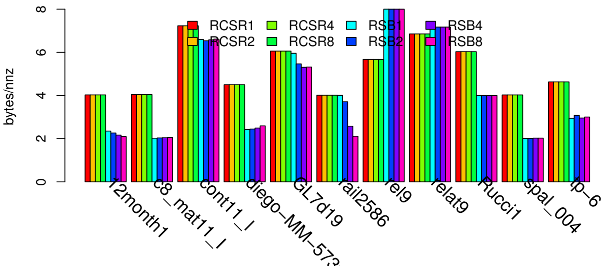


Fig. 9. Index storage requirement (in bytes) per nonzero on **M2** (non square matrices).

divides the matrix in much more leaves, thus avoiding the scaling problem.

The case for matrix *diego-smtxMM-573x230k* is different (and interesting). On **M1**, this matrix performs best as RSCR, while on **M2**, best as RSB. On both machines, though, while not scaling up to 8 threaded RCSR, it scales (although very *slightly*) for RSB, up to 8, but poorly. Poor scaling is evident: RSB-8 on **M1** is only 88% faster than RSB-1; on **M2**, only 123%. By looking at the number of submatrices, we could not say their number is too low. It is only after inspecting the distribution of submatrices (see Fig. 5), that we notice a big unbalance: actually, most of the submatrices are located on the top of the matrix, and it seems that RSB arranged submatrices in "block rows". Given the row-lock-based nature of our *SpMV* algorithm, such a distribution is enough to destroy the parallelism of the computation on this matrix. Here, after completing the bigger-dimensioned submatrices across various row intervals of the matrix, threads will try to acquire a lock on the intervals located on the upper border, with no success for most of them: only a few of them will be able to work at a time, on the upper submatrices. Contention will last during the whole computation for most of the threads, then, because our current *SpMV* algorithm has no mechanism for concurrent update of a single subvector.

Matrix *rail2586* constitutes another special case. For being *wide*, it fits particularly well when stored in a row-oriented storage as CSR. However, for having its nonzeroes scattered quite uniformly around the matrix, it would end up having very sparse submatrices, if it had not as much as 3097 nonzeroes per row, globally. But it happens that for being so wide, the proper introduction of CSRH leaves is only possible after a certain number of subdivisions. On **M1** (Fig. 12), it happens that there are not enough subdivisions for switching much of the submatrices to CSRH. So, the use of RSB for *rail2586* on **M1** does not lighten the index overhead significantly (it remains at about 4 bytes per nonzero), and the performance remains the same (notwithstanding the submatrices reduction: from RCSR's 352, to RSB-8's 55). For architectural reasons, RSB on **M2** ends up partitioning the matrix more finely, and thus falling to switch to CSRH in 335, out of the 352 leaves of RSB-8. The matrix is thus partitioned in number of matrices which is the double of RCSR's. However, in this case, the performance gain expected from RSB is negligible: less than 1%. We conjecture that the *flat* distribution of submatrices in the matrix, and its considerable width, cause a considerable overhead to the memory subsystem, which in turn is forced to continuously load elements from the right hand side vector, which would barely fit in the cache.

We notice that some matrices gain a considerable speedup from the RSB representation: *rajat31* (56%), *lhr71* (17%), *torso1* (18%) on **M2** (Fig. 4), *venkat01* (67%), *cage15* (50%) on **M1** (Fig. 2), *wb-edu* on both (68% on **M1**, 43% on **M2**). The assembled instances of these matrices as RSB differs from RCSR, for the relevant number of COO/COOH submatrices. On **M2**, *rajat31* gets partitioned in 1534 leaves, of which 896 COOH, and 126 COO; *wb-edu* in 4336 leaves, of which 2511 COOH, 254 COO; *torso1* in 357 leaves, of which 39 COOH; *lhr71* in 87 leaves, of which 34 COOH. In all these cases, index overhead is cut down approximately in a half. On matrices *rajat31* and *wb-edu*, index overhead falls down respectively from 12.3 to 3 bytes/nnz and from 11.15 to 3.12 bytes/nnz. This means that RSB *cures* cases where RCSR alone produced subdivisions abusing from CSR leaves; that is, producing CSR leaves with less nonzeroes than rows. The case for matrix *cage15* on **M1** is alike, in that it gets partitioned in 751 leaves, 132 of which are COO, 316 COOH, 6 CSR, 297 CSRH. With RSB, this configuration of *cage15* saves approximately 30% index overhead (from 6.3 bytes/nonzero), which is not much compared to other cases. So probably, the gain is due to the *fuller* submatrices (RSB-8 assembles 751 of them; RCSR as much as 4457). Performance gain on *torso1* is probably due only to index overhead saving: in RSB-8 on **M1**, it gets partitioned in 59 CSRH leaves only, (from 176 CSR), saving 64% of indexing overhead (from 4.6 bytes/nonzero, Fig. 6), which is quite good.

B. Results, Symmetric Matrices

Bar plots in Fig. 10 and 11 present the comparative performance results of RCSR and RSB for symmetric matrices. We observe performance enhancements nearly in all cases. There are three exceptions, though: *crankseg_1*, *ct20stif*, *F1* on **M1**.

We comment these exceptions first, and the remaining cases next.

On **M1**, matrix *F1* in RSB (Fig. 10) does not scale from 4 to 8 threads. On less than 8 threads, *F1* is processed faster with RCSR; e.g.: with 1 thread, *F1* gets partitioned by RSB in 10 submatrices only, all fullword CSR. But with 8 threads, RSB partitions *F1* in 72 leaves, of which 70 are CSRH and 2 COOH. With RCSR, a number of 573 leaves were obtained, which is much more. Given the higher number of subdivisions, load balancing in RCSR ran for sure smoother, while RSB did fall in a lock contention problem here, it seems. Please recall (See [5]) that our symmetric *SpMV* implementation variant incurs in a higher locking overhead than unsymmetric. On **M2**, the situation is almost reversed: for 8 cores, it is RSB that partitions *F1* in more leaves (573: 504 CSRH and 69 COOH), while RCSR divides the matrix in 278 leaves *only*. The index overhead of RCSR is quite high on *F1*: 5.08 bytes/nnz on **M2**, 5.4 on **M1**; on RSB it is always less than this, on both machines. However, the RSB index overhead depends on the threads count: on **M2** (Fig. 13) with more threads, the overhead tends to grow too, from 2.6 to 3.3 bytes/nnz, suggesting that further subdivisions could degrade performance. On the other hand, on **M1**, when going from 1 to 8 threads, this overhead decreases from 4.25 to 2.52 bytes/nnz (Fig. 12). These observations suggest us that the performance improvement over 1-core RCSR (on both **M1** and **M2**) is due to less index overhead, which itself is a consequence of less submatrices fragmentation. We believe that some *optimum partitioning* for 8 cores *F1* is between all of these four instances of RCSR/RSB on **M2**/**M1**; that is, the algorithm should have partitioned *F1* less coarsely on RSB/**M1**, more coarsely on RCSR/**M1**, and so on.

The cases for matrices *ct20stif* and *crankseg_1* (still on **M1**) are different. With *ct20stif* we observe that 2-threaded RSB fails from partitioning, thus cutting off two-cores parallelism completely (Fig. 10). On more cores the heuristic succeeds partitioning the matrix, but too coarsely to gain a sufficient workload balance. Please note that this matrix is among the smallest in our test set ($1.3 \cdot 10^6$ nonzeroes), stressing the limit of our rule of thumb (*sizing* matrices around the cache sizes). On both **M1** and **M2** machines, index usage for *ct20stif* keeps very low: for RSB it ranges from 2.27 to 2.52 bytes per nonzero, coming from RCSR's approximate 4.5. With an analogy to the previous case, on machine **M2**, partitioning is finer than on **M1**, from the single thread case on (1-threaded RSB partitions *ct20stif* to 7 submatrices), and an adequate workload balancing follows. Thus with *ct20stif* on **M2**, we do not loose the 8 threaded case, and RSB's performance is higher than RCSR's. Here, the sparser leaf submatrices are assembled as COOH (2 out of 7 on 8 cores **M1**, 2 out of 60 on **M2**), the remaining ones in CSRH. Notice that both *F1* and *ct20stif* matrices had more than 25 nonzeroes/row, which is quite sufficient to achieve good results with RCSR/RSB. Matrix *crankseg_1* is a little bit sparser (10 nnz/row). It suffers from the same *poor partitioning* problem on **M1**, having respectively 3,10,16,39 leaves for 1,2,4,8 threads, and loosing 30% of performance on 8 threads. On the other hand, on **M2**, matrix *crankseg_1* performs quite well, achieving an improvement to RCSR. The improvement itself is about 21% on 8 cores, when the matrix is partitioned in 37 COOH and 202 CSRH submatrices.

After having discussed the problematic cases, let's look at the remaining ones.

In one case there is almost no change: *nd24k* on **M1** (Fig. 10). Here, RCSR partitions the matrix in 503 CSR leaves, RSB in 87 CSRH leaves. The index overhead (Fig. 12) gets almost halved (from 4 bytes bytes/nonzero). We are not aware of the reason for the missing performance increase, here, but note that this is our symmetric matrix with the higher nnz/row count (199, see table I). On **M2** (Fig. 11), the same matrix witnesses a slight (5%) speedup, while being partitioned by RSB in 503 (all CSRH, except 5 COOH ones) pieces, and 278 ones by RCSR. The index overhead (Fig. 13) similarly to that of **M1**, halves from RCSR (4.2 bytes/nnz) to RSB (2.1 bytes/nnz). We conjecture that the 87 leaves on **M1** somehow limited parallelism, but we would need to investigate further to confirm this.

In one case, on **M1**, RSB performance boosts up as high as 66%, when compared to RCSR: it is for matrix *s3dkq4m2* (Fig. 10). Here, RCSR partitions in 127 leaves, while RSB in 15 only (8 CSRH, 7 COOH). We observe the index overhead (Fig. 12) is almost halved, switching from RCSR to RSB (for > 1 threads). We deem that this speedup is due to a case in which the matrix offers caching potential (the whole result vector and a matrix portion): on **M2**, where the L3 cache is considerably smaller than on **M1**, the performance of *s3dkq4m2* improves by only 2%, passing from 63 leaves of RCSR to 120 CSRH and 7 COOH leaves of RSB. Performing a run with *cold caches* (that is, making sure that any location caching the matrix or the involved vectors gets overwritten between each *SpMV*), on **M1** the performance of RSB is approximately 7% lower, while on **M2** it made no difference (and the boost becomes 55%, rather than 66%). Please note that the smallest symmetric matrix in the test set is not *s3dkq4m2* but *ct20stif*, which we have commented before.

When switching from RCSR to RSB on **M2** (Fig. 11), we observe speedups in all cases. Probably, L3 cache on **M2**, smaller than on **M1**, induced too coarse partitionings, thus limiting the scalability of our symmetric *SpMV*.

We can now comment the cases where the biggest improvement was observed: *af_shell10* (30%), *BenElechi1* (29%), *bone010* (24%), *fcondp2* (20%), *ldoor* (19%) on **M1** (Fig. 12), and *fcondp2* (28%), *crankseg_1* (21%), *ldoor* (16%), *F1* (12%) on **M2** (Fig. 13). For *af_shell10* on **M1**, we observe that RSB instantiates 255 submatrices (192 CSRH, 48 COOH, 15 COO), while RCSR used to instantiate 1534 CSR leaves. This matrix is also the one to experience the higher saving in index overhead: from 5.22 to 2.5 bytes per nonzero (more than 50%, Fig. 12). Matrix *BenElechi1* gets partitioned by RSB in 63 leaves: 32 CSRH, 30 COOH, 1 COO; by RCSR in 382 CSR matrices. Index usage (Fig. 12) halves: from 4.66 to 2.25 bytes per non zero. Similarly to the *af_shell10* case, we experience a smaller number of leaf matrices, a more appropriate leaf matrix selection, and a consequent reduction in indexing overhead. On **M2** (Fig. 11), the same matrix improves only by 1.6%. By looking at its partitioning, we notice that it is partitioned in 127 leaves by RCSR, which is much less than RSB's 255 leaves (238 CSRH, 16 COOH, 1 COO). For *bone010*, RCSR assembles 1316 CSR matrices; RSB assembles 170 CSRH, 2 CSR, and 5 COO. Index usage is reduced down from 4.6 to 2.5 bytes/nnz (Fig. 13). On **M2**, RSB assembles 1054 CSRH, 279 COOH, and 5 COO submatrices, while RCSR allocates 630 CSR leaves (index overhead shifting from 4.53 to 2.55 bytes/nonzero). Again, it seems the partitioning proceeded too deeply. Matrix *fcondp2* is partitioned in 31 leaves (19 CSRH, 1 CSR, 11 COOH) with RSB, and with RCSR in 255 leaves. Index overheads falls from 4.63 to 2.42 bytes/nnz. On the same matrix, on **M2** the improvement is even higher, this time. Here, RSB partitions in 257 leaves (182 CSRH, 75 COOH), while RCSR in 127 leaves only. Index overhead falls from 4.56 to 2.5 bytes/nonzero. So, in contrast to the preceding cases, matrix *fcondp2* benefits from increased subdivision, on **M2**. Matrix *ldoor* is partitioned in 157 leaves (122 CSRH, 5 CSR, 26 COOH, 4 COO, 3.14 bytes/nnz) by RSB, and 789 leaves by RCSR (5.47 bytes/nnz, Fig. 12). On **M2**, the performance gain is smaller than on **M1** (16%, rather than 19%). Partitioning of *ldoor*, here, produces 804 (471 CSRH, 329 COOH, 4 COO) submatrices, while RCSR produces 431 leaves. Also index overhead falls more gently: from 5.30 to 3.36 bytes/nnz.

We conclude by observing that there is a strong correlation between the index saving and performance gain: milder index savings on **M2** showed milder performance improvements, while bigger index savings on **M1** were accompanied by higher improvements.

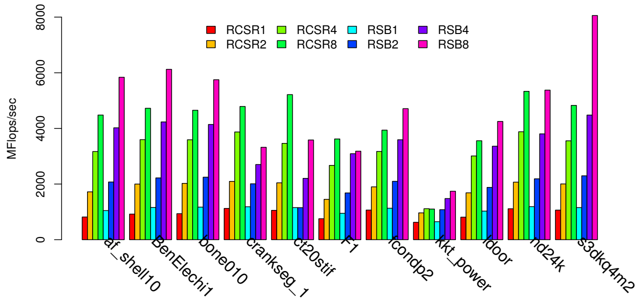


Fig. 10. Symmetric *SpMV* on **M1**.

C. Comparative analysis

Let us now look at the performance of all matrices as RCSR, RCSRH, and RSB, using 8 threads. For unsymmetric matrices, we also give performance results for the CSB prototype. Unfortunately, we had to skip matrix *cage15* (the one with the highest nonzeroes count), because CSB was unable to instantiate it (the CSB implementation needed more memory than the 24 GB available on **M1**).

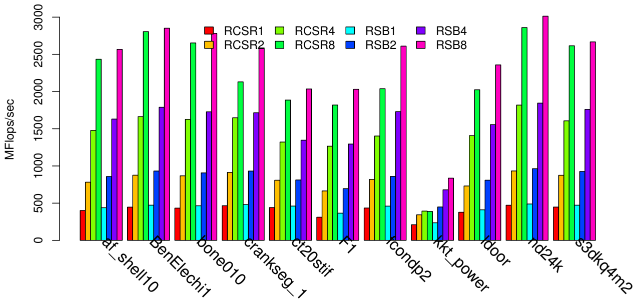


Fig. 11. Symmetric *SpMV* on **M2**.

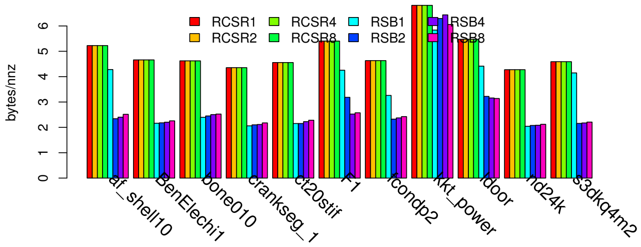


Fig. 12. Index storage requirement (in bytes) per nonzero on **M1** (symmetric matrices).

We observe that for **M2** (Fig. 15): matrices which favor RSB most (over CSB) are *c8_mat11_I*,*spal_004*,*wb-edu*; one matrix looses against RCSR (*cont11_l*); the majority of RSB cases is faster than RCSR (19 matrices out of 20). Summarizing, RSB performs faster than CSB (and is also the fastest among the four cases) in 7 cases out of 20. CSB is the fastest in 12 cases; in one case it is faster than RSB, but not the fastest one.

On **M1** (Fig. 14) we observe that: RSB is much faster than CSB on *wb-edu* and *venkat01*; 6 matrices seem to perform very similarly in both CSB or RSB; the remaining ones perform better in one of the two formats. Some matrices loose performance in RSB, over RCSR: matrices *av41092*,*c8_mat11_I*,*cont11_l*; (slightly) *diego-smtxMM-573x230k*,*sme3Dc*; other matrices favor RSB over RCSR: about 15, out of 20.

For space reasons, we omit figures showing comparative performance for symmetric matrices on RCSR, RCSRH, RSB formats, but include some general comments.

On **M2**, we notice RSB as the fastest format 5 times out of 12; on **M1**, 4 times. Here, RCSRH is the fastest in 7 cases; in all cases, very near to RSB. On **M1**, we see a similar situation, but notice a performance degradation in some additional cases: they are due to the poor partitioning problem discussed in Section V-B. In no case RCSR was the fastest format for

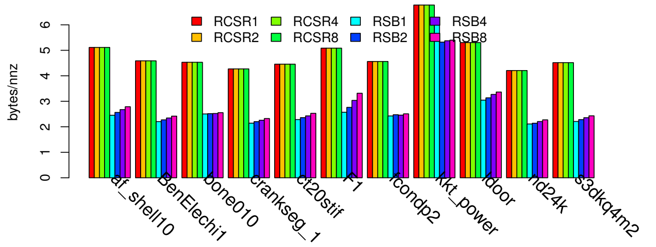


Fig. 13. Index storage requirement (in bytes) per nonzero on **M2** (symmetric matrices).

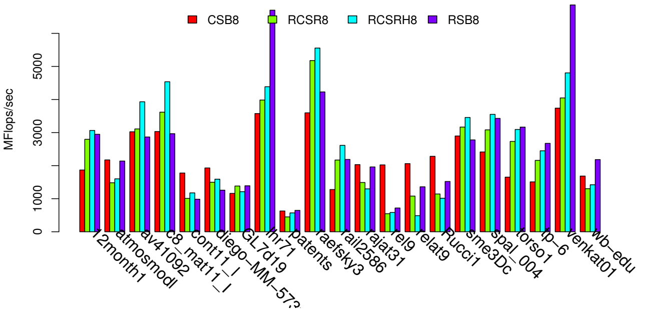


Fig. 14. Results for 8 cores on **M1**, comparing CSB, RCSR, RCSRH, and RSB (unsymmetric matrices).

symmetric matrices (exception made for the poorly scaling three matrices) on **M1**: (*crankseg_1*, *ct20stif*, *F1*).

VI. Conclusions and Future Work

In this paper, we have shown a possible improvement of our BLAS-oriented recursive storage for sparse matrices. We have found that, by using index compression and format diversification techniques, we can improve the floating point performance of *SpMV*. We have also found that, for unsymmetric matrices, the performance of our modified format (RSB) is comparable to that of a scalable sparse matrix format (CSB: currently for unsymmetric only). During comparison with RCSR and CSB, we noticed some particular cases that expose *weak points* of both RSB and RCSR; consequently allowing us to identify room for further improvement: (i) To redefine our format in order to obtain some estimate on the parallelism expected from a given partitioning (in Section V-B, we noticed that, despite the apparently adequate partitioning, some instances of matrices (e.g.: smaller symmetric) did not scale on 8-threaded *SpMV*). (ii) To modify the *SpMV* algorithm to be more parallel, by working around the need for row locking (e.g.: by using temporary vectors, as CSB does [2, Sec.4], although this may be challenging in our case). (iii) While our primary interest is focused on bigger matrices, tuning the partitioning algorithm for small matrices could prove useful to ensure parallelism in these cases, too. (iv) Properly subdividing matrices which are big, but with an extremely low nonzeroes/row ratio would be challenging (and fruitful), as well.

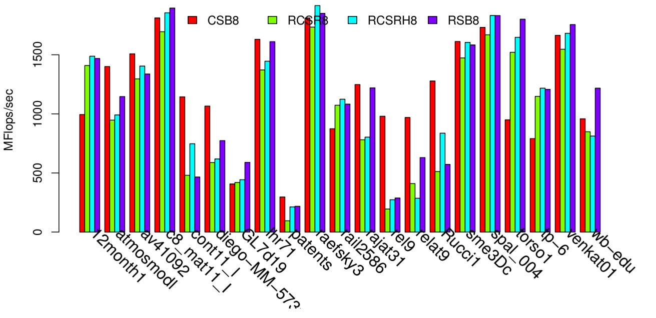


Fig. 15. Results for 8 cores on **M2**, comparing CSB, RCSR, RCSRH, and RSB (unsymmetric matrices).

Some ideas we have introduced should be developed further. For instance, a more aggressive form of tuning could diversify index types at the *leaf level* and continue using traditional CSR or COO layouts, if profitable. Probably future architectures (with much higher number of cores, and even higher risks for stall due to higher memory latencies and longer instruction pipelines) would render such approaches advantageous.

In summary, we can state that our work illustrates that combinations of hierarchical indexing and index compression techniques can be useful to achieve high efficiency of computing on sparse matrices (on general purpose hardware). In this light, we see the RSB format as a candidate format for a complete multicore sparse BLAS implementation (that is, support for symmetric storage, solve operations, parallel transposed *SpMV*, etc.).

Finally, we would like to thank Jamie Wilcox and Victor Gamayunov from Intel EMEA Technical Marketing HPC Lab for their technical support during experiments.

References

[1] K. Kourtis, G. Goumas, and N. Koziris, “Improving the performance of multithreaded sparse matrix-vector multiplication using index and value compression,” *Computing Frontiers*, pp. 87–96, 2008.

[2] A. Buluc, J. T. Fineman, M. Frigo, J. R. Gilbert, and C. E. Leiserson, “Parallel sparse matrix-vector and matrix-transpose-vector multiplication using compressed sparse blocks,” in *SPAA*, F. M. auf der Heide and M. A. Bender, Eds. ACM, 2009, pp. 233–244.

[3] S. Williams, L. Oliker, R. Vuduc, J. Shalf, K. Yelick, and J. Demmel, “Optimization of sparse matrix-vector multiplication on emerging multicore platforms,” in *Proceedings of the 2007 ACM/IEEE Conference on Supercomputing*. ACM New York, NY, USA, 2007.

[4] M. Martone, S. Filippone, S. Tucci, M. Paprzycki, and M. Ganzha, “Utilizing recursive storage in sparse matrix-vector multiplication - preliminary considerations,” in *CATA*, T. Philips, Ed. ISCA, 2010, pp. 300–305.

[5] M. Martone, S. Filippone, M. Paprzycki, and S. Tucci, “On blas operations with recursively stored sparse matrices,” in *Proceedings of the International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, September 2010.

[6] ——, “On the usage of 16 bit indices in recursively stored sparse matrices,” in *Proceedings of the International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, September 2010.

[7] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. Philadelphia, PA: SIAM, 1994.

[8] M. Martone, S. Filippone, M. Paprzycki, and S. Tucci, "About the assembly of recursive sparse matrices," in *Proceedings of the International Multiconference on Computer Science and Information Technology*, October 2010.

[9] "Standard for information technology— portable operating system interface (posix) (ieee std 1003.1)," 2008.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 337–342

ISBN 978-83-60810-27-9
ISSN 1896-7094

Higher order FEM numerical integration on GPUs with OpenCL

[1]Przemysław Płaszewski, [12]Krzysztof Banaś, [2]Paweł Macioł
[1]AGH University of Science and Technology,
Department of Applied Computer Science and Modelling,
al. Mickiewicza 30, 30-059 Kraków, Poland
[2]Cracow University of Technology,
Insitute of Computer Modelling,
ul. Warszawska 24, 31-155 Kraków, Poland

***Abstract*—Paper presents results obtained when porting FEM 2D linear elastostatic local stiffness matrix calculations to Tesla architecture with OpenCL framework. Comparison with native NVIDIA CUDA implementations has been provided.**

I. Motivation

IN FINITE element simulations usually two computation stages most significantly impact performance of the whole process:

- obtaining global stiffness matrix
- solving system of linear equations

In case of non-linear, higher order element geometry, higher order approximations and p and hp-adaptations, process of obtaining global stiffness matrix requires computationally intensive separate calculations of local (element) stiffness matrices. For some problems it can be most time consuming stage of FEM calculations.

Aim of our work is parallelizing this stage utilizing modern graphics processor units (GPUs) and OpenCL platform.

The paper is organized as follows. The first two sections are devoted to the definition of the problem of FEM numerical integration. In the third section we summarize the problem in terms of requirements for numerical algorithms. Then we show how to design parallel algorithms that solve the formulated computational problem on modern GPUs. The results of experiments close the paper.

II. Finite element numerical integration

The standard procedure in FEM computations consist in obtaining weak formulation of a problem, discretizing problem domain Ω into finite elements and utilizing appropriate basis functions—constructed from element shape functions—to create a system of linear equations, with the global stiffness matrix as the system matrix, that is then solved to provide approximate solution.

Generation of global stiffness matrix is usually performed by calculating integrals over finite elements, then assembling obtained that way local matrices into global one. Since integrals evaluation over multiple different element geometries (possibly curved) would pose a problem, elements are mapped to a reference element with simple geometry and integrals are calculated over its area. Every element is processed independent of the others thus many can be calculated in parallel.

III. Model problem

We chose 2D linear elastostatics problem. As a model finite element we use quadrilateral with curved, second order, geometry. Solution is approximated with hierarchical shape functions up to order $p = 7$, constructed from tensor products of 1D Lobato hierarchical functions [2]. Reduced space was used with number of shape functions equal to

$$n = 4 + 4(p-1)_+ + (p-2)(p-3)_+/2$$

where q_+ denotes $max(q, 0)$. Local stiffness matrix dimension is equal to $2n$, where n is number of shape functions for a particular order p. Matrix entries are results of calculating the integral [1]

$$k^{(e)}_{IJ} = \iint_{\Omega_{ref}} (([D^*]\{\varphi_I\})^T [E][D^*]\{\varphi_J\}) \mid J \mid d\xi d\eta \quad (1)$$
$$I, J = 1, 2, ..., 2n$$

where $k^{(e)}_{IJ}$ is (I, J) matrix entry, Ω_{ref} is $[-1,1] \times [-1,1]$ reference quadrilateral, $[E]$ is 3×3 material matrix, $\mid J \mid$ is the determinant of the Jacobian matrix of geometry transformation, $\{\varphi_I\}$ and $\{\varphi_J\}$ are columns of $2n \times 2$ matrix of shape functions. $[D^*]$ is the matrix differential operator

$$[D^*] = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix}$$

where

$$\begin{aligned} \frac{\partial}{\partial x} &= J^*_{11}\frac{\partial}{\partial \xi} + J^*_{12}\frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial y} &= J^*_{21}\frac{\partial}{\partial \xi} + J^*_{22}\frac{\partial}{\partial \eta} \end{aligned}$$

and $[J^*]$ is inverse Jacobian matrix.

Integration over reference quadrilateral is done using Gauss quadratures where the double integral is replaced by double sum

$$\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta) d\xi d\eta \approx \sum_{i=1}^{n_g}\sum_{j=1}^{n_g} f(\xi_i,\eta_j) W_i W_j$$

978-83-60810-27-9/09/$25.00 © 2010 IEEE

In case of bilinear mapping, integration is accurate when number of gauss points n_g in both dimensions is not less than $(p+1)^2$ where p is order of approximation (i.e. highest polynomial order in shape functions set).

IV. Computation overview

Evaluating the double integral boils down to calculating double sum, which can be achieved with single loop over gauss points in two dimensions. In every iteration, for every entry in a matrix, the result of calculating (1) is multiplied by Gauss weights and added to the (first zeroed) element stiffness matrix matrix.

The integrated formula differs for different matrix entries only in shape functions used—for every entry a distinct pair of shape functions is evaluated (taking into account the symmetry of a matrix).

Jacobian determinant and the inverse of Jacobian matrix (which appears in $[D^*]$ operator) are the same for all matrix entries thus can be calculated once per iteration.

Assuming constant Young modulus and Poisson ratio over a single finite element, material matrix $[E]$ can be calculated once per element.

Calculation of single local stiffness matrix requires:

- single calculation of material matrix $[E]$
- n_g calculations of Jacobian determinant and inverse of the Jacobian matrix, where n_g is number of gauss points
- $n_g\frac{n(n+1)}{2}$ calculations of entries (1), where n is matrix dimension (symmetric matrix case)

V. Parallel numerical integration in OpenCL

Finite element local stiffness matrices are calculated independent of each other therefore can be carried out concurrently. The process of calculating a single local stiffness matrix can also be parallelized by simultaneous computations of its entries. Therefore we can identify two possible parallelization levels:

- processing finite element matrices
- processing matrix entries

This naturally fits to OpenCL parallelization model [4] of:

- work-groups executed concurrently, independent of each other
- work-items executed concurrently, cooperating inside each work-group

Our implementation takes advantage of above similarities and divides computations as follows:

- each work-group is responsible for processing single finite element
- work-item is responsible for calculating one or more matrix entries

Despite the fact that the OpenCL platform aims to provide unified execution environment for broad range of hardware solutions like multi-core CPUs, graphics processors and accelerators like IBM Cell, programmer still needs to realize differences between them and optimize his code accordingly. Our implementation targets Nvidia Tesla architecture.

Calculations on device, in our case GPU, are directed by host CPU. Its role is to upload elements data into device global memory, execute kernel and download output matrices when kernel finishes. If not all elements can be processed in single kernel launch—for example due to limited amount of global device memory—whole computations can be divided across multiple sequences of:

- uploading data to device memory
- executing kernel
- downloading data

Kernel launch is non-blocking therefore host can process remaining batches of elements while waiting for GPU to finish, thus increasing overall performance.

Parallel calculations overview is presented on Fig.1.

Each work-group reads the data representing particular finite element from global device memory. Then work-items in cooperation evaluate matrix entries and store them in local memory—shared by all items in a work-group. When done, matrix is copied from local to global device memory accessible by host.

Data that are common for all finite elements i.e. gauss points and weights is calculated once on CPU, stored in constant memory and available for all work-groups. Constant memory is cached on a GPU and well serves the purpose of supplying the multiprocessors with the data that are the same for all elements of a given order p.

During the execution work-groups are queued and successively consumed by device computation units. Each computation unit—in our case a GPU multiprocessor—can execute more than one work-group at a time, however splitting single work-group across many computation units is not allowed.

The number of simultaneously processed work-groups on single computation unit (resident work-groups) depends on factors like consumption of registers, consumption of multiprocessor local memory and size of a work-group. GPU scheduling hardware will send as much work-groups as possible for concurrent execution on single multiprocessor until there are no resources left. In situation when number of consumed registers, shared memory or number of work-items exceeds the half of maximum available for particular hardware, only one work-group at a time will be resident on single multiprocessor.

Optimal use of multiprocessor resources is important to achieve high occupancy i.e. high number of work-items from one or more work-groups executed concurrently on multiprocessor. In our case resource consumption depends on approximation order—the higher is the order p the bigger is the local stiffness matrix and more local memory is needed by work-group during the computations.

Resource limits for Nvidia GPU devices of compute capability 1.3 (all cards we tested) are:

- Multiprocessor registers: 16384
- Local memory per multiprocessor: 16KiB
- Maximum number of work-items in work-group: 512
- Maximum number of resident work-items on multiprocessor: 1024

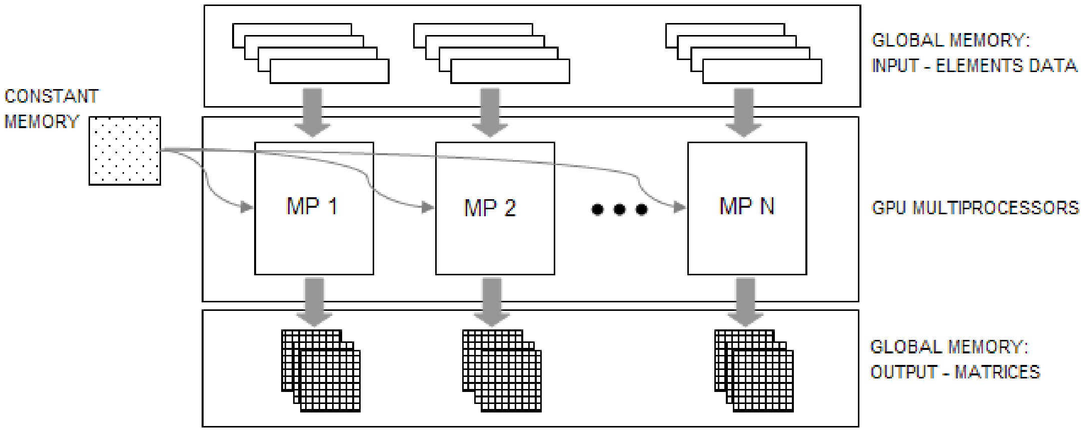


Fig. 1. Parallel calculations on GPU.

Available registers are uniformly distributed among resident work-items executed on multiprocessor and local memory is divided across resident work-groups.

In order to generate local stiffness matrix for single finite element, following information needs to be supplied (23 float values in total):

- geometry of quadrilateral (coordinates of its vertices and second order geometry mapping nodes)
- edge orientations
- material information (Young modulus, Poisson ratio)

We have padded each element data to 32 and placed all elements data in contiguous region of global memory in order to achieve specific coalesced access pattern [3] by every work-group. For the same reason we also used padding for output matrices.

Every work-group (i.e. all its work-items executing in parallel) performs single coalesced read from input global memory region in the beginning of computations to get element data and coalesced writes at the end of execution to store generated matrix.

Total size of input data is

$$32 * sizeof(float) * numelems$$

Total size of output matrices is

$$matpaddedsize(p) * sizeof(float) * numelems$$

where $matpaddedsize(p)$ is matrix size with padding and depends on order of approximation p. Due to symmetry we are only calculating and storing upper-half matrices, thus reducing both local and global memory usage. Sizes of matrices for different approximation orders are presented in table I.

Depending on the order p different kernel is being executed on the device. Flow of computation for every kernel is similar. The differences come from different number of gauss points, different set of hierarchical shape functions and different sizes of matrices and manifest themselves in:

- amount of local memory needed to compute stiffness matrix
- number of iterations over gauss points
- amount of local memory needed to compute derivatives of shape functions in every Gauss point
- number of matrix entries per work-item
- number of work-item engaged in calculations of stiffness matrix (which is less-or-equal to the work-group size)

For every kernel we configured computations in such way not to exceed the available computation unit resources (registers and local memory), maximize global memory throughput with coalesced reads and writes, maximize computation unit occupancy and minimize divergent branches inside groups of 32 consecutive work-items called warps.

Execution configuration included proper sizing of work-groups and defining number of matrix entries calculated by single work-item. We experimented with broad range of configurations for every kernel in order to chose optimal ones—those are presented in table I.

In case of higher p orders, limited multiprocessor resources—especially local memory size of only 16KiB on Tesla architecture—prevented simultaneous execution of more than one work-group thus reducing multiprocessor occupancy.

We implemented kernels for orders of approximation up to 7—for higher orders half-matrix size is bigger than amount of local memory available on Tesla architecture and different approach to computations would be needed.

VI. Kernel computations

Main steps of kernel execution are:

1) Zero matrix in local memory (all work-items)
2) Load element data from global memory (first warp)
3) Calculate 3×3 material matrix $[E]$ (single work-item)
4) Loop over gauss points:
   a) Calculate values of shape functions derivatives (single work-item)

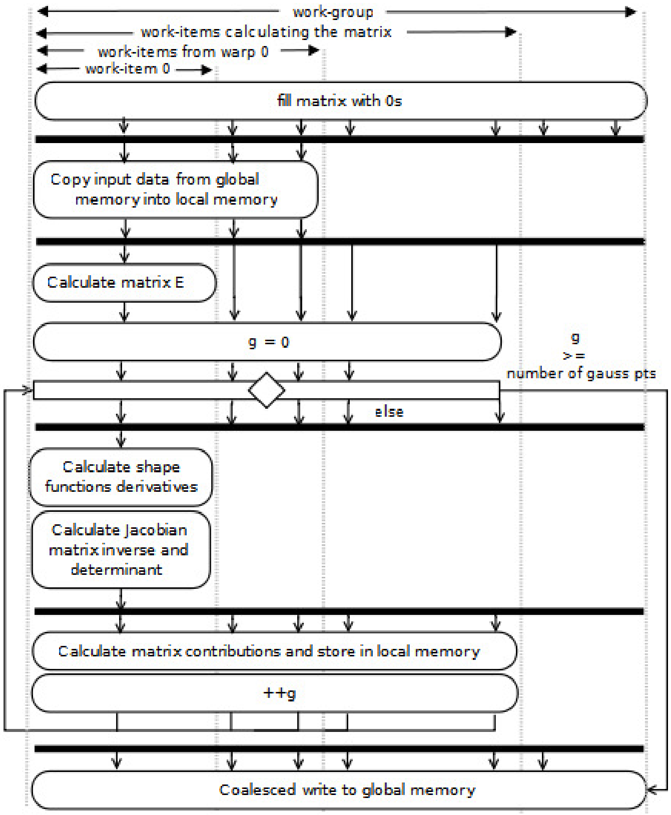

work-group
work-items calculating the matrix
work-items from warp 0
work-item 0
fill matrix with 0s
Copy input data from global memory into local memory
Calculate matrix E
g = 0
g >= number of gauss pts
else
Calculate shape functions derivatives
Calculate Jacobian matrix inverse and determinant
Calculate matrix contributions and store in local memory
++g
Coalesced write to global memory

Fig. 2. Work-group execution.

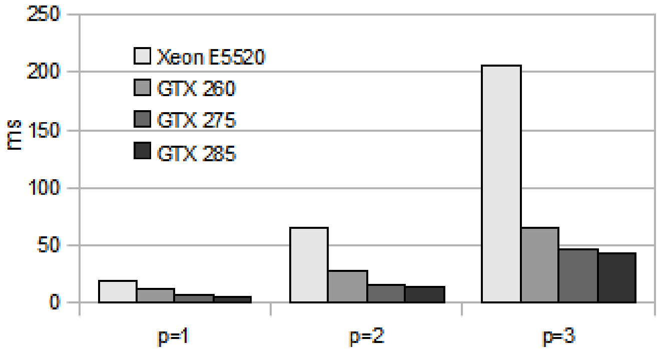


Fig. 4. Execution times for 10000 elements.

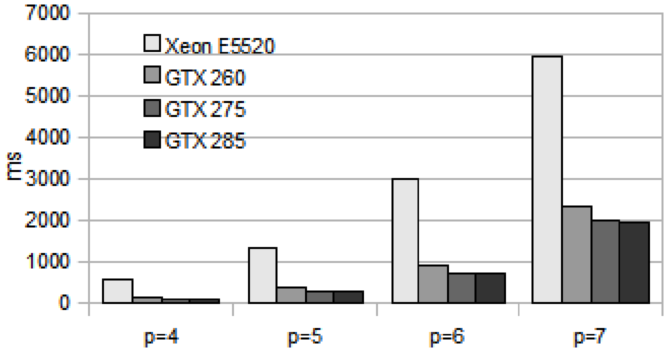


Fig. 5. Execution times for 10000 elements (continued).

b) Calculate Jacobian determinant and the inverse of Jacobian matrix (single work-item)
c) Calculate stiffness matrix contributions (in parallel—number of work items depends on kernel—see table I)

5) Upload matrix to global memory (all work-items)

Fig. 2 presents the calculation flow for a work-group. Black horizontal bars indicate OpenCL $CLK\_LOCAL\_MEM\_FENCE$ synchronization barriers. Those are needed to ensure that all calculations from previous steps are done before utilizing their results in steps that follows. We experimented with disabling synchronizations in order to observe their influence on overall algorithm performance—execution times dropped by not more than 5%.

Material matrix $[E]$, Jacobian determinant, inverse of Jacobian matrix and shape function calculations never took more than 10% of total execution time. Attempts to spread shape functions calculations across more work-items and perform them in parallel resulted in higher register consumption per work-item thus reducing occupancy and decreasing performance for most kernels.

In order to effectively utilize limited local memory, local stiffness half-matrix is stored in linear contiguous fashion row by row, every consecutive row being one shorter than previous. Every work-item in a work-group is identified by its local id obtained with get_local_id device function. Assignment of matrix entries to distinct work-items relies on mapping formula involving square root operation. Fig. 3 presents work-items assignments for kernel $p = 3$. For example work-item with id 44 is responsible for calculating five matrix entries at positions (22,11), (23,11), (12,12), (13,12) and (14,12). Work-items with numbers from 60 to 63 are left idle during matrix calculations, but do participate in writing padded matrix to global memory when done.

Matrices in every kernel are padded to the nearest multiply of 32 and their size is multiply of work-group size to achieve coalesced writes.

VII. Results

Figures 4 and 5 show results we obtained while calculating 10^4 matrices on 3 NVDIA GTX series graphics cards. For comparison we performed tests of cache optimized sequential code run on single Nehalem core of Intel Xeon E5520 processor.

Best performing kernel ($p = 4$) achieved 5.3 speedup over sequential implementation. Lowest speedups of 3.3 and 3 were observed with kernels for $p = 1$ and $p = 7$ respectively.

For GPU implementation timings include transfers of data between host and device through PCI-Express bus ($clEnqueuReadBuffer$ and $clEnqueueWriteBuffer$ host functions). Data transfer is amortized—especially for lower order kernels—only when number of processed elements is high enough—for example for $p = 1$ and 1000 elements transfers take 25% of total execution time while for $p = 5$ only 4%.

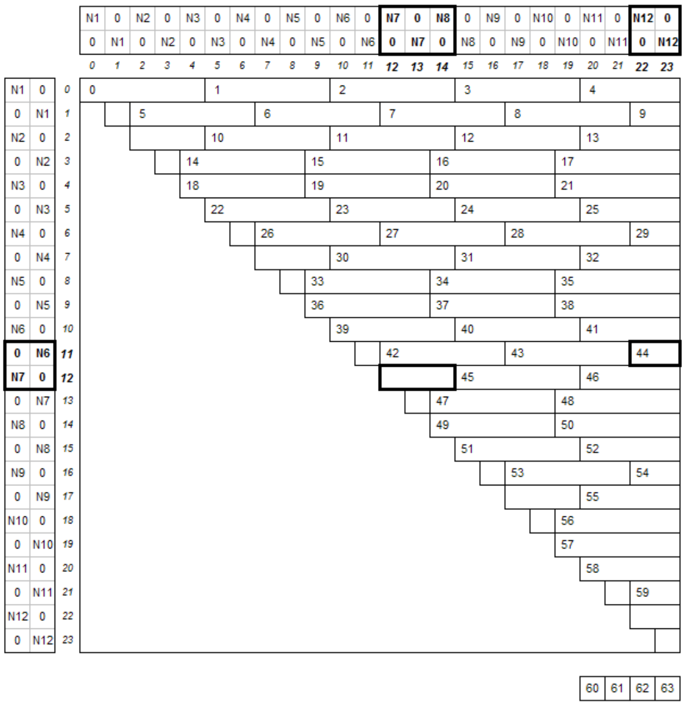


Fig. 3. Work-items to matrix entries assignment for $p = 3$.

TABLE I
KERNEL PARAMETERS

	p=1	p=2	p=3	p=4	p=5	p=6	p=7
Work-group size	64	192	64	128	192	384	448
Work-items involved in matrix generation	36	136	60	119	181	366	418
Matrix dimension	8	16	24	34	46	60	76
Padded matrix entries number	64	192	320	640	1152	1920	3136
Register consumption per work-item	12	13	18	19	20	19	18
Local memory consumption per work-group (bytes)	568	1208	1848	3288	5528	8824	13944

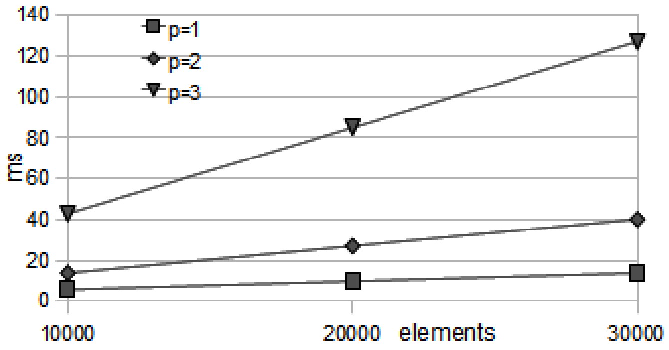


Fig. 6. Execution times for different elements number on GTX 285.

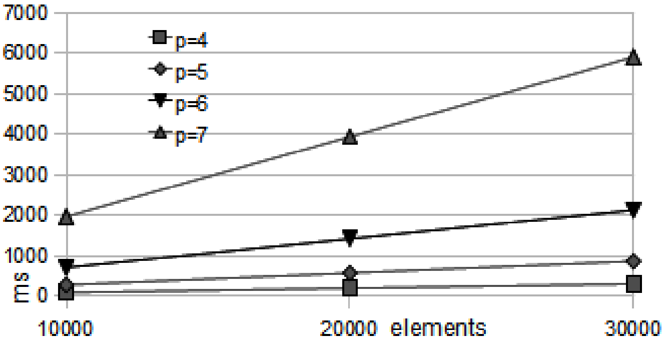


Fig. 7. Execution times for different elements number on GTX 285 (continued).

Figures 6 and 7 present execution times with varying number of elements processed. As expected execution time scales in linear fashion with elements number—as every element is represented on device by work-group and work-groups are queued for execution on multiprocessors.

Of 3 cards tested 2 (GTX 275 and GTX 285) have the same number of multiprocessor (32) and differ in global memory bus bandwidth—448 bit and 512 bit respectively. Performance difference between those two cards is minimal as compared to

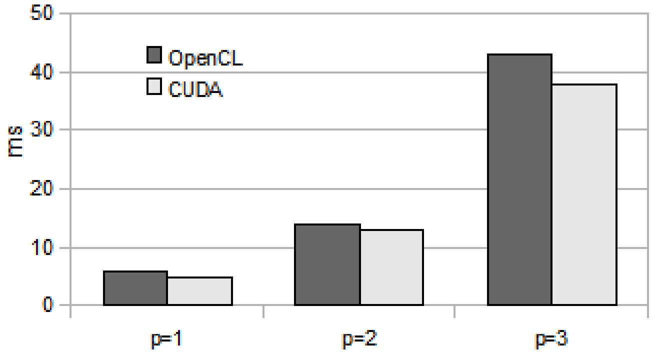


Fig. 8. Performance comparision of OpenCL and CUDA with 10000 elements on GTX 285.

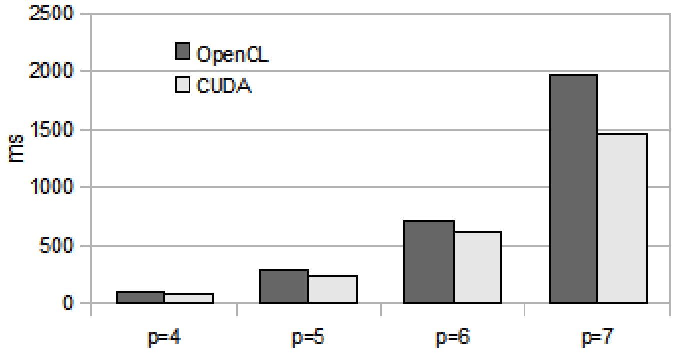


Fig. 9. Performance comparision of OpenCL and CUDA with 10000 elements on GTX 285 (continued).

the difference between cards with different number of multiprocessors (GTX 260 with 27 and GTX 275 and GTX 285 both having 30). Our implementation isn't memory constrained since all operations except single input and output coalesced writes are performed on local multiprocessor memory and the biggest size of transferred batch is only 13KiB for matrix of $p = 7$.

One can expect linear performance scaling with increasing number of GPU multiprocessors. Number of simultaneously processed work-groups on all multiprocessors is given as:

$$R_{wg} \times M$$

where R_{wg} is number of resident work-groups per multiprocessor and M number of multiprocessors. Let t_1 be the time that takes to process $R_{wg} \times M_1$ work-groups on GPU G_1 having M_1 multiprocessors. Similarly t_2, M_2 for GPU G_2. Assuming identical performance (i.e same clock rate and architecture) of multiprocessors in both GPUs $t_1 = t_2 = t$. Total time of processing N work-groups on GPU G_1 is:

$$T(G_1, N) = \frac{N}{M_1 \times R_{wg}} t$$

and on G_2:

$$T(G_2, N) = \frac{N}{M_2 \times R_{wg}} t$$

Speedup calculated as a ratio of those two times:

$$\frac{T(G_1, N)}{T(G_2, N)} = \frac{\frac{N}{M_1 \times R_{wg}} t}{\frac{N}{M_2 \times R_{wg}} t} = \frac{M_2}{M_1}$$

indicates linear scaling with number of multiprocessors.

According to the above statement observed speedup should be not less than 11% between GTX 260 and GTX 275 processors. Our benchmarks demonstrate speedup in fact being higher—especially for lower orders of approximation—due to increased GPU clock-rate and better system components (i.e. CPU, motherboard) on machine equipped with GTX 275 card.

We compared OpenCL performance with native NVIDIA CUDA 3.0 implementations (see Figures 8 and 9) of our kernels. For $p = 1$ and $p = 2$ kernels were compiled with same register consumptions on both platforms and results are close. Initial compilation of other kernels resulted in increased register usage of OpenCL kernels as compared to CUDA.

Since higher register consumption resulted in much lower occupancy and significantly decreased performance we forced OpenCL compiler to use the same number of registers as in the CUDA build. Consumption decreased, but unfortunately OpenCL compiler was unable to achieve this without spills to slow private memory thus slightly decreasing performance (however not that much as with increased register consumption).

Overall performance of OpenCL in our case is comparable to CUDA, except when compiler is not able to optimize its output the way more mature CUDA tools do.

References

[1] Barna Szabo and Ivo Babuska, *Finite Element Analysis,* Wiley-Interscience; 1991.

[2] Pavel Solin and Karel Segeth and Ivo Dolezel, *Higher-Order Finite Element Methods,* Chapman & Hall/CRC; 2003.

[3] Mark Harris, "Optimizing CUDA," *in Supercomputing conference,* Reno, NV, 2007.

[4] Khronos OpenCL Working Group, *The OpenCL Specification 1.0,* 2009.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 343–348

ISBN 978-83-60810-27-9
ISSN 1896-7094

Parallelization of SVD of a Matrix-Systolic Approach

Halil Snopce
South East Europian University,
Ilindenska bb., 1200 Tetovo, Republic of Macedonia
Email: h.snopce@seeu.edu.mk

Ilir Spahiu
Pedagogical Faculty “Kliment Ohridski”, bul. ‘Krste Misirkov’ bb. 1000 Skopje, Republic of Macedonia
Email: i.spahiu@seeu.edu.mk

Abstract—**In this paper we investigate the parallelization of Hestenes-Jacobi method for computing the SVD of an mxn matrix using systolic arrays. In the case of real matrix an array of R^2 processors is proposed, such that each row contains n columns. In order to extend this idea we have presented three transformations which are used for transforming the complex into the real matrix. After the additional computations, we show how the same array may be used for the SVD of a complex matrix.**

I. Introduction

THE Singular Value Decomposition (SVD) is a matrix decomposition of a great importance in different engineering applications. It is particularly useful in the areas of signal and image processing, robotics, pattern recognition etc. This decomposition can be used for determining the rank of a matrix in numerically reliable manner [2,7]. Different approaches are known in the literature for the parallel computation of the SVD of a matrix where the elements are meanly real numbers. In [5], an expandable array for parallel computing of SVD of large matrices is proposed. In [11], the use of CORDIC method for SVD is demonstrated. In [14] is given a fast Jacobi-like algorithm for the parallel solution of the SVD not focusing in CORDIC form, but by applying approximate rotations. In some applications one needs to use complex matrices as well (like in beam-forming algorithms in the signal processing [8]). In this paper we propose an approach which offers parallelization using systolic arrays. First we give the method of constructing the corresponding systolic array where the matrix is real, and then we analyze how the corresponding systolic array for the SVD can be designed if the elements of the matrix are complex numbers. This differs from the case of real matrix, because of the fact that complex arithmetic and matrix transformations in this arithmetic require greater number of computational steps. In [9] it is proposed the SVD of a matrix with complex elements where the matrix is with special structure. On the other hand in [10] this idea is extended for an arbitrary complex 2x2 matrix. In [12] is presented a systolic design concept for iterative algorithms when the VLSI design keeps evolving into nanoscale. In [15] is designed the parallelism of the so called Hestenes-Jacobi method for the SVD using the ring array. A systolic array for the computation of the SVD is presented in [5]. The array uses $(n/2)^2$ processors and is capable of processing the SVD of a square matrix. The time complexity is $O(n\log n)$, which is to be compared with the best serial algorithms, which have a $O(n^3)$ time complexity. In [13,16] the efficiency of this array is improved. In this paper it is given a model of systolic array for the SVD of an mxn matrix which can be found with the Hestenes-Jacobi method. Then, in the case of a complex matrix, there are given some transformations which transform the complex into the real matrix. Using these transformations the same systolic array can be used for the SVD of complex matrix.

II. The SVD of A Matrix

The SVD of an $m\times n$ matrix A is given by:

$$A = U\Sigma V^T =$$

$$\begin{bmatrix} u_1 & u_2 & \dots & u_{m-1} & u_m \end{bmatrix}_{m\times m} \begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \dots & & & \\ 0 & 0 & \sigma_r & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}_{m\times n} \begin{bmatrix} v_1^T \\ v_2^T \\ : \\ v_{n-1}^T \\ v_n^T \end{bmatrix}_{n\times n} \quad (1)$$

where U and V are orthogonal $m\times m$ and $n\times n$ matrices respectively, (i.e., $U^T U=I_m$ and $VV^T=I_n$) and Σ is a diagonal $m\times n$ matrix such that $\Sigma=diag(\sigma_1,\sigma_2,\dots)$; $\sigma_1\geq\sigma_2\geq\dots\geq\sigma_r\geq\sigma_{r+1}=\dots=\sigma_k=0$ where $r=rankA$. In the case $m=n,\ \Sigma$ is square diagonal matrix of order n . In the definition given above the σ_i -s are the singular values of A .

The SVD decomposition is often based on diagonalizing rotations which are orthogonal transformations which preserve Eigenvalues and Eigenvectors as well as singular values and singular vectors. The sequence of rotations A_k , such that $\lim_{k\to\infty} A_k=\Sigma$, is applied during the process.

III. Jacobi Rotations

The classical Jacobi rotation [1] has been used by Jacobi in 19th century as a tool for solving the least square problem. This rotation is also called Given's rotation. This method uses a sequence of plane rotations to diagonalize a symmetric $n \times n$ real matrix A. We denote the Jacobi rotation of an angle θ in the (i, j) plane by $J(i, j, \theta)$. This is a square matrix equal to the identity matrix except the four additional elements in the intersection of i-th and j-th rows and columns:

$$\begin{array}{r|ccccccc|}
 & & & i & & j & & \\
 & 1 & & & & & & \\
 & & \cdots & & & & & \\
i & & & c & \cdots & s & & \\
J(i,j,\theta)= & & & & \cdots & & & \\
j & & & -s & \cdots & c & & \\
 & & & & & & \cdots & \\
 & & & & & & & 1
\end{array}$$

where $c = \cos\theta$ and $s = \sin\theta$. It's not difficult to verify that this is an orthogonal transformation because of the fact that $J(i,j,\theta)^T J(i,j,\theta) = I$ for each θ. The rotation angle θ is chosen such that to annihilate the $n(n-1)$ off-diagonal elements of the given matrix. Each Jacobi transformation brings the matrix A closer to the diagonal form [2].

In the two sided Jacobi method for the SVD decomposition of a nonsymmetric matrix, the annihilation of the off-diagonal elements is done using a pair of rotations [3], such that:

$$\begin{bmatrix} a_{ii}^{k+1} & 0 \\ 0 & a_{jj}^{k+1} \end{bmatrix} = \begin{bmatrix} c_1 & -s_1 \\ s_1 & c_1 \end{bmatrix}^T \cdot \begin{bmatrix} a_{ii}^k & a_{ij}^k \\ a_{ji}^k & a_{jj}^k \end{bmatrix} \cdot \begin{bmatrix} c_2 & s_2 \\ -s_2 & c_2 \end{bmatrix} \qquad (2)$$

If θ_1 and θ_2 are the angles generating the pairs (s_1, c_1) and (s_2, c_2) then the solution given in [3] is:

$$tg(\theta_1 + \theta_2) = \frac{a_{ji}^k + a_{ij}^k}{a_{jj}^k - a_{ii}^k}, \quad tg(\theta_2 - \theta_1) = \frac{a_{ji}^k - a_{ij}^k}{a_{jj}^k + a_{ii}^k} \qquad (3)$$

IV. HESTENES-JACOBI METHOD

Computationally better results offers the so called Hestenes-Jacobi method [4]. This method works with a smaller unit of computation (only with the rows of the matrix) compared with the first explained Jacobi method (which modifies both the rows and columns). Since Hestenes-Jacobi transformations are orthogonal transformations which leave singular values unchanged, it is not important how many such transformations are applied. For a given $m \times n$ matrix A the Hestenes-Jacobi method produces an orthogonal matrix U as a product of plane rotations and a matrix N which rows are orthogonal.

$$UA = N = [\eta_1 \quad \eta_2 \quad \ldots \quad \eta_n];\ \eta_i^T \eta_j = 0 \text{ for } i \neq j \qquad (4)$$

The matrix N can be normalized by computing the square of the row norms $\sigma_i = \eta_i^T \eta_i$ and writing: $N = \Sigma\Sigma^{-1}N = \Sigma V$ (in this case the computation of V is done by dividing the element η_i of N by $\sigma_i = \eta_i^T \eta_i$. Because of $\eta_i^T \eta_j = 0$ for $i \neq j$, the matrix Σ is a diagonal matrix). The matrix V is orthogonal and the σ_i are the non-negative squares of the singular values. Because U is orthogonal we have:

$$UA = \Sigma V \Rightarrow A = U^T \Sigma V \qquad (5)$$

This equation is the same as the equation of the SVD of the matrix A. Multiplying both sides of (5) by A^T we have:

$$A^T A = (U^T \Sigma V)^T U^T \Sigma V = V^T \Sigma^T U U^T \Sigma V = V^T \Sigma^T \Sigma V$$

This means that Hestenes computation mathematically is equivalent to the Jacobi method applied to $A^T A$ where singular values are equal to the squares of σ_i.

The orthogonal matrix U may be taken as a plane rotation matrix $J(i, j, \theta)$. If $A_1 = A$, then $A_{k+1} = A_k J$. The iterations result in the matrix N defined by (4). Working with a submatrix of order 2×2 we have:

$$\begin{aligned} \begin{bmatrix} a_i^{k+1} & a_j^{k+1} \end{bmatrix} &= \begin{bmatrix} a_i^k & a_j^k \end{bmatrix} \cdot \begin{bmatrix} c & -s \\ s & c \end{bmatrix} = \\ &= \begin{bmatrix} a_i^k \cdot c + a_j^k \cdot s & -a_i^k \cdot s + a_j^k \cdot c \end{bmatrix} \end{aligned} \qquad (6)$$

From the condition of orthogonality we have:

$$(a_i^k \cdot c + a_j^k \cdot s)^T \cdot (-a_i^k \cdot s + a_j^k \cdot c) = 0 \qquad (7)$$

From these relations the iteration formulae for updating the value of an inner product is given by the relation bellow:

$$\begin{aligned} &(a_i^{k+1})^T \cdot a_j^{k+1} = (c^2 - s^2)(a_i^k)^T \cdot a_j^k + cs\left[(a_i^k)^T a_i^k - (a_j^k)^T a_j^k\right] \\ &\Leftrightarrow (a_i^{k+1})^T \cdot a_j^{k+1} = (c^2 - s^2)(a_i^k)^T \cdot a_j^k + cs\left[\left\|a_i^k\right\|^2 - \left\|a_j^k\right\|^2\right] \end{aligned} \qquad (8)$$

Writing $g_{ij}=\left(a_i\right)^T \cdot a_j$, the following equality will be fulfilled:

$$\left(c^2-s^2\right)g_{ij}+cs\left[\|a_i^k\|^2-\|a_j^k\|^2\right]=0 \qquad (9)$$

From the equality (9) we can obtain the relation:

$$\lambda=ctg\,2\theta=\frac{\cos 2\theta}{\sin 2\theta}=\frac{\|a_j^k\|^2-\|a_i^k\|^2}{2g_{ij}} \qquad (10)$$

Putting $t=tg\,\theta$, to the relation $tg\,2\theta=\frac{2tg\,\theta}{1-tg^2\theta}$ there will be obtained the new relation $\frac{1}{\lambda}=\frac{2t}{1-t^2}$ which is equivalent to the quadratic equation $t^2+2\lambda t-1=0$. One solution for t and then for c and s is given by the relation below:

$$t=-\mathrm{sgn}\,\lambda\left(|\lambda|+\sqrt{1+\lambda^2}\right) \text{ and } c=\frac{1}{\sqrt{1+t^2}}\ ;\ s=ct \qquad (11)$$

V. Systolic Array for the SVD of a Matrix

We present a systolic array consisting of 9 processors (generally the number of processors is RxR). The systolic array first computes the values $g_{ij}=\left(a_i\right)^T \cdot a_j$. After receiving the row norms $\|a_i\|^2$ and $\|a_j\|^2$, each processor computes the rotation values s_{ij} and c_{ij} according to the formulas (8) and (10). The initial position of the corresponding systolic array is given in the figure below:

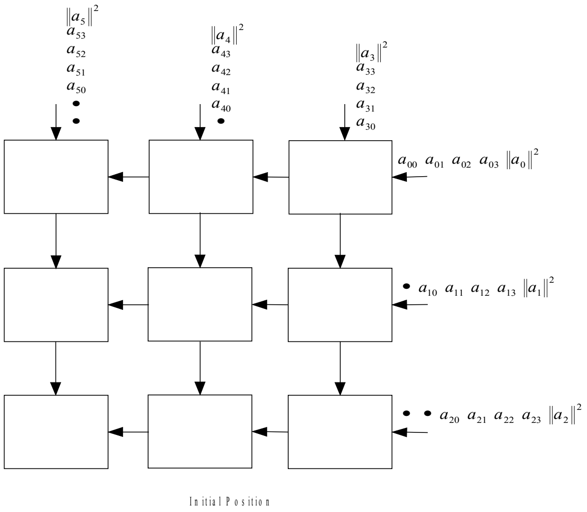

$\|a_5\|^2$ a_{53} a_{52} a_{51} a_{50}
$\|a_4\|^2$ a_{43} a_{42} a_{41} a_{40}
$\|a_3\|^2$ a_{33} a_{32} a_{31} a_{30}
a_{00} a_{01} a_{02} a_{03} $\|a_0\|^2$
a_{10} a_{11} a_{12} a_{13} $\|a_1\|^2$
a_{20} a_{21} a_{22} a_{23} $\|a_2\|^2$
Initial Position

Fig. 1 Systolic computation of rotation angles according (11) for R=3 and n=4

In Fig. 1 it is shown the first step of the systolic array for computing the SVD of a matrix of order mxn $(m=6,\ n=4)$. After 13 steps (the data movement proceeds systolically such that the first three rows move horizontally from right to left and the last three rows move vertically from top to bottom) the values c_{ij} and s_{ij} are computed. The processor that receives row i from the right and row j from the top computes c_{ij} and s_{ij} . Then, the same array may be used for obtaining the correspondent values of the matrix A_{k+1} according to the relation (6). This is the second part of the systolic array. In the same way as in the previous case, the processor receives values from right and top. Then it uses the calculated values for c_{ij} and s_{ij} , and the generalized form of the relation (6) to compute the values of a^{k+1} . The initial form of this array is given in the Fig. 2.

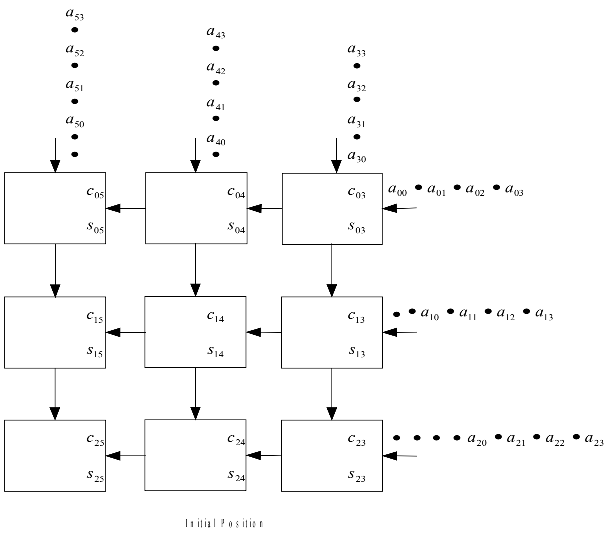

a_{53} a_{52} a_{51} a_{50}
a_{43} a_{42} a_{41} a_{40}
a_{33} a_{32} a_{31} a_{30}
c_{05} s_{05}
c_{04} s_{04}
c_{03} s_{03}
a_{00} a_{01} a_{02} a_{03}
c_{15} s_{15}
c_{14} s_{14}
c_{13} s_{13}
a_{10} a_{11} a_{12} a_{13}
c_{25} s_{25}
c_{24} s_{24}
c_{23} s_{23}
a_{20} a_{21} a_{22} a_{23}
Initial Position

Fig. 2 Obtaining the values of A_{k+1} using the values c_{ij} and s_{ij} computed in fig. 1

In the presented cases there are required 13 time steps for computing the values of c_{ij} and s_{ij} (Fig.1), as well as 15 time steps in the calculation of the values of A_{k+1} (Fig. 2). In general, in the case of Fig.1 there are required $n+1$ time steps for the computation of the values of inner product $g_{ij}=\left(a_i\right)^T \cdot a_j$. According to (11) there are used 4 time steps for the computation (one step for each of factors λ, c, t, s). We have to take into the consideration that this computation will start 2(R-1) time steps later. So the total number of time steps in the case of fig. 1 is n +2R+3. In the case of Fig. 2 there are used more number of steps because of the use of two multiplicative and additive operations. The total number is 2(n +R)+1. In the case presented in Fig. 2 these operations are merged together.

Therefore the exact number of time steps will be twice bigger. So, the total number of time steps will be n +2R+3+4(n +R)+1=5 n +6R+4.

VI. SVD OF A COMPLEX MATRIX

The polar representation of each complex number $z = a + ib$ may be written in the form $z = R_z e^{i\theta_z}$, where

$$e^{i\theta} = \cos\theta + i\sin\theta, \; R_z = \sqrt{a^2 + b^2},$$
$$\theta_z = \tan^{-1}\left(\frac{b}{a}\right) \text{ and } 0 \le \theta_z \le 2\pi$$

Let

$$\mathrm{M} = \begin{bmatrix} a_{11} + ib_{11} & a_{12} + ib_{12} \\ a_{21} + ib_{21} & a_{22} + ib_{22} \end{bmatrix} = \begin{bmatrix} Ae^{i\theta_a} & Be^{i\theta_b} \\ Ce^{i\theta_c} & De^{i\theta_d} \end{bmatrix} \tag{12}$$

be a 2x2 complex matrix. There are proposed several unitary transformations for the diagonalization of a complex 2x2 matrix. The two-sided unitary transformation proposed in [3] is:

$$\begin{bmatrix} c_\phi e^{i\theta_\alpha} & -s_\phi e^{i\theta_\beta} \\ s_\phi e^{i\theta_\gamma} & c_\phi e^{i\theta_\delta} \end{bmatrix} \begin{bmatrix} Ae^{i\theta_a} & Be^{i\theta_b} \\ Ce^{i\theta_c} & De^{i\theta_d} \end{bmatrix} \begin{bmatrix} c_\psi e^{i\theta_\xi} & -s_\psi e^{i\theta_\eta} \\ s_\psi e^{i\theta_\mu} & c_\psi e^{i\theta_\omega} \end{bmatrix} \tag{13}$$
$$= \begin{bmatrix} We^{i\theta_\omega} & 0 \\ 0 & Ze^{i\theta_z} \end{bmatrix}$$

where

$$\tan(\theta_\alpha - \theta_\beta) = -\frac{AC\sin(\theta_a - \theta_c) + BD(\theta_b - \theta_d)}{AC\cos(\theta_\alpha - \theta_\beta) + BD\cos(\theta_b - \theta_d)}$$
$$\tan(\theta_\eta - \theta_\omega) = -\frac{AB\sin(\theta_a - \theta_b) + CD(\theta_c - \theta_d)}{AB\cos(\theta_a - \theta_b) + CD\cos(\theta_c - \theta_d)} \tag{14}$$
$$\tan(\theta_\phi - \theta_\psi) = -\frac{Be^{i(\theta_\alpha + \theta_\omega + \theta_b)} - Ce^{i(\theta_\beta + \theta_\eta + \theta_c)}}{De^{i(\theta_\beta + \theta_\omega + \theta_d)} - Ce^{i(\theta_\alpha + \theta_\eta + \theta_a)}}$$
$$\tan(\theta_\phi + \theta_\psi) = -\frac{Be^{i(\theta_\alpha + \theta_\omega + \theta_b)} - Ce^{i(\theta_\beta + \theta_\eta + \theta_c)}}{De^{i(\theta_\beta + \theta_\omega + \theta_d)} - Ce^{i(\theta_\alpha + \theta_\eta + \theta_a)}}$$

If M is a real matrix then conditions given above can be simplified taking:

$$\theta_\alpha = \theta_\beta = \theta_\gamma = \theta_\delta = 0$$

The obtained result is identical to the two by two transformation given in (3).

Another method is proposed in [6]. This method is given by the two-sided unitary transformation:

$$\begin{bmatrix} \chi^{\frac{1}{2}} c_\xi & -\chi^{\frac{1}{2}} s_\xi \\ -\chi^{\frac{1}{2}} s_{\xi'} & \chi^{\frac{1}{2}} c_{\xi'} \end{bmatrix} \begin{bmatrix} Ae^{i\theta_a} & Be^{i\theta_b} \\ Ce^{i\theta_c} & De^{i\theta_d} \end{bmatrix} \begin{bmatrix} \chi^{\frac{1}{2}} c_z & -\chi^{\frac{1}{2}} s_{z'} \\ \chi^{\frac{1}{2}} s_z & \chi^{\frac{1}{2}} c_{z'} \end{bmatrix} \tag{15}$$
$$= \begin{bmatrix} We^{i\theta_\omega} & 0 \\ 0 & Ze^{i\theta_z} \end{bmatrix}$$

where

$$z = \psi + i\phi, \; z' = \psi - i\phi,$$
$$\xi = \varepsilon + i\eta, \; \xi' = \varepsilon - i\eta,$$
$$\chi = \sec h2\phi$$

Transformations in (15) are unitary because the fact that $\chi(c^2 + s^2) = 1$.

The main objective in the ordering the SVD of a complex matrix is transforming that into the real form. For that purpose some special forms of transformations are given which will help in that contest.

Transformation 1: The transformation that converts two elements of a complex 2x2 matrix into real values (two elements of a second row) is given by:

$$\begin{bmatrix} e^{i\theta_\alpha} & 0 \\ 0 & e^{i\theta_\alpha} \end{bmatrix} \begin{bmatrix} Ae^{i\theta_a} & Be^{i\theta_b} \\ Ce^{i\theta_c} & De^{i\theta_d} \end{bmatrix} \begin{bmatrix} e^{i\theta_\beta} & 0 \\ 0 & e^{-i\theta_\beta} \end{bmatrix}$$
$$= \begin{bmatrix} Ae^{i\theta_\omega} & Be^{i\theta_x} \\ C & D \end{bmatrix} \tag{16}$$

where

$$\theta_\alpha = -\frac{\theta_d + \theta_c}{2} \text{ and } \theta_\beta = \frac{\theta_d - \theta_c}{2}$$

Proof: The left side of (16) is equal with:

$$\begin{bmatrix} A \cdot e^{i(\theta_\alpha + \theta_a + \theta_\beta)} & B \cdot e^{i(\theta_\alpha + \theta_b - \theta_\beta)} \\ C \cdot e^{i(\theta_\alpha + \theta_c + \theta_\beta)} & D \cdot e^{i(\theta_\alpha + \theta_d - \theta_\beta)} \end{bmatrix}$$

Summing and subtracting the equalities $2\theta_\alpha = -\theta_d - \theta_c$ and $2\theta_\beta = \theta_d - \theta_c$ we have: $\theta_\alpha + \theta_\beta + \theta_c = 0$ and $\theta_\alpha + \theta_d - \theta_\beta = 0$. Hence, the two elements of the second row will be real numbers.

Similarly there can be obtained two other transformations given below:

Transformation 2: The transformation that converts two elements of a complex 2x2 matrix into real values (two elements of a second column) is given by:

$$\begin{bmatrix} e^{i\theta_\alpha} & 0 \\ 0 & e^{-i\theta_\alpha}\end{bmatrix}\begin{bmatrix} Ae^{i\theta_a} & Be^{i\theta_b} \\ Ce^{i\theta_c} & De^{i\theta_d}\end{bmatrix}\begin{bmatrix} e^{i\theta_\beta} & 0 \\ 0 & e^{i\theta_\beta}\end{bmatrix} = \begin{bmatrix} Ae^{i\theta_\omega} & B \\ Ce^{i\theta_y} & D\end{bmatrix} \qquad (17)$$

where

$$\theta_\alpha = \frac{\theta_d - \theta_b}{2} \text{ and } \theta_\beta = -\frac{\theta_d + \theta_b}{2}$$

Transformation 3: The transformation that converts two diagonal elements (main diagonal) of a complex 2x2 matrix into real values is given by:

$$\begin{bmatrix} e^{i\theta_\alpha} & 0 \\ 0 & e^{i\theta_\alpha}\end{bmatrix}\begin{bmatrix} Ae^{i\theta_a} & Be^{i\theta_b} \\ Ce^{i\theta_c} & De^{i\theta_d}\end{bmatrix}\begin{bmatrix} e^{i\theta_\beta} & 0 \\ 0 & e^{-i\theta_\beta}\end{bmatrix} = \begin{bmatrix} A & Be^{i\theta_x} \\ Ce^{i\theta_y} & D\end{bmatrix} \qquad (18)$$

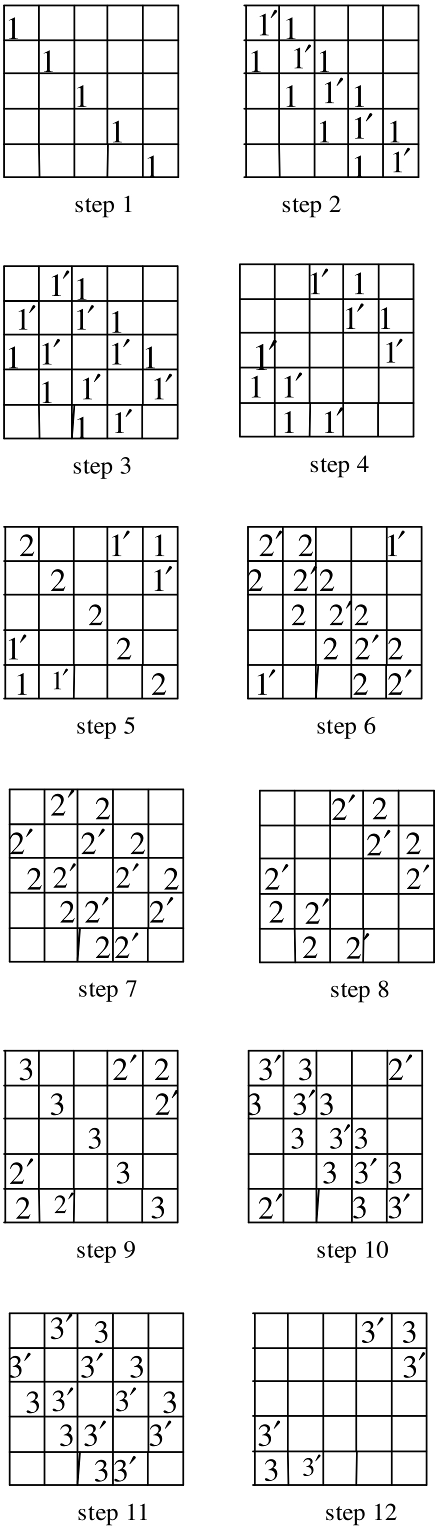
where

$$\theta_\alpha = -\frac{\theta_d + \theta_a}{2} \text{ and } \theta_\beta = \frac{\theta_d - \theta_a}{2}$$

Now, considering the transformations given above, it is possible to give general explanation of this procedure. For a complex matrix given as in (12), the first step is applying the transformation 1. After applying the transformation 2 and the transformation 3 the result will be a real matrix. Then the procedure is the same as in equations (2) and (3).

What about the systolic array of the SVD for a complex matrix? Each of the three transformations requires the generation and use of six angles, four unitary (a,b,c,d) and two rotational (α,β). These six angles must be propagated along both the rows and columns of processors on the main diagonal. The angles are generated by these processors and then are used for the diagonalization of a 2x2 complex matrices.

To improve the systolic array in the aspect of computation steps which are used, the performance will be as follows: during the computation by the diagonal processors using transformation 2, at the same time the nearest off diagonal processors may be used for computing using transformation 3. Similarly, the computations according transformation 3, may be performed before finishing the computations according transformation 1 and transformation 2. Thus, while the main diagonal are still computing the values according transformation 2, the nearest off diagonal processors are applying the transformation 1 etc. This pipelining of transformations by formulas (16), (17) and (18) is given in the following figure:



step 1 step 2

step 3 step 4

step 5 step 6

step 7 step 8

step 9 step 10

step 11 step 12

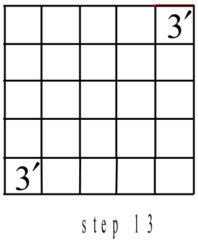


Fig. 3 Steps of computations on systolic array for the transforming the complex into the real matrix

After this procedure, the obtained matrix will be a real matrix, so the same systolic array as in figures 1 and 2 may be used for calculations.

References

[1] Carl G. J. Jacobi. *Uber eine neue Auflosungsart der bei der Methode der kleinsten Quadrate vorkommenden linearen Gleichungen.* Astronomishe Nachricten, 22, 1845. English translation by G.W. Stewart, Technical Report 2877, Department of Computer Science, University of Maryland, April 1992.

[2] G. H. Golub and C. F. Van Loan. *Matrix Computations.* John Hopkins University Press, Baltimore and London, 2nd edition, 1993.

[3] G. E. Forsythe and P. Henrici. *The cyclic Jacobi Method for Computing the Principal Values of a Complex Matrix.* Transactions of the American Mathematical Society, 94(1): 1-23, 1966.

[4] M. R.Hestenes. *Inversion of Matrices by Biorthogonalization and Related Results.* Journal of the Society for Industrial and Applied Mathematics, 6(1): 51-90, March 1958.

[5] R. P. Brent, F. T. Luk, and C. F. Van Loan. *Computation of the Singular Value Decomposition using Mesh-Connected Processors.* Journal of VLSI and Computer Systems, 1(3):242-270, 1985.

[6] E. G. Kogbetliantz, *Solution of Linear Equations by Diagonalization of Coefficients Matrix.* Quarterly of Applied Mathematics, 14(2): 123-132, 1955.

[7] G. H. Golub and C. F. Van Loan. *Matrix Computations,* Second edition . Johns Hopkins University Press,Baltimore, MD, 1989.

[8] C. M. Rader. Wafer Scale Systolic Array for Adaptive Antenna Proccessing. IEEE Int. Conf. on Acoustics, Speech and Signal Proccessing, 2069-2071, 1988.

[9] A. J. Van Der Veen and E. F. Deprettere. *A Parallel VLSI Direction Finding Algorithm.* Proc. SPIE Advanced Algorithms and Architectures for Signal Processing, 975(III): 289-299, 1988.

[10] J. R. Cavallaro and A. C. Elster. A Cordic *Proccessor Array for the SVD of a Complex Matrix*. In R. J. Vaccaro, editor, SVD and Signal Processing II, 227-239. Elsevier, New York, 1991.

[11] J. R. Cavallaro and F. T. Luk *CORDIC Arithmetic for an SVD Processor*. Journal of Parallel and Distributed Computing, 5(3):271-290, 1988.

[12] C. C. Sun and J. Gotze, *A VLSI design for parallel iterative algorithms.* Proceedings of the 9th int. conf. on Communications and information technologies, pp. 688-692, IEEE press Piscataway, NY, USA, 2009.

[13] Ahmedsaid, A., Amira, A., and Bouridane, *A.: Improved SVD systolic array and implementation on FPGA,* in: IEEE International Conference on Field Programmable Technologie, pp. 3-42, 2003.

[14] Gotze, J., Paul, S., and Sauer, M.: *An Efficient Jacobi-Like Algorithm for Parallel Eigenvalue Computation,* IEEE Transactions on Computers, 42, 1058-1065, 1993.

[15] Franklin T. Luk, A Parallel *Method for Computing the Generalized Singular Value Decomposition,* Journal of Parallel and Distributed Computing, volume 2, issue 3, pp. 250-260, august 1985,

[16] N. Suryanarayanan, *Efficient Architectures for Eigen Value Decomposition,* 2009, unpublished. (http://homepages.cae.wisc.edu/~ece734/project/s09/nikhil_rpt.pdf)

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 349–353

ISBN 978-83-60810-27-9
ISSN 1896-7094

Solving a Kind of BVP for ODEs on heterogeneous CPU + CUDA-enabled GPU Systems

Przemysław Stpiczyński
Institute of Theoretical and Applied Informatics
of the Polish Academy of Sciences, Gliwice, Poland
Department of Computer Science
Maria Curie-Skłodowska University, Lublin, Poland
Email: przem@hektor.umcs.lublin.pl

Joanna Potiopa
Department of Computer Science
Maria Curie-Skłodowska University
Lublin, Poland
Email: joannap@hektor.umcs.lublin.pl

***Abstract*—The aim of this paper is to show that a special kind of boundary value problem for second-order ordinary differential equations which reduces to the problem of solving a tridiagonal system of linear equations with almost Toeplitz structure can be efficiently solved on modern heterogeneous computer architectures based on CPU and GPU processors using an algorithm based on the *divide and conquer* method for solving linear recurrence systems with constant coefficients.**

I. Introduction

SEVERAL problems in scientific computing can be reduced to the following boundary value problem [8]:

$$-\frac{d^2u}{dx^2} = f(x) \quad \forall x \in [0,1], \tag{1}$$

where

$$u'(0) = 0 \text{ and } u(1) = 0. \tag{2}$$

Numerical solution to the problem (1)–(2) reduces to the problem of solving a tridiagonal system of linear equations. Simple algorithms based on Gaussian elimination achieve poor performance, since they do not fully utilize the underlying hardware, i.e. memory hierarchies, vector extensions and multiple (or many) processors (cores). The matrix of such systems have a special (almost Toeplitz) form and it clear that it should also be exploit. The problem can be solved in parallel using the *divide and conquer* approach [10] and novel data formats for dense matrices with the square blocked full column major order [2]. The performance of the algorithm can also be improved by using non-square tiles [11] which can better fit into L1 cache [1].

Graphical processing units (GPUs [5]) have recently been widely used for scientific computing due to their large number of parallel processors which can be exploit using the Compute Unified Device Architecture (CUDA) programming language [4]. GPUs offer very high performance at low costs for data-parallel computational tasks, when computations are carried out in single precision [3], [7]. 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 (possibly using double precision to improve numerical properties of algorithms).

The aim of this paper is to show that the divide and conquer method for solving (1) can be efficiently implemented on such heterogeneous systems including a multicore CPU and CUDA-enabled GPU.

II. The method

We want to find an approximation of the solution to the problem (1) in the grid points

$$0 = x_1 < x_2 < \ldots < x_{n+1} = 1,$$

where $x_i = (i-1)h$, $h = 1/n$, $i = 1, \ldots, n+1$. Let $f_i = f(x_i)$ and $u_i = u(x_i)$. Using the approximation for the second derivative

$$u''(x_i) \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

and the boundary conditions

$$u''(0) \approx \frac{u(0-h) - 2u_1 + u_2}{h^2}, \quad u'(0) \approx \frac{u_2 - u(0-h)}{2h}$$

we get the following equations

$$\begin{array}{rcl} 2(u_1 - u_2) & = & h^2 f_1, \\ -u_{i-1} + 2u_i - u_{i+1} & = & h^2 f_i, \ i = 2, \ldots, n-1, \\ -u_{n-1} + 2u_n & = & h^2 f_n. \end{array} \tag{3}$$

Thus, we can rewrite (3) as the problem of solving the system of linear equations [8]:

$$A\mathbf{u} = \mathbf{d}, \tag{4}$$

where the matrix A is of the following form

$$A = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{n \times n},$$

and the vectors satisfy $\mathbf{u} = (u_1, \ldots, u_n)^T$, $\mathbf{d} = (d_1, \ldots, d_n)^T$ and $d_1 = \frac{1}{2}h^2 f_1$, $d_i = h^2 f_i$ for $i = 2, \ldots, n$. The matrix A

can be factorized as $A = LR$, where L and R are bidiagonal Toeplitz matrices

$$L = \begin{pmatrix} 1 & & & & \\ -1 & 1 & & & \\ & \ddots & \ddots & & \\ & & -1 & 1 & \\ & & & -1 & 1 \end{pmatrix} \tag{5}$$

and

$$R = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \\ & & & & 1 \end{pmatrix}. \tag{6}$$

The solution to the system (4) can be found using Gaussian elimination without pivoting. A simple algorithm for solving (4) based on such a factorization can comprise two sequential stages, namely solving the systems $L\mathbf{y} = \mathbf{d}$ (forward reduction) and $R\mathbf{u} = \mathbf{y}$ (back substitution). The first stage can be done using

$$\begin{cases} y_1 = d_1 \\ y_i = d_i + y_{i-1} \qquad \text{for } i = 2, \ldots, n. \end{cases} \tag{7}$$

Then (second stage) we find the final solution to the system (4) using

$$\begin{cases} u_n = y_n \\ u_i = y_i + u_{i+1} \qquad \text{for } i = n-1, n-2, \ldots, 1. \end{cases} \tag{8}$$

More sophisticated approach can be based on the *divide-and-conquer* algorithm for solving linear recurrence systems [9]. The main idea of the algorithm is to rewrite the considered systems as block-bidiagonal systems of linear equations [9], [12]. Without loss of generality, let us assume that there exist two positive integers r and s such that $rs \le n$ and $s > 1$. The method can be used for finding $y_1, \ldots, y_{rs}$. To find $y_{rs+1}, \ldots, y_n$, we use (7) directly. For $j = 1, \ldots, r$, we define vectors

$$\mathbf{d}_j = \left(d_{(j-1)s+1}, \ldots, d_{js}\right)^T \in \mathbb{R}^s$$

and

$$\mathbf{y}_j = \left(y_{(j-1)s+1}, \ldots, y_{js}\right)^T \in \mathbb{R}^s.$$

Then we find all $\mathbf{y}_j$ using the following formula

$$\begin{cases} \mathbf{y}_1 = L_s^{-1}\mathbf{d}_1 \\ \mathbf{y}_j = L_s^{-1}\mathbf{d}_j + y_{(j-1)s}\mathbf{e} \qquad \text{for } j = 2, \ldots, r, \end{cases} \tag{9}$$

where $\mathbf{e} = (1, \ldots, 1)^T$ and the matrix $L_s \in \mathbb{R}^{s \times s}$ is of the same form as L given by (5). Analogously, we can perform the second stage using

$$\begin{cases} \mathbf{u}_r = R_s^{-1}\mathbf{y}_r \\ \mathbf{u}_j = R_s^{-1}\mathbf{y}_j + u_{js+1}\mathbf{e} \qquad \text{for } j = r-1, \ldots, 1, \end{cases} \tag{10}$$

where $\mathbf{u}_j = \left(u_{(j-1)s+1}, \ldots, u_{js}\right)^T \in \mathbb{R}^s$ and $R_s \in \mathbb{R}^{s \times s}$ is of the same form as R given by (6). However, it is better to find the following matrix

$$U = (\mathbf{u}_1, \ldots, \mathbf{u}_r) \in \mathbb{R}^{s \times r} \tag{11}$$

instead of individual vectors $\mathbf{u}_j$. Also, there is no need to use temporary vectors $\mathbf{y}_j$. It is clear that we have to allocate only one $s \times r$ array U and initially store vectors $\mathbf{d}_j$ in its columns. Then (Step 1A) we use a previously updated row to update the next row of the array. During Step 1B we update the bottom row of the table. Finally (Step 1C) we update $s-1$ first entries of each column (except for the first one). Similarly we proceed during the second stage of the algorithm (Figure 1). During Step 2A we update rows of the table *bottom-up*. Then (Step 2B) we update the first row of the table from right to left. Finally (Step 2C) using elements from the first row, we update the remaining entries of each column (except for the last one).

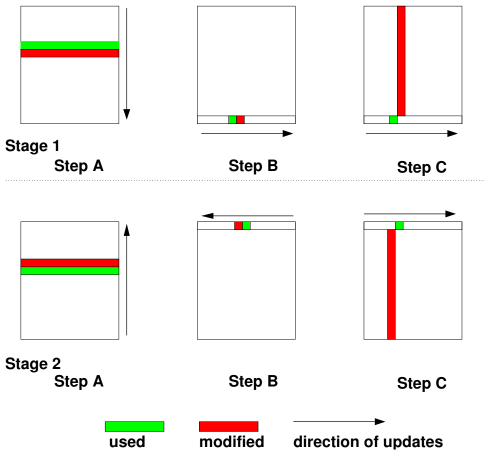


Fig. 1. *Divide and conquer* algorithm using a 2-D array

Both stages can be easily vectorized and parallelized. The array can be divided into blocks of columns and each processor can be responsible for computing one block. Steps A and C can be performed in parallel but steps B (in both stages) are sequential.

III. CUDA-BASED IMPLEMENTATION OF THE ALGORITHM

The detailed description of nVIDIA CUDA architecture can be found in [6]. Such a *computing device* (usually GPU) comprises a number of streaming multiprocessors (SM). Each SM consists of eight scalar cores (streaming processors, SP). SMs are responsible for executing blocks of threads. Threads in a block are grouped into so-called *warps*, each consisting of 32, that are managed and executed together. A device has its own memory system including the *global memory* (large but slow), *constant* and *texture* read-only memories providing reduction of memory latency. Each SM has also a 16 kB 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 by all threads of 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 and threads must access words in a sequence, namely k-th thread in a half-warp must access the k-th word in a segment [6]. CUDA programs consist of a number of C functions called *kernels* that are to be executed on devices as

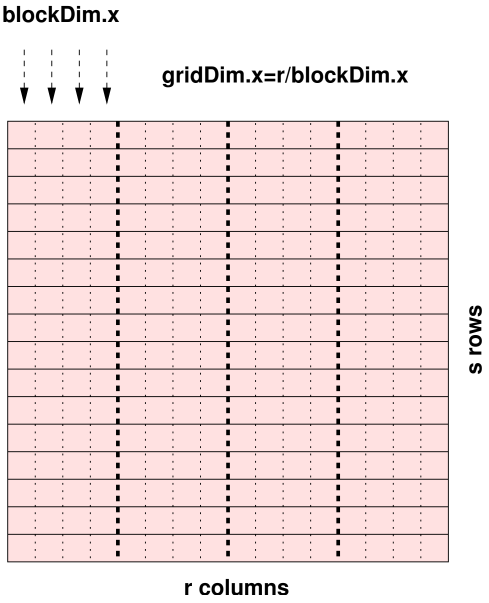


Fig. 2. Storage for the *divide and conquer* algorithm on GPUs

threads. Kernels are called from programs executed on CPUs. Host programs are also responsible for allocation of variables in the device global memory. The are also some CUDA API-functions used to copy data between host and device global memories. Each thread can identify its number, the number of its block and the block size using built-in variables `threadIdx`, `blockIdx` and `blockDim`, respectively.

Now let us consider the implementation of the *divide and conquer* algorithm. The basic idea is depicted in Figure 2. To allow coalesced memory access, elements of the $s \times r$ array U should be stored row-wise in the global memory of a device. Each thread is responsible for computing one column of the array. The number of its column can be computed as follows:

```
m =blockIdx.x*blockDim.x+threadIdx.x;
```

Each block of threads is responsible for computing one *panel* – a group of adjacent columns. For simplicity, we assume that

$$r = (\#blocks) \times (\#threads\ in\ block).$$

The source code of the host function is shown in Figure 3. Just after the array in the global memory of a device and auxiliary arrays in the host memory are allocated, the first kernel is executed to initialize the array (Figure 4). Note that the parallelized steps, namely 1A, 1C, 2A and 2C, are executed as kernels (see Figures 5–7), while sequential steps 1B and 2B are executed on CPU using double precision. In case of Step 1B, the bottom row of the array is copied from the device global memory, then it is updated using (9) and finally it is sent back to the device global memory. Analogously during Step 2B, the top row of the array is copied and updated using (10).

IV. Results of experiments

```
void cuda_bvp(int r, s, bsize, float *u_h){
/*
  s,r   - the number of rows and columns in the array
  bsize - the number of CUDA threads in a block
  u_h   - the array for the solution
*/

  int j,n=r*s;
  size_t size = N * sizeof(float);

// allocate arrays on host & device
  cudaMalloc((void **) &u_d, size);
  float  vf_tmp = (float *)malloc(r*sizeof(float));
  double vd_tmp = (double *)malloc(r*sizeof(double));

// initialize the s x r array on the device
  cuda_bvp_set <<< r/bsize, bsize >>> (u_d, s, r);

// perform Step 1A on the device
  cuda_bvp_1a <<< r/bsize, bsize >>> (u_d, s, r);

// perform Step 1B on CPU using double precision
  cudaMemcpy(vs_tmp, &u_d[(s-1)*r], sizeof(float)*r,
                                 cudaMemcpyDeviceToHost);
  vd_tmp[0]=vs_tmp[0];
  for(j=1;j<r;j++){
    vd_tmp[j] = (double)vs_tmp[j]+ vd_tmp[j-1];
    vs_tmp[j] = (float)vd_tmp[j];
  }
  cudaMemcpy(&u_d[(s-1)*r], vs_tmp, sizeof(float)*r,
                                 cudaMemcpyHostToDevice);

// perform step 1C and 2A on the device
  cuda_bvp_1c2a <<< r/bsize, bsize >>> (u_d, s, r);

// perform Step 2B on CPU using double precision
  cudaMemcpy(vs_tmp, u_d, sizeof(float)*r,
                                 cudaMemcpyDeviceToHost);
  vd_tmp[r-1]=vs_tmp[r-1];
  for(j=r-2;j>=0;j--){
    vd_tmp[j] = (double)vs_tmp[j]+vd_tmp[j+1];
    vs_tmp[j]=(float)vd_tmp[j];
  }
  cudaMemcpy(u_d, vs_tmp, sizeof(float)*r,
                                cudaMemcpyHostToDevice);

// perform Step 2C on the device
  cuda_bvp_2c <<< r/bsize, bsize >>> (u_d, s, r);

// copy results to the host array
  cudaMemcpy(u_h, u_d, sizeof(float)*N,
                                cudaMemcpyDeviceToHost);
}
```

Fig. 3. The source code of the *divide and conquer* algorithm

Both considered algorithms: **A1** – sequential based on (7) and (8), and **A2** – described in the previous section, have been tested on a computer with Intel Core2 Duo (2.66 MHz, 4GB RAM) and nVidia GeForce GTX 260 (216 cores, 1792MB RAM) running under Linux with `gcc` and nVidia `nvcc` compilers for various equations of the form (1) and problem sizes (Table I, II and III, IV). To observe the accuracy of the algorithms, we have considered the following special cases:

- **A1S** – sequential algorithm based on (7) and (8) using single precision, executed on CPU,
- **A1D** – sequential algorithm based on (7) and (8) using double precision, executed on CPU,
- **A2S** – parallel algorithm, steps 1A, 1C, 2A, 2C executed on GPU, steps 1B, 2B executed on CPU, all compu-

```
__global__ void cuda_bvp_set(float *u, int s,int r{

  int k,m;
  float h=1.0/((float)(s*r));
  float h2=h*h;

// the number of my column
  m=blockIdx.x*blockDim.x+threadIdx.x;

  for(k=0;k<s;k++){
    u[m+k*r] = h2*f(h*(m*s+k));
    // f should be replaced by
    // the right hand function !!!
  }

  if((blockIdx.x==0)&&(threadIdx.x==0))
    u[0]*=0.5;
}
```

Fig. 4. CUDA kernel for the initialization of the array

```
__global__ void cuda_bvp_1a(float *u, int s, int r){

  int k,m;

  m=blockIdx.x*blockDim.x+threadIdx.x;

  for(k=1;k<s;k++)
    u[m+k*r]+=u[m+(k-1)*r];
}
```

Fig. 5. CUDA kernel for Step 1A

```
__global__ void cuda_bvp_1c2a(float *u, int s,int r){

  int m,k;

  m=blockIdx.x*blockDim.x+threadIdx.x;

// step 1C
  if(m>0) {
    float a=u[(s-1)*r+m-1];
    for (k=0;k<s-1;k++)
        u[m+k*r]+=a;
  }

// step 2A
  for(k=s-2;k>=0;k--)
    u[m+k*r]+=u[m+(k+1)*r];
}
```

Fig. 6. CUDA kernel for Step 2A and Step 1C

```
__global__ void cuda_bvp_2c(float *u, int s, int r) {
  int m,k;

  m=blockIdx.x*blockDim.x+threadIdx.x;

  if(m!=r-1) {
    float a=u[m+1];
    for (k=1;k<s;k++)
        u[m+k*r]+=a;
  }
}
```

Fig. 7. CUDA kernel for Step 2C

tations in single precision,

A2D – the same as **A2S**, but all computations in double precision,

A2SD – parallel algorithm, steps 1A, 1C, 2A, 2C executed on GPU (using single precision), steps 1B, 2B executed on CPU (double precision).

As exemplary test problems, let us consider the following.

P1: Solve

$$-\frac{d^2u}{dx^2} = \frac{\pi^2}{4}\cos(\frac{\pi}{2}x) \quad \forall x \in [0,1], \tag{12}$$

with boundary conditions (2). The exact solution to (12) is as follows

$$u(x) = \cos(\frac{\pi}{2}x).$$

P2: Solve

$$-\frac{d^2u}{dx^2} = 20000e^{-100x^2}(1-200x^2) \quad \forall x \in [0,1], \tag{13}$$

with boundary conditions (2). The exact solution to (13) is as follows

$$u(x) = 100e^{-100x^2} - 100e^{-100}.$$

Tables I and II show the accuracy of the considered algorithms for both problems. The relative error of the computed solution is calculated due to

$$error = \frac{\|\mathbf{u}-\bar{\mathbf{u}}\|_2}{\|\mathbf{u}\|_2} \tag{14}$$

where $\mathbf{u}$ and $\bar{\mathbf{u}}$ are the exact and computed solutions respectively, and $\|\cdot\|_2$ is the Euclidean norm. Tables III and IV show execution time (in seconds) of **A1D**, **A2D** and **A2SD**, and speedup of **A2D** over **A1D**. Note that it is impossible to run **A2D** on our GPU for $n = 268435456$ (2048MB of RAM required, but only 1792MB available), thus tables III and IV do not contain results for **A2D** and $n = 268435456$. Moreover, for this case, tables I and II show the results obtained on CPU using the nVidia compiler simulation feature (the option `-deviceemu`). We can observe that

- the algorithm **A1S** (single precision) achieves very poor accuracy, the use of double precision (Algorithm **A1D**) gives much better results,
- the algorithm **A2S** (single precision) achieves reasonable accuracy; the use of double precision during the steps 1B and 2B (Algorithm **A2SD**) improves the accuracy of the results,
- the algorithm **A2D** (double precision) gives better accuracy than **A1D**,
- parallel algorithms **A2D** and **A2SD** are much faster than the sequential algorithm **A1D**; the speedup grows when the problem size (value of n) grows,
- **A2SD** is about 2-3 times than **A2D** and up to 95 times faster than **A1D**, thus if someone can accept results produced by **A2SD**, the use of **A2SD** can be profitable,
- CUDA-enabled GPUs are much slower when computations are carried out using double precision, however

TABLE I
RELATIVE ERROR OF THE ALGORITHMS FOR THE PROBLEM **P1**

n	**A1S**	**A1D**	**A2S**	**A2D**	**A2SD**
1048576	1.732620e-04	1.930917e-13	6.381482e-07	1.877603e-13	2.569113e-07
4194304	3.847218e-03	2.545430e-14	9.156066e-07	1.265400e-14	2.232157e-07
16777216	2.864740e-02	5.558266e-14	3.364642e-06	1.419160e-15	3.982652e-07
67108864	6.955456e-01	1.703449e-13	2.283507e-05	2.604335e-15	2.496036e-06
268435456	9.801750e-01	1.078518e-13	1.238516e-05	4.416135e-15	3.569703e-06

TABLE II
RELATIVE ERROR OF THE ALGORITHMS FOR THE PROBLEM **P2**

n	**A1S**	**A1D**	**A2S**	**A2D**	**A2SD**
1048576	2.618970e-03	1.314334e-11	2.893085e-06	1.312754e-11	1.341402e-07
4194304	8.263400e-03	9.951346e-13	5.194502e-06	8.205207e-13	1.115430e-05
16777216	4.822094e-02	5.650246e-13	2.334787e-05	5.152762e-14	1.634096e-05
67108864	1.723210e-01	1.884351e-12	6.568387e-05	6.961652e-15	4.321754e-05
268435456	2.839243e-01	6.616377e-13	1.179028e-05	1.320262e-14	6.531590e-06

TABLE III
EXECUTION TIME (SEC.) OF THE ALGORITHMS AND SPEEDUP OF **A2D** OVER **A1D** FOR THE PROBLEM **P1**

n	**A1D**	**A2D**	**A2SD**	**Speedup**
1048576	0.0460	0.0053	0.0029	8.62
4194304	0.1868	0.0102	0.0060	17.35
16777216	0.7570	0.0341	0.0134	22.32
67108864	3.0033	0.1167	0.0420	25.69
268435456	15.5967	N/A	0.1611	N/A

TABLE IV
EXECUTION TIME (SEC.) OF THE ALGORITHMS AND SPEEDUP OF **A2D** OVER **A1D** FOR THE PROBLEM **P2**

n	**A1D**	**A2D**	**A2SD**	**Speedup**
1048576	0.0492	0.0059	0.0040	8.23
4194304	0.2010	0.0119	0.0082	16.82
16777216	0.8076	0.0382	0.0194	21.12
67108864	3.2142	0.1284	0.0578	25.03
268435456	16.6759	N/A	0.2023	N/A

the next generation GPU architecture called Fermi is pretty much faster when double precision is used (visit `http://www.nvidia.com/` for more details), thus the use of **A2D** seems to be very attractive for such new devices.

V. CONCLUSIONS AND FUTURE WORK

We have showed that the kind of boundary value problem for second-order ordinary differential equations which reduces to the problem of solving a tridiagonal system of linear equations with almost Toeplitz structure can be efficiently solved on modern heterogeneous computer architectures based on CPU and GPU processors using the *divide and conquer* algorithm for solving linear recurrence systems with constant coefficients. The algorithm achieves reasonable accuracy and excellent speedup. We have showed that the use of double precision in sequential parts of the algorithm can improve its accuracy. In the future we will consider the problem of the numerical stability of our algorithm.

REFERENCES

[1] A. Buttari, J. Langou, J. Kurzak, and J. Dongarra, “A class of parallel tiled linear algebra algorithms for multicore architectures,” *Parallel Computing*, vol. 35, pp. 38–53, 2009.

[2] F. G. Gustavson, “New generalized data structures for matrices lead to a variety of high performance algorithms,” *Lect. Notes Comput. Sci.*, vol. 2328, pp. 418–436, 2002.

[3] A. Leist, D. P. Playne, and K. A. Hawick, “Exploiting graphical processing units for data-parallel scientific applications,” *Concurrency and Computation: Practice and Experience*, vol. 21, pp. 2400–2437, 2009.

[4] J. Nickolls, I. Buck, M. Garland, and K. Skadron, “Scalable parallel programming with CUDA,” *ACM Queue*, vol. 6, pp. 40–53, 2008.

[5] J. Nickolls and W. J. Dally, “The GPU computing era,” *IEEE Micro*, vol. 30, pp. 56–69, 2010.

[6] nVIDIA, *nVIDIA CUDA Programming Guide*. nVIDIA Corporation, 2009, available at http://www.nvidia.com/.

[7] S. Ryoo, C. I. Rodrigues, S. S. Stone, J. A. Stratton, S.-Z. Ueng, S. S. Baghsorkhi, and W. mei W. Hwu, “Program optimization carving for GPU computing,” *J. Parallel Distrib. Comput.*, vol. 68, no. 10, pp. 1389–1401, 2008.

[8] L. R. Scott, T. Clark, and B. Bagheri, *Scientific Parallel Computing*. Princeton University Press, 2005.

[9] P. Stpiczyński, “Solving linear recurrence systems using level 2 and 3 BLAS routines,” *Lecture Notes in Computer Science*, vol. 3019, pp. 1059–1066, 2004.

[10] P. Stpiczyński, “Solving a kind of boundary value problem for ODEs using novel data formats for dense matrices,” in *Proceedings of the International Multiconference on Computer Science and Information Technology*, M. Ganzha, M. Paprzycki, and T. Pełech-Pilichowski, Eds., vol. 3. IEEE Computer Society Press, 2008, pp. 293–296.

[11] P. Stpiczyński, “A parallel non-square tiled algorithm for solving a kind of BVP for second-order ODEs,” *Lecture Notes in Computer Science*, vol. 6067, pp. 87–94, 2010.

[12] P. Stpiczyński and M. Paprzycki, “Fully vectorized solver for linear recurrence systems with constant coefficients,” in *Proceedings of VECPAR 2000 – 4th International Meeting on Vector and Parallel Processing, Porto, June 2000*. Facultade de Engerharia do Universidade do Porto, 2000, pp. 541–551.

Computational Linguistics—Applications

The CLA Workshop is located within the framework of the IMCSIT conference to create a dialog between researchers and practitioners involved in Computational Linguistics and related areas of Information Technology.

IMSCIT is a multi-disciplinary conference gathering scientists form the different fields of IT & Computer Science together with representatives of industry and end-users. IMSCIT with its motto: "new ideas are born not inside peoples' heads but in the space between them", quickly became a unique place to share thoughts and ideas.

Workshop Goals

The Computational Linguistics – Applications Workshop was established in 2008 in response to the fast-paced progress in the area.

Traditionally, computational linguistics was limited to the scientists specialized in the processing of a natural language by computers. Scientific approaches and practical techniques come from linguistics, computer science, psychology, and mathematics. Nowadays, there is a number of practical applications available. These applications are sometimes developed by smart yet NLP-untrained developers who solve the problems using sophisticated heuristics. CLA aims to be a meeting place for both parties in order to share views and ideas. It will help scientist to better understand real world needs and practitioners not to reinvent the wheel.

Computational Linguistics needs to be applied to make the full use of the Internet. There is a definite need for software that can handle unstructured text and information to allow search for information on the web. The priority aim of the research in this area is to enable users to communicate with the computer in their native language.

CLA'10 Workshop is a place where the parties meet to exchange views and ideas with a benefit to all involved. The Workshop will focus on practical outcome of modeling human language use and the applications needed to improve human-machine interaction.

Paper Topics

This call is for papers that present research and practical developments on all aspects of Natural Language Processing used in real-life applications, such as (this list is not exhaustive):

- ambiguity resolution
- anaphora resolution
- applied CL software and systems
- computational morphology
- computational phonology
- corpus annotation and corpus-based language modeling
- creation of lexical resources
- dialogue systems
- entity recognition
- extraction of linguistic knowledge from text corpora
- information retrieval and information extraction
- machine learning methods applied to language processing
- machine translation and translation aids
- multi-lingual dialogue systems
- ontology and taxonomy evaluation
- opinion mining and sentiment classification
- paraphrasing and entailment
- parsing issues
- parts-of-speech tagging
- proofing tools
- prosody in dialogues
- question answering
- semantic networks and ontologies
- semantic role labeling
- semantic web
- speech recognition and generation
- summarization
- text classification
- text summarization
- word sense disambiguation

Program Committee

Alberto Abad, NESC-ID Lisboa, Portugal

Joseba Abaitua, Universidad de Deusto, Spain

Farag Ahmed, Otto-von-Guericke-University Magdeburg, Germany

Kisuh Ahn, Samsung Electronics Corporation, Korea, Republic of

Marianna Apidianaki, University College Ghent, Language and Translation Technology Team, Belgium

Naoya Arakawa, Agra Corp., Japan

Maria Aretoulaki (PhD), DialogCONNECTION Ltd, United Kingdom

Tania Avgustinova, DFKI GmbH, Germany

Mateusz Berezecki, Digg Inc., USA

Aliaksei Bondarionok, Microsoft Bing, USA

Janne Bondi Johannessen, University of Oslo, Norway

Wieslaw Byrski, Wyższa Szkoła Zarządzania i Bankowości, Poland

Nicoletta Calzolari, ILC-CNR,, Italy

Tommaso Caselli, ILC – CNR, Italy

Borys Czerniejewski, IPM sp. z o.o., Poland

Brian Davis, DERI Galway, NUIG, Ireland

Eric De La Clergerie, INRIA, France

Gerard de Melo, Max Planck Institute for Informatics, Germany

Luca Dini, CELI, Italy

Elzbieta Dura, Lexware Labs Ltd, University of Skovde, Sweden

Krzysztof Dyczkowski, Adam Mickiewicz University, Faculty of Mathematics and Computer Science, Poland

Matthias Eck, Carnegie Mellon University – Silicon Valley, USA

Agata Filipowska, Uniwersytet Ekonomiczny w Poznaniu, Poland

Rob Freeman, Independent, New Zealand

Piotr W. Fuglewicz, TiP Sp. z o.o., Poland

Mirosław Gajer, AGH University of Science and Technology, Poland

Geleijnse Gijs, Philips Research, Netherlands

Lee Gillam, University of Surrey, United Kingdom

Voula Giouli, Institute for Language & Speech Processing / Athena RC, Greece

Filip Graliński, Adam Mickiewicz University, Poland

Gregory Grefenstette, Exalead, France

Christian F Hempelmann, RiverGlass, Inc., USA

Ales Horak, Masaryk University, Czech Republic

Krzysztof Jassem, UAM, Poland

Marcin Junczys-Dowmunt, Adam Mickiewicz University, Poland

Heiki Kaalep, University of Tartu, Estonia

László Kálmán, Department of Theoretical Linguistics, Hungary

Shashi Kant, Cognika Corporation, USA

Alexander Kharlamov, Institute of higher nervous activity and neurophysiology, Russian Academy of Science, Russian Federation

Yuriy Koroliov, Ukraine

Natalia Kotsyba, Warsaw University, Poland

Marek Kowalkiewicz, SAP Research, Australia

Joern Kreutel, Germany

Sebastian Kruk, Knowledge Hives sp. z o.o., Poland

Olga Lazarenko, Kharkov University of Humanities, Ukraine

Charles Lehalle, Crédit Agricole Cheuvreux, France

Igor Leturia Azkarate, Elhuyar Fundazioa, Spain

Johannes Leveling, Dublin City University, Ireland

Andre Lynum, University of Oslo, Norway

Bente Maegaard, University of Copenhagen, Denmark

Cerstin Mahlow, University of Zurich, Switzerland

David Martins de Matos, INESC ID Lisboa / IST, Portugal

Stan Matwin, Canada

Saeedeh Momtazi, Saarland University, Germany

Stavros Ntalampiras, University of Patras, Greece

Proscovia Olango, The University of Groningen, Netherlands

Paulo Oliveira, University of Joinville, Brazil

Petya Osenova, Sofia University and Bulgarian Academy of Sciences, Bulgaria

Vincenzo Pallotta, Webster University, Switzerland

Marco Palomino, University of Westminster, United Kingdom

Joana Paulo Pardal, L2F INESC-ID Lisboa / IST Tech Univ Lisbon, Portugal

Maciej Piasecki, Wroclaw University of Technology, Poland

Michael Piotrowski, University of Zurich, Switzerland

Jakub Piskorski, Polish Academy of Sciences, Poland

Violaine Prince, France

Gabor Proszeky, MorphoLogic & Pázmány University, Hungary

Adam Przepiórkowski, Institute of Computer Science, Polish Academy of Sciences, Poland

Didier Schwab, Laboratoire d'Informatique de Grenoble, France

Giovanni Semeraro, Univ. of Bari "Aldo Moro", Italy

Violeta Seretan, University of Geneva, Switzerland

Dmitry Shaporenkov, Microsoft, Norway

Siyamed Sinir, Karniyarik LLC , Turkey

Daniel Sonntag, DFKI – German Research Center for AI, Germany

Sofia Stamou, Patras University, Greece

Veronica Stefan, Associate Professor PhD at Valahia University of Targoviste, Romania

Holger Stenzhorn, Saarland University Hospital, Germany

Petr Strossa, University of Economics, Prague, Czech Republic

Stan Szpakowicz, SITE, University of Ottawa, Canada

Isabel Trancoso, INESC-ID / IST, Portugal

Thorsten Trippel, Universität Tübingen, Germany

Alexander Troussov, IBM Ireland, Ireland

Brian Ulicny, VIStology, Inc., USA

Aristides Vagelatos, Research Academic Computer Technology Institute, Greece

Vincent Vandeghinste, Katholieke Universiteit Leuven, Belgium

Paula Vaz Lobo, Spoken Language Laboratory/INESC-ID/IST, Portugal

Pavel Velikhov, NIISI RAS, Russian Federation

Simo Vihjanen, Lingsoft, Inc., Finland

Ruprecht Von Waldenfels, Germany

Krzysztof Węcel, Poznań University of Economics, Poland

Eric Wehrli, University of Geneva, Switzerland

Sander Wubben, Tilburg university, Netherlands

Feiyu Xu, DFKI GmbH, Germany

Tianfang Yao, Shanghai Jiao Tong University, China

Tetiana Zabolotnia, NTUU "KPI", Ukraine

Annie Zaenen, Palo Alto Research Center (PARC), USA

Artur Zarski, Microsoft, Poland

Organizing Committee

Piotr W. Fuglewicz, TiP Sp. z o.o., Poland

Krzysztof Jassem (Chairman), UAM, Poland

Maciej Piasecki, Wroclaw University of Technology, Poland

Adam Przepiórkowski, Institute of Computer Science, Polish Academy of Sciences, Poland

ISBN 978-83-60810-27-9
ISSN 1896-7094

Using Self Organizing Map to Cluster Arabic Crime Documents

Meshrif Alruily, Aladdin Ayesh, Abdulsamad Al-Marghilani
Software Technology Research Laboratory
De Montfort University
The Gateway, Leicester, LE1 9BH UK
Email: meshrif,aayesh,abduls@dmu.ac.uk

***Abstract*—This paper presents a system that combines two text mining techniques; information extraction and clustering. A rule-based approach is used to perform the information extraction task, based on the dependency relation between some intransitive verbs and prepositions. This relationship helps in extracting types of crime from documents within the crime domain. With regard to the clustering task, the Self Organizing Map (SOM) is used to cluster Arabic crime documents based on crime types. This work is then validated through experiments, the results of which show that the techniques developed here are promising.**

I. Introduction

ONE OF the most important motivations for creating this system is because of the lack of Arabic systems in general, and the crime domain in particular. Furthermore, the crime domain has been chosen as an application area because of its social importance. Information extraction aims to extract specific, predefined entities from text. In this current research, one of our aims is to develop a system that is able to recognize crime phrases in a given document in order to extract types of crime. Feldman and Sanger [1] have stated that entities, such as peoples names, organizations names, locations, attributes (e.g. age of a person), and crime type can all be extracted. According to Toral and Munoz [2] and Collins and Singer [3], there are two types of evidence that help in identifying entities. Internal evidence can be deduced from the sentence that contains the entity by noticing a particular sequence of words. On the other hand, external evidence is gained from the context. In the first stage, the rule based approach (based on syntactical analysis) is adopted. In the second stage, the extracted types of crime are then used to by Self Organizing Map (SOM) in order to perform clustering. So instead of processing the whole content of each document, the rule based approach is used to guide the SOM to cluster the data by extracting important or meaningful patterns. According to Flexer [4] SOM is a very common tool for clustering and visualizing high dimensional data spaces. More details about SOM will be presented in one of the following sections.

The rest of the paper is organized as follows. In section II, a background and a review of the related work are given. The crime domain is described in section III. Domain analysis is presented in section IV. Section V presents the proposed clustering system. Section VI provides the results of the experiments and an evaluation of their performance. Finally, the conclusion of this work is presented in section VII.

II. Background

According to Michailidis [5], the Message Understanding Conference (MUC-6) introduced Named Entity Recognition (NER) in 1995, and it has been used in many different text-based applications, such as information extraction, question and answering, information retrieval and text classification [5], [6]. The approaches that are used to identify named entities are as follows [5]:

- Hand-craft rules, known as linguistic approaches.
- Machine learning approaches.
- Hybrid, which combines hand-craft recognition grammar with machine learning methods.

Alruily et al. [7] have developed a software package to extract crime information from texts. Their approach is based on a dictionary that is created manually. On the other hand, the task of automatically constructing lists of entities has been studied by many researchers. Riloff [8] has developed a program called AutoSlog that automatically constructs a domain-specific dictionary for information extraction. Nadeau et al. [9] used an approach that has two aspects: retrieve pages with seed, and a web page wrapper in order to build or generate gazetteers. Toral and Munoz [2] have proposed an approach to automatically build and maintain dictionaries of proper nouns using a noun hierarchy and a POS tagger. Also, Chau et al. [10] used named entity extraction techniques to identify meaningful entities from police narrative reports. To the authors' knowledge, there is no work available regarding the SOM technique applied to Arabic texts within the crime context for crime analysis. On the other hand, in the English language, Chen et al. [11] have developed a system based on SOM to cluster and visualize crime-related data.

III. Crime Domain

As far as we know, no information systems have been applied to the crime domain in the Arabic language. Therefore, the major problem we faced was the lack of data. This issue has been solved by compiling news articles on crime incidents, published by some Arabic newspapers. The reason for exploiting newspapers is that it is difficult to obtain official reports or narrative reports from police stations, especially in

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Arab countries. The news articles contain the information that the police reports would normally include. So, collecting these data was an important step in gaining a better understanding of the crime domain and the nature of the data that our system will deal with.

A. Types of Crime

The crime domain includes several types of crime, starting from civic crimes, such as drinking and driving, to international crimes, for instance, homicide by terrorists [12]. In this research, types of crime have been categorized into six main types, as in Table I.

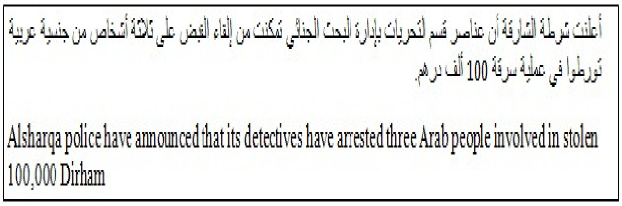
TABLE I
TYPES OF CRIME.

English	Arabic	Pronunciation
Theft	السرقة	Alsareqah
Fraud	الاَحتيَال	alehtial
Drug and alcohol smuggling	تهريب المخدرَات	Tahreeb almokhdrat
Magic and sorcery	السحر وَالشعوذه	Alseher walshaawathah
Sex crime	الجنس	Aljens
Violent crime	العنف	Alonf

As previously mentioned, the aim is to extract crime types from Arabic news articles for work that will be presented later. This extracted information is considered as ”keywords”; which are significant words that are able to give clues about the main idea of the document or article. Consequently, keywords play an important role in many text mining tasks, such as clustering, summarization and document retrieval. In this research, the extracted words will be treated as keywords to guide the Self Organizing Map (SOM) in order to perform clustering. So, the previous task of keyword extraction is an important process that must be considered carefully. Accordingly, the crime domain must be studied and its characteristics explored in order to find the appropriate extraction algorithm.

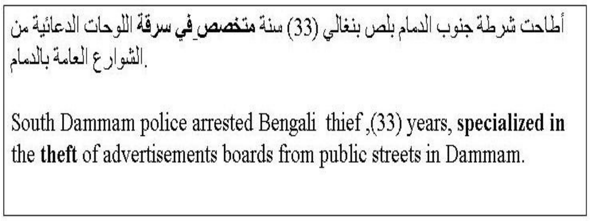
B. Event Description

Most Arabic newspaper reports have the same structure, with respect to writing style, in the crime domain. Most journalists or reporters start with a sentence containing the name of the police station that has investigated the crime, followed by a description of that crime. The crime description is about the type of crime committed and the type of criminal. Following these, details of any victims and other information are described. The reason for this formulaic approach is because they are dealing with a specialist domain that has its own language, and which can be called here the language of crime. According to Almas and Ahmad [13], each special language has a limited vocabulary and idiosyncratic syntactic structures. The journalists who work with these restricted languages seem to share the same words and same sentence structures. In other words, the usage of their words has the same behaviour. Fig. 1 shows a description for a theft crime published by Alriyadh newspaper [14]. The location of the committed crime can be deduced from the following pattern: ”شرطة الشَارقة / shurtat alsharqa / Alsharga police” . Furthermore, other related crime information can be extracted from the text, such as the nationalities of the people involved in the crime, e.g. ”جنسية عربية / jnsyt arabiat / Arabic nationality”. The type of crime can also be extracted from this pattern: ”تورطوَا في سرقة / tawaratwo fi sareekat / involved in stealing”. Also, the following is another example that has been taken from Aljazirah newspaper articles [15]:


أعلنت شرطة الشارقة أن عناصر قسم التحريات بإدارة البحث الجنائي تمكنت من إلقاء القبض على ثلاثة أشخاص من جنسية عربية
تورطوا في عملية سرقة 100 ألف درهم.

Alsharqa police have announced that its detectives have arrested three Arab people involved in stolen 100,000 Dirham

Fig. 1. Article Excerpt from Alriyadh Newspaper.


أطاحت شرطة جنوب الدمام بلص بنغالي (33) سنة **متخصص_في سرقة** اللوحات الدعائية من
الشوارع العامة بالدمام.

South Dammam police arrested Bengali thief ,(33) years, **specialized in** the **theft** of advertisements boards from public streets in Dammam.

Fig. 2. Article Excerpt from Aljazirah Newspaper.

It can be seen from the above example in Fig. 2 that the same writing style is followed. The name of the police station, which carries the name of the location, is stated first. The nationality of the criminal is also mentioned, and the type of offense is described. However, the crime type information is what we want to concentrate on in this research.

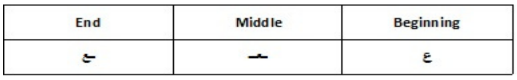
IV. DOMAIN ANALYSIS

A. Arabic Language

In this research, the Arabic language is studied; it is one of the Semitic languages and it is used in over 21 Arab countries. This language consists of 29 letters that can be used to form a word. Moreover, other languages, such as Farsi and Urdu use mostly Arabic characters [16]. From the sentence construction point of view, Arabic words can be divided into three classes: nouns, verbs and particles [17], [18], [19]. When working with the Arabic language, some other important characteristics need to be taken into account [20]:

1) A character may have up to three different forms, each form corresponds to the position of that character in the word (beginning, middle or end), such as letter ”ع / Ayn ” in Table II.

TABLE II
POSITION OF THE CHARACTER IN THE WORD.



End	Middle	Beginning
ـع	ـعـ	عـ

2) Arabic does not have capital letters; this characteristic represents a considerable obstacle to the NER task because in other languages capital letters represent a very important feature.
3) Finally, it is a language with a very complex morphology because it is highly inflectional.

A linguistic study of Arabic words and grammatical structures will be required before extracting the most appropriate structures for common Arabic sentence forms within the crime domain. Hence, the use of the linguistic internal structures of Arabic sentences will allow us to identify logical sequences of words. As previously mentioned, the structure of Arabic can comprise of three categories : noun (اسم), verb (فعل) and particle (حرف).

- Noun
  This category in Arabic comprehends any word that describes a thing, idea or person. It can be divided into two types: primitive and derivative. Primitive nouns are nouns that are not derived. Derivative nouns are nouns that are derived from verbs, other nouns, and particles. The Arabic nouns are inflected for gender (masculine and feminine) and number (singular, dual and plural). Also Nouns are either definite, which starts with the article ”ال / al” or indefinite, which has no ”ال” article at the beginning of the noun. Moreover affixes and clitics, such as some prepositions, conjunctions and possessive pronouns, can be attached to them. The clitic is subdivided into proclitic (located at the beginning of a stem) and enclitics (located at the end of a stem). For example, Table III shows the different morphological segments for the word ”وبدرجَاتهم” which means ”and by their grades”.

TABLE III
EXAMPLE FOR MORPHOLOGICAL SEGMENTS.

	enclitic	affix	stem	proclitic	proclitic
Arabic	هم	ات	درج	ب	و
pronunciation	hm	at	drg	be	wa
Gloss	their	s	grade	by	and

- Verb
  This word type points out an event or action. Verbs are also inflected in terms of number (singular, plural, dual), gender (masculine, feminine), person (1st, 2nd, 3rd), voice (active and passive) and mood (subjunctive, indicative and jussive). Furthermore, from the tense point of view, the verb can be in the past, present or future.
- Particle
  This class includes prepositions, conjunctions, interrogative particles, exceptions, and interjections. In other words, it includes the words that are not nouns or verbs, and sometimes these words are called function words.

In the Arabic language, sentences are divided into two types, as follows:

- A nominal sentence, according to Hadj et al [19], a nominal sentence can start with a noun or a particle.
- A verbal sentence
  A verbal sentence can start with a verb or a particle. The verb is divided into two types: transitive and intransitive. With respect to transitive verbs, a sentence, e.g., ”قطفت التفَاحة / qtft altfaht / I picked up the apple” contains one object or more as well as the subject. In other words, it takes more than one argument. On the other hand, a sentence that contains an intransitive verb has no object, and is composed of a verb and only one argument, e.g., ”مرض خَالد / mrd Khaled / Khaled became ill”. In some cases, some intransitive verbs can be converted into transitive verbs, for example, ”ذهبت إِلَى دبي / thhbt ela dubai / I went to Dubai”. In this example, the verb has a subject and a quasi sentence (the prepositional phrase) ”إِلَى دبي / ela dubai / to Dubai” which is the complement of the sentence ”ذهبت / thhbt / I went”; this help in determining the meaning of the whole sentence. Most Arab linguists state that most of the intransitive verbs can not refer to the object of the sentence but they can be strengthened by some prepositions, which are called transitive prepositions, such as ”البَاء / bi / because”, ”الَّام / allam / for”, ”من / mn / from”, ”علَى / ala / on”, ”في / fi / in”, ”إِلَى / ela / to”, in order to refer to the object. These verbs are called ”transitive verbs by preposition” [21], and they play an important role in achieving our goal.

B. Intransitive Verbs and Prepositions in the Crime Domain

As shown in the above section, our study of the crime domain corpus has led us to identify the characteristics of the language used. The first feature is that the past tense is used when describing crime incidents, whether the verbs describe the action of the crime itself or indicate a phrase that carries information about the crime type. Moreover, the modifier, sometimes called the qualifier, such as prepositional phrases and adjectives, are used for describing the type of offence. However, in this research, we concentrate on using the characteristics of some intransitive verbs and their propositions in order to recognize the type of crime. In other words, the correlation or dependency relationship between some intransitive verbs and some prepositions will be exploited. The Arabic language has approximately fourteen prepositions, most of which are short. Most of them are formed from three letters, such as ”علَى / ala” or from two, such as ”في / fi”, but they can be formed with only one Arabic letter, such as ”ل / li” or ”ب / bi”. The structure of a prepositional phrase is usually composed of two parts: preposition and noun-phrase [22]. For example, ”الولد في البيت / alwalad fi albyt” means ”the boy in the house”. In this example, the preposition is ”في / fi” and the noun is ”البيت / albyt”. Table IV shows a list of Arabic prepositions with their English translations. Only the most frequently used prepositions in crime domain texts are presented but more illustration regarding the listed prepositions is given in order to clarify them semantically.

TABLE IV
LIST OF PREPOSITION IN ARABIC LANGUAGE.

Arabic Preposition	Pronunciation	English Translation
علَى	Ala	on
في	Fi	in
إِلَى	Ela	to
ب	Bi	because
ل	Li	for

The preposition ”ب / bi” has different meanings, so based on a whole sentence, its meaning can be inferred. In this research context, the preposition ”ب / bi” means ”because”. In other words, it answers the question ”why”. Also, the preposition ”في / fi” represents the preposition ”in” in the English language, for example, ”تورط في قتل / twrat fi qtl / involved in killing” and ”متخصّص في قتل / motakasys fi qtl / specialized in killing”. With regard to the preposition ”ل / li”, it has many meanings but in the current domain being studied it again means ’because’; put simply, it justifies the reason, for example, ” هو ابلغ الشرطة لتعرضه للّسرقة / howa ablqa alshortat lita’arrudh lilsareekat / he reported to the police because he was robbed”. Thus, in this context ”ب / bi” and ”ل / li” carry very similar meanings. Sometimes the preposition ”إِلَى / ela” takes the place of the preposition ”ل” (e.g. ”تعرض إِلَى السرقة / ta’arrada ela lilsareekat / s/he was exposed to theft”). Thus, these prepositions can be utilized in many different ways. Additionally, they can work as a link between some verbs and nouns, whether they are adjacent or not. Moreover, the meanings of sentences that contain some specific verbs cannot be identified without prepositional phrases. Based on this, the main forms where a type of crime is located in the text can be identified. Fig. 3 depicts this case. Thus, in order to mark the crime phrases in the text, the system looks for the verbs in the text and (their prepositions), and the type of crime should not be more than three words away from the preposition. Therefore, only three words are extracted after the preposition. As a result, in order to recognize and extract types of crime, the list of these intransitive verbs and their prepositions (that indicate the patterns of crime type) must be defined beforehand.

From the above illustrations, it can be seen that there are strong correlations between some verbs and prepositions. These relationships are considered as key elements, and form an important part in accomplishing this work. Accordingly, these can lead to discovering a local grammar for crime type patterns, which will help in their extraction. Fig. 4 shows the various frequent patterns used in news articles for describing the type of crime committed.

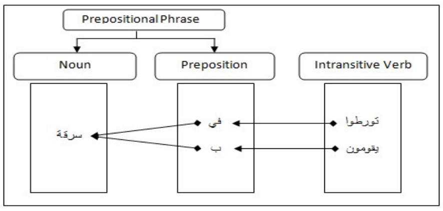


Fig. 3. Types of crime in Prepositional Phrases.

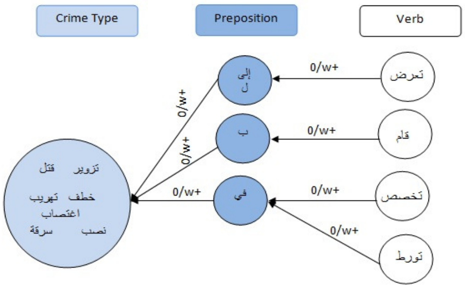


Fig. 4. Crime types Local Grammar.

V. CLUSTERING SYSTEM

Fig. 5 shows the proposed framework, which consists of stages, divided as follows:

- Normalization
  In this stage, some letters that perform the same function, and that can be written in different forms in a word, are normalized. For example, the letters ”أ” , ”إ” and ”آ” are converted into the letter ا.
- Information Extraction
  The system here is focused on prepositional phrases. Hence, the dependency relationship between prepositions and intransitive verbs is exploited. The advantage of this method is that there is no need for an annotated corpus, whether manual or automatic. In other words, the system has no linguistic components, such as PoS taggers or shallow parsers. Instead, lists of intransitive verbs and their prepositions are provided to the system in order to extract the desired patterns. Fig. 4 shows the list of verbs, with their appropriate prepositions, that are used in this research. In other words, it describes the local grammar for extracting types of crime. To sum up, in the processing phase, the system looks for words in a text that match the words in the verb list. When a match occurs, the next step is to search for a preposition that always comes with the verb. After that, the three words

that immediately follow the preposition are extracted to represent the whole content of the document. Moreover, the crime type should appear within these three words.

- Stemming
  Through this process, the stem of a word can be obtained by eliminating the word's affixes. After stemming, the content of file is automatically converted into numbers by giving each word of interest a unique number. Also, during this process the stopwords are removed.
- Clustering and Visualization
  For the clustering process, the Self Organizing Map (SOM) has been chosen to cluster documents that were generated by the information extraction process, based on their similarity. The SOM technique is popular and widely used for clustering and visualizing high dimensional data spaces. According to Eyassu and Gamback [23] SOM has many different structures but the most popular architecture is composed of two layers of processing units; the input layer and the output layer. These two layers are fully interconnected. The idea behind SOM is that it performs mapping for similar input vectors to similar areas on the output grid. The following is the SOM algorithm:
  1) Initialize weight randomly
  2) Initialize neighbourhood ratio
  3) Set input pattern
  4) Calculate Euclidean distance
  5) Find the winner neuron (smaller distance)
  6) Update winner and neighbour weight neurons
  7) Repeat Steps 3 to 7 until the convergence criterion is satisfied

  As can be seen, this algorithm is iterative. The first step is to randomly initialize the weight vectors of the output map. At each iteration (training), a sample vector is randomly chosen from the input data. This phase is called the learning process or competitive learning. Through competitive learning, the Euclidean distance is calculated for choosing the Best Matching Unit (BMU). The wining neuron or BMU is the one most similar to the input pattern. That is, its weight is close to the input pattern. As a result, all neurons on the output layer enter into a competition with each other. The neuron on the output layer that has the smallest distance to the input pattern is the winner. Once the winning neuron has been selected, its weight and the weights of its neighbour are both updated in order to make them more similar to the input pattern. This process is repeated with other documents until accurate results are found or the maximum number of iterations (epochs) are reached.

VI. Experimental Result and Evaluation

A. Corpus

Text mining research relies on the availability of a suitable corpus. As a result, many corpora have been created for specific purposes. For this research, two corpora have been collected from different Arabic newspapers published in different Arabic countries, such as Alriyadh, Aljazeera and Okaz from Saudi Arabia, Elkhabar newspaper issues from Algeria, Addustour from Jordan and Ahram from Egypt. The first corpus contains 26 documents and the other includes 24 documents. The reason for compiling corpora from different resources is to avoid the problem of bias, which could occur if the system is tested on documents that were collected from only one country.

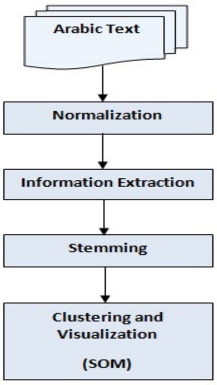


Fig. 5. System Architecture.

B. Experiments

Two experiments have been carried out on 26 documents in order to show how the information extraction process guides the Self Organizing Map (SOM) to gain accurate results. The size of this corpus is 25.3 KB. The SOM has been trained on the same documents, obtaining good results; the best learning rate, radius and iteration are 0.5, 8 and 1000, respectively.

- First Experiment
  In the first experiment, the information extraction process is used. So, instead of processing the whole of each document's content, the extracted patterns from each document are used by SOM to perform clustering. As a result, the size of the corpus has become 1.56 KB. Only 21 unique words represent the 26 text documents and they comprise 57 tokens. Fig. 6 shows the results of the first experiment after achieving the clustering process.

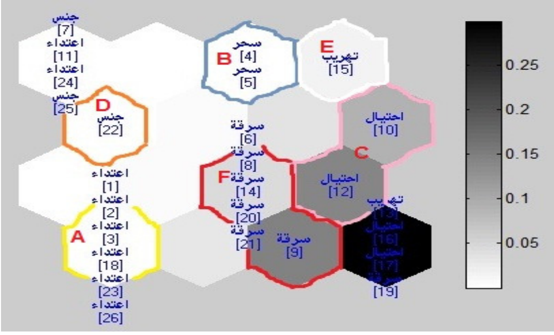


Fig. 6. Clustering Result for First Experiment (A: Violante, B: Magic, C: Fraud, D: Sex, E: Smuggling, F: Theft).

- Second Experiment
  This experiment does not rely on the information extraction process. So the whole content of each file is stemmed and used for the clustering process. The 26 textual files are represented by 32 unique words. These 32 words form 228 tokens. Also, the results of this experiment can be seen in Fig. 7.

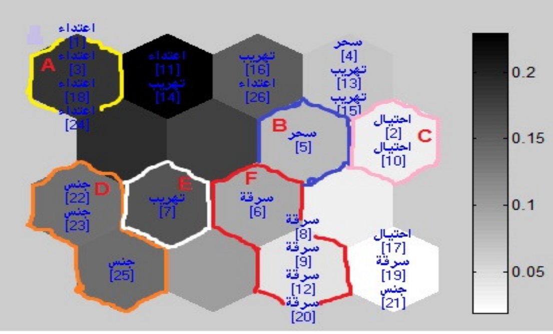


Fig. 7. Clustering Result for Second Experiment (A: Violante, B: Magic, C: Fraud, D: Sex, E: Smuggling, F: Theft).

In order to examine the proposed system and the SOM (at the same learning rate, radius and iteration value) on a new and untouched corpus other experiments have been also carried out. This new corpus contains 24 documents, and its size is 28.5 KB. The third and fourth experiments are as follows:

- Third experiment
  This is exactly like the first experiment because the information extraction process is used. The size of the whole corpus after the extraction became 1.83 KB. The number of unique words is 13 which, form 44 tokens. These 44 tokens are then used by SOM to cluster the documents. The clustering results can be seen in Fig. 8.

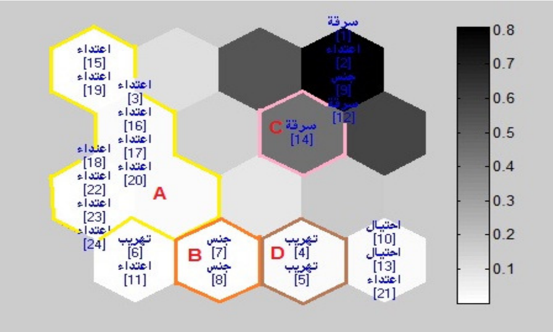


Fig. 8. Clustering Result for Third Experiment (A: Violante, B: Sex, C: Theft, D: Smuggling).

- Fourth experiment
  This experiment is as the second experiment. It uses the whole content of each document because no information extraction technique is used. 23 unique words represent the tested documents, and they form 235 tokens. As a result, 235 tokens are used by SOM to cluster the documents in this experiment. Fig. 9 shows the clustering outcome of SOM.

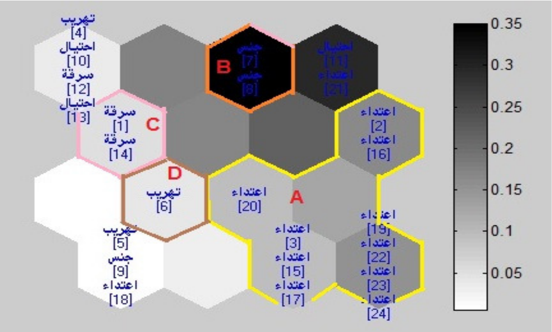


Fig. 9. Clustering Result for Fourth Experiment (A: Violante, B: Sex, C: Theft, D: Smuggling).

C. Result Analysis

The average distance between each data vector and its BMU (quantization error) in the first experiment is 0.686, and in the second experiment it is 0.949. So the performance of the SOM is better when using the information extraction process. As is well-known, SOM clusters documents based on their similarity. Moreover, each document is treated as a vector of words. So the number of a word's frequency that occurs in a file affects its clustering and sometimes this leads to a wrong cluster. For example, File 23 from Group A in Fig. 6 is clustered as a violent crime, which is true, but in Fig. 7 it is clustered as a sex crime. The reason for this clustering mistake in Experiment 2 is because of the phrase ”سعودي الجنسية / saudi aljensyat”, which means ”Saudi nationality”, occurred many times when talking about the criminal. So the word ”الجنسية / aljensyat” means ”nationality” in English, but after stemming this word by removing its affixes (the article ”ال” and the letter ”ة”) the word becomes ”جنس / jens”, which means ”sex”; this affected its clustering in the second experiment. Also, another clustering mistake occurred in Experiment 2 for File 14. This file belongs to Group F in the first experiment; i.e. it is clustered as a theft crime, but in the second experiment it has been labelled as ”smuggling”, and is far from its Group (F) in Fig. 7. The reason for this being wrongly clustered is because of the word ”هرب / hrb”, which appears many times in the file; it has two meanings in the Arabic language, and it means ”flee” or ”smuggle” in English. The file is totally about a theft crime but in the crime description the phrase ”thieves fled / هرب الّصوص ” is stated many times in different ways, and ”هرب / hrb” in our context means ”smuggle”. As a result, Group F in Experiment 2 does not contain the file number 14. Also, Fig. 10 shows that the letters attached to the vertical axis represent categories of crime types, as in Table V, and the numbers that are underneath the horizontal axis refer to the files. This graph clarifies how our system labels files with the right category names using the extracted patterns, and it worked better than when using the full content. In Experiment 1, the file numbers 12, 16 and 17 have incorrect crime type category names. On the other hand, in Experiment 2 file numbers 2, 7, 14, 17, 21 and 23 also have the wrong category names. With regards to Experiments 3 and 4, the quantization errors are 0.6 and 0.888, respectively.

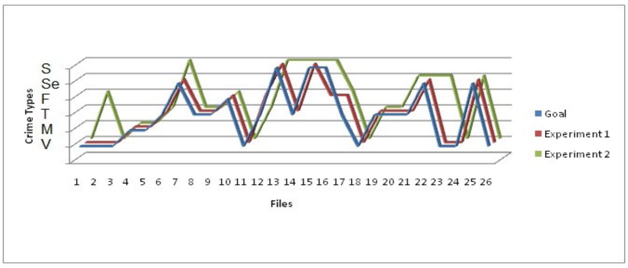


Fig. 10. Result of Labeling Crime Documents.

TABLE V
MEANING OF VERTICAL AXIS

Letter	V	M	T	F	Se	S
Crime Type	Violent	Magic	Theft	Fraud	Sex	Smuggling

VII. CONCLUSION

In this paper, we have developed a system for clustering textual documents containing information about different types of crimes. The Self Organizing Map (SOM) technique was chosen to perform the clustering. Moreover, the rule-based approach, based on intransitive verbs and propositions, was used to help in obtaining good clustering results. Also, a comparison study was carried out through four experiments in order to show the effects of the rule-based method on the Self Organizing Map. The results show that although SOM used fewer tokens in Experiments 1 and 3, it was able to achieve clustering that was as good as or better than in Experiments 2 and 4. Therefore, it can be confirmed that the SOM technique has been improved in terms of its performance. The reason behind this remarkable achievement is because of the system's ability to extract keywords based on syntactic principles, and this led directly to the improved clustering results.

REFERENCES

[1] R. Feldman and J. Sanger, "Information extraction," in *The Text Mining Handbook: Advanced Approches in Analyzing Unstructured Data: Cambridge university*, no. 94-130, 2006.

[2] A. Toral and R. Munoz, "A proposal to automatically build and maintain gazetteers for named entity recognition by using wikipedia," 2006.

[3] M. Collins and Y. Singer, "Unsupervised models for named entity classification," in *In Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999, pp. 100–110.

[4] A. Flexer, "On the use of self-organizing maps for clustering and vi," *Intelligent Da*, vol. 5, no. 5, pp. 371–384, 2001.

[5] I. Michailidis, K. Diamantaras, S. Vasileiadis, and Y. Frre, "Greek named entity recognition using support vector machines, maximum entropy and onetime," in *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006, pp. 47–52.

[6] P. Srikanth and K. N. Murthy, "Named entity recognition for telugu," in *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, 2008, pp. 41–50.

[7] M. Alruily, A. Ayesh, and H. Zedan, "Crime type document classification from arabic corpus," in *Second International Conference on Developments in eSystems Engineering*. Los Alamitos, CA, USA: IEEE Computer Society, 2009, pp. 153–159.

[8] E. Riloff, "Automatic constructing a dictionary for information extraction tasks," in *The Eleventh National Conference on Artifical Inteligence*, 1993, pp. 811–816.

[9] D. Nadeau, P. D. Turney, and S. Matwin, "Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity," 2006, pp. 266–277.

[10] M. Chau, J. J. Xu, and H. Chen, "Extracting meaningful entities from police narrative reports," in *dg.o '02: Proceedings of the 2002 annual national conference on Digital government research*. Digital Government Society of North America, 2002, pp. 1–5.

[11] H. Chen, H. Atabakhsh, T. Petersen, J. Schroeder, T. Buetow, L. Chaboya, C. OToole, M. Chau, T. Cushna, D. Casey, and Z. Huang, "Coplink: Visualization for crime analysis," in *dg.o 03: proceedings of the 2003 annual national conference on Digital government research*, 2003.

[12] P. Thongtae and S. Srisuk, "An analysis of data mining applications in crime domain," in *Proc. IEEE 8th International Conference on Computer and Information Technology Workshops CIT Workshops 2008*, 2008, pp. 122–126.

[13] Y. Almas and A. Kurshid, "Lolo: a system based on terminology for multilingual extraction," in *IEBeyondDoc 06: Proceeding of the Workdhop om Information Extraction Byeond The Document*, 2006, pp. 56–65.

[14] Alriyadh. Crimes articles. [Online]. Available: http://www.alriyadh.com/

[15] Aljazirah, "Crimes articles." [Online]. Available: http://www.al-jazirah.com/

[16] A. M. AL-SHATNAWI and K. OMAR, "Methods of arabic language baseline detection the state of art," *Arab Research Institute in Sciences and Engineering (ARISER)*, vol. 4, pp. 158–193, 2008.

[17] R. Al-Shalabi, G. Kanaan, B. Al-Sarayreh, K. Khanfer, A. Al-Ghonmein, H. Talhouni, and S. Al-Azazmeh, "Proper noun extracting algorithm for arabic language," in *International Conference on IT, Thailand*, 2009.

[18] S. KHOJA, "Apt: Arabic part-of-speech tagger," in *Proc. of the Student Workshop at NAACL*, 2001.

[19] Y. M. E. Hadj, I. Al-Sughayeir, and A. Al-Ansari, "Arabic part-of-speech tagging using the sentence structure," in *Proceeding of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt*, 2009, pp. 241–245.

[20] Y. Benajiba, P. Rosso, and J. Ruiz, "Anersys: An arabic named entity recognition system based on maximum entropy," in *CICLing*, 2007, pp. 143–153.

[21] معجم الَافعَال المتعدية بحرف, موسَى بن ˜محمد المليَاني ˜الأَحمدي.
بيروت, دَار العلم لّلَايين, 1986.

[22] A. C. Satterthwait, "Computational research in arabic," *Mechanical Translation*, vol. 7, pp. 62–70, 1963.

[23] S. Eyassu and B. Gamback, "Classifying Amharic news text using self-organizing maps," *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pp. 71–78, 2005.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Quality Benchmarking Relational Databases and Lucene in the TREC4 Adhoc Task Environment

Ahmet Arslan
Anadolu University
Computer Engineering Department
Eskisehir, Turkey
Email: aarslan2@anadolu.edu.tr

Ozgur Yilmazel
Anadolu University
Computer Engineering Department
Eskisehir, Turkey
Email: oyilmazel@anadolu.edu.tr

***Abstract*—The present work covers a comparison of the text retrieval qualities of open source relational databases and Lucene, which is a full text search engine library, over English documents. TREC-4 adhoc task is completed to compare both search effectiveness and search efficiency. Two relational database management systems and four different well-known English stemming algorithms have been tried. It has been found that language specific preprocessing improves retrieval quality for all systems. The results of the English text retrieval experiments by using Lucene are at par with top six results presented at TREC-4 automatic adhoc. Although open source relational databases integrated full text retrieval technology, their relevancy ranking mechanisms are not as good as Lucene's.**

I. Introduction

RELATIONAL database management systems (RDBMS) have been the preferred way of managing data for the past two decades. In recent years, data that many applications manage are moving more from structured to unstructured (free form text). Although relational databases are designed to handle structured data, many web applications use databases to manage and query unstructured data. With the increase in unstructured text, development of search engine libraries - which are specifically designed to quickly and effectively search large volumes of unstructured text - has been gaining momentum. There are several open source search engine libraries available with different features [1]. Many database vendors (IBM DB2[1], Microsoft SQL Server[2], MySQL[3], Oracle[4], PostgreSQL[5]) have recognized the need for free form text search and started implementing features that would support full-text search capabilities. Search engine libraries and relational databases each have unique advantages but also they have overlapping capabilities in common. In our previous work [2], we compared the full text search capabilities of different open source relational databases and Lucene on Turkish text documents. In this work we are extending our experiments by using English data set and deeply exploring full text search configuration parameters of relational databases.

The paper is organized as follows. Section II briefly summarizes the related work, Section III explains the retrieval systems and stemming algorithms that we used, Section IV shows experimental results and our analysis on them, Section V gives summary of our observations about relational databases' full text search, and Section VI provides concluding remarks.

II. Related Work

There are many studies which evaluated the performance of search engine libraries over Text REtrieval Conference (TREC) test collections. TREC[6] is an annual conference aiming to encourage research in information retrieval (IR) based on large test collections. Database vendors also have evaluated their full-text search capabilities by participating in TREC competitions [3], [4]. Earlier papers [5], [6], [7] have described full text search capabilities and features of different relational databases, focusing specifically on the integration of free form text and structured data. The studies on comparison of IR systems are common, but there are no studies on Relational Database Management Systems' information retrieval qualities. The other studies have focused on hybrid IR-DB system solutions and integration of IR and databases.

Some unpublished articles discuss and compare different aspects of relational databases and search engines libraries. In [8], Marc Krellenstein explores the benefits of a full text search engine in comparison to a database. The article by David Smiley [9], addresses a scenario in which a web application needs to have a full text search capability. It discusses using the text search features of relational database versus using Apache Solr[7] - an open source search platform built on top of Lucene.

In a recent paper Yinan Jing and Chunwang Zhang [10] compared Lucene and a relational database in terms of query time. The data set used in their work was composed of auto-generated numeric and alphanumeric fields. They performed their test on these structured fields which are not tokenized therefore they didn't use full-text search but rather structured queries. A systematic comparison of text retrieval quality of relational databases and search engine libraries has not been done.

[1] http://www-01.ibm.com/support/docview.wss?uid=swg27004103
[2] http://msdn.microsoft.com/en-us/library/ms142571.aspx
[3] http://dev.mysql.com/doc/refman/5.5/en/fulltext-search.html
[4] http://www.oracle.com/technology/products/text/index.html
[5] http://www.postgresql.org/docs/8.4/static/textsearch-intro.html
[6] http://trec.nist.gov/
[7] http://lucene.apache.org/solr/

978-83-60810-27-9/09/$25.00 © 2010 IEEE

III. Experimental Setup

In this study TREC-4 adhoc task was completed by querying forty nine topics (numbers 202-250) over 567,529 documents. Topic 201 was ignored since it retrieved no relevant documents and wasn't used in the actual evaluation of TREC-4 adhoc competition. DOCNO field - which is common for all documents - used as unique identifier in our experiments and the rest of the document - except DOCNO and DOCID fields - is taken as single textual field named contents. Topics are queried over this single field. The details of the document set and topics used in TREC-4 can be found here [11].

We used one open source information retrieval library (Apache Lucene 3.0.1) and two open source Relational Database Management Systems with full-text search capabilities (MySQL 5.5.3, PostgreSQL 8.4.4) for retrieval experiments. The latest versions of each system were used in our study in order to take account into latest features and improvements. We would like to experiment with other popular relational databases such as IBM DB2, Microsoft SQL Server and Oracle; however they do not permit disclosure results of any program benchmark tests without their prior consent.

All of the experiments were completed on Apple Mac Pro with two 2.8 GHz Quad-Core Intel Xeon processors and 6 GB 800 MHz DDR2 memory running Mac OS X Version 10.5.7. All test programs were implemented with the Java programming language, running on JDK 1.6.

A. Stemming Algorithms for English

Stemming is the most employed technique in IR to enhance retrieval quality in terms of recall. We investigated publicly available stemmers for the English language in our experiments. Probably the two most well-known stemming algorithms for English language are the Porter [12] stemming algorithm and the Lovins [13] stemming algorithm. Lovins is the first ever published stemming algorithm and removes 297 different endings using longest-match algorithm. Porter stemming algorithm created and maintained by Dr. Martin Porter. The algorithm can be best defined by its author's own words:

"The Porter stemming algorithm (or 'Porter stemmer') is a process for removing the commoner morphological and inflectional endings from words in English. Its main use is as part of a term normalisation process that is usually done when setting up Information Retrieval systems."[8]

English (Porter2)[9] stemming algorithm is a result of Dr. Martins attempt to improve the structure of the original Porter algorithm.

These algorithms are rule based and do not use dictionary or lexicon and preferred for their linear running time complexity since dictionary based stemmers can be sometimes slow for real-world web applications.

TABLE I
Analyzer Building Blocks

ootb	**Porter**	**Lovins, English**	**KStem**
LetterTokenizer	LetterTokenizer	LetterTokenizer	LetterTokenizer
LowerCaseFilter	LowerCaseFilter	LowerCaseFilter	LowerCaseFilter
StopFilter	StopFilter	StopFilter	StopFilter
	PorterStemFilter	SnowballFilter	KStemFilter

KStem [14] stemming algorithm is a dictionary based inflectional stemming algorithm which uses human readable dictionary. It is a less aggressive stemmer than the standard Porter stemmer. It was written by Bob Krovetz, ported to Lucene by Sergio Guzman-Lara (UMASS Amherst).

We used these four different stemming algorithms for English to improve text retrieval performances of each system. To measure this improvement we included out-of-the-box settings of each system. In out-of-the-box option no preprocessing is done on the documents and queries. Built-in English support and stop-word list of each system is used.

B. Lucene

Lucene[10] is a powerful, free, open source IR library written entirely in Java. It is suitable for nearly any application that requires full-text search and its popularity is increasing because of simplicity, high performance, maturity and scalability.

Analysis, in Lucene, is the process where free form text is converted into tokens by tokenization, lowercasing, stemming and etc. Analysis process begins with a Tokenizer which breaks free form text into tokens. And then, the created token stream is fed into nested TokenFilters. TokenFilters can add, modify or delete its input token stream. For example lowercasing, removing common words, reducing words to a common base form or injecting synonyms occurs in TokenFilters. Lucene has several Tokenizer and TokenFilter implementations. An analyzer is an encapsulation of the analysis process which is an essential part of Lucene. Custom Analyzers can be built from a Tokenizer and a TokenFilter chaining pattern.

Five different analyzers are used in our runs and Table I shows the Analyzer building blocks used to create them. First column of Table I represents StopAnalyzer that comes with out-of-the-box Lucene. StopAnalyzer's default English stopword set contains 33 common words.

LetterTokenizer divides text at non-letters by capturing tokens as maximal strings of adjacent letters, as defined by `java.lang.Character.isLetter()` method. We have used this tokenizer because TREC-4 topics do not have any alphanumeric or numeric words. All tokens are composed of letters. (Except three acronyms U.S., U.K. and e.g. which are taken as stop words in our runs) Therefore we didn't index any numeric or alphanumeric tokens.

LowercaseFilter normalizes the token by lowercasing its text.

[8] http://tartarus.org/\textasciitildemartin/PorterStemmer/

[9] http://snowball.tartarus.org/algorithms/english/stemmer.html

[10] http://lucene.apache.org

StopFilter removes tokens that exist in a provided list of stop words. In custom analyzers, we used 70 stopwords which are superset of StopAnalyzer's default stopword list.

SnowballFilter - that comes in contrib package of Lucene -, stems words using a Snowball-generated stemmer. Snowball[11], which is created by Dr. Martin, is a small string manipulation language specifically designed for creating stemming algorithms for use in Information Retrieval. A range of non-English stemmers are implemented by using snowball script, including Danish, Dutch, Finnish, French, German, Hungarian, Italian, Norwegian, Portuguese, Romanian, Russian, Spanish, Swedish and Turkish [15]. There exists three English-specific stemmers (named English, Lovins, and Porter) implemented by using snowball script and available at Dr. Martins web site. These exact names can be passed as parameter to constructor of SnowballFilter to initialize a stem filter for that language in Lucene. In our experiments we used English and Lovins.

PorterStemFilter is java implementation of Porter algorithm and has the same behavior as SnowballFilter when Porter is used for the name argument to the SnowballFilter constructor. PorterStemFilter is much faster since it does not use Snowball Program.

KStemFilter stems words according to Bob Krovetz's kstem algorithm. Lucene does not have KStemmer implementation with the out-of-the-box settings. Source code of the stemmer is downloaded from web site of Center for Intelligent Information Retrieval - University of Massachusetts Amherst[12].

StopAnalyzer and four different custom Lucene analyzers—representing each stemming option—are used to create five Lucene indices. The same analyzers are used to search topics over each index.

C. MySQL

MySQL has support for full-text indexing and searching based on a space-vector model. Full-text indices can be created only on CHAR, VARCHAR, or TEXT columns of MyISAM tables. Full-text searching is performed using MATCH()...AGAINST() syntax which is introduced on June 2000. However MySQL has no linguistic support (stemming) for English or any other language. MySQL has a default stopword list for English[13] and removes them during indexing and searching. Like words included in the built-in stop word list, also words that are less than four or greater than 84 characters are also ignored by default in full-text searches. Table II shows MySQL's user-overridable full-text search parameters including their default values and descriptions. These parameters are defined by the system variables and can be obtained by executing

```
SHOW VARIABLES LIKE 'ft_%';
```

SQL statement. The other modifications such as disabling 50% threshold and changing tokenization behavior are non-trivial tasks that require source code modification and recompilation of MySQL.

1) Indexing: Five different MySQL tables with two columns (docno, contents) are created. For out-of-the-box option documents are inserted directly into the table. For the remaining stemming options, first they passed through respective Lucene Analyzer and then inserted into their tables. Full-text indices are built on tables thus:

```
ALTER TABLE docs ADD FULLTEXT (contents);
```

The indices are created after loading all data to tables because for large data sets, it is much faster than loading data into tables that have an existing FULLTEXT index.

2) Searching: MySQL full text search have basically three modes: natural language mode, Boolean mode and with query expansion.

Natural Language Full-Text Search[14] performs a natural language search for a query against a text collection. There are no operators in this mode and the stop-word list applies. Another interesting property of this mode is the elimination of query words that occur in more than or equal to half of the collection. In another words if a query word is present in at least 50% of the documents, it is treated as a stop-word. This mode is the default mode in MySQL and automatically sorts search results in order of decreasing relevance. Run using this mode is executed as follows:

```
SELECT docno, MATCH (contents) AGAINST ('<topic>')
AS score FROM docs WHERE MATCH (contents)
AGAINST ('<topic>') LIMIT 1000;
```

Relevance ranking algorithm of Natural Language mode uses Vector Space Model where rows and queries are represented as weighted vectors. MySQL uses a variant of the classic tf-idf (term frequency-inverse document frequency) weighting scheme along with pivoted document length normalization. Details of the ranking algorithm can be found here [16].

Boolean Full-Text Search[15] allows usage of implied Boolean operators ([no operator], +, -) and various advanced search methods like wildcard (*) and phrase search. A leading plus sign means required or mandatory operator. The word after the plus sign must exist in every row returned. A leading minus sign means prohibited operator. The word after the minus sign must not exist in any row returned. When no operator is specified, it means that this word is optional and should exist in returned rows. Rows that contain optional words will get higher scores. Complete list of supported operators are shown at the first row of Table II. Stop-word list applies but 50% threshold limitation does not apply with this mode. This mode does not automatically sort search results in order of decreasing relevance therefore 'ORDER BY score DESC' clause is added to SQL sentence in two runs using this mode:

[11] http://snowball.tartarus.org

[12] http://ciir.cs.umass.edu/

[13] http://dev.mysql.com/doc/refman/5.5/en/fulltext-stopwords.html

[14] http://dev.mysql.com/doc/refman/5.5/en/fulltext-natural-language.html

[15] http://dev.mysql.com/doc/refman/5.5/en/fulltext-boolean.html

TABLE II
MYSQL VARIABLES ASSOCIATED WITH FULLTEXT SEARCHING

Variable Name	Default Value	Description
ft_boolean_syntax	+ - >< () ˜ * : ”” & |	List of operators supported by boolean full-text searches performed using IN BOOLEAN MODE.
ft_max_word_len	84	Maximum length of the word to be included in a FULLTEXT index.
ft_min_word_len	4	Minimum length of the word to be included in a FULLTEXT index.
ft_query_expansion_limit	20	Number of top matches to use for full-text searches performed using WITH QUERY EXPANSION
ft_stopword_file	(built-in)	File from which to read the list of stopwords for full-text searches.

SELECT docno, MATCH (contents) AGAINST ('<topic>' IN BOOLEAN MODE) AS score FROM docs WHERE MATCH (contents) AGAINST ('<topic>' IN BOOLEAN MODE) ORDER BY score DESC LIMIT 1000;

Relevance ranking algorithm of Boolean Mode is quite different from Natural Language Mode. This mode provides only simplistic relevance ranking [17]. It is defined as the sum of weights of matched words in query string. Weights are defined by boolean operators. This ranking mechanism produces always 1 when + operator is used before query terms. For example in the pure required type query '+term1 +term2 +term3 +term4' weights of each term will be 1/4. When no operator is used, score is equal to count of matching query terms. For example in the pure optional type query 'term1 term2 term3 term4' weights of each term will be 1. If a document contains three of these terms, it will get score of three. This ranking algorithm does not use collection-wide statistics (inverse document frequency) therefore Boolean Mode full text searches does not require FULLTEXT indices. Other interesting property of this ranking is that calculated scores are always greater than or equal to one.

Full-Text Searches with Query Expansion[16] applies classic blind relevance feedback which is also known as Pseudo relevance feedback. It performs natural language mode search twice and assumes top few (controlled by ft_query_expansion_limit variable which has a default value of 20) results of first search are relevant. And then, it appends these documents to the original query to perform second search. This mode can be activated by adding WITH QUERY EXPANSION or IN NATURAL LANGUAGE MODE WITH QUERY EXPANSION modifiers to the query. Functions of these two modifiers are exactly the same and they yield same results. Since this mode is modified version of a natural language search, it automatically sorts search results in order of decreasing relevance. Runs using this mode are executed as follows:

SELECT docno, MATCH (contents) AGAINST ('<topic>' WITH QUERY EXPANSION) AS score FROM docs WHERE MATCH (contents) AGAINST ('<topic>' WITH QUERY EXPANSION) LIMIT 1000;

D. PostgreSQL

PostgreSQL supports full text indexing of textual documents and relevance ranking for full text database searching.

In PostgreSQL, dictionaries[17] allow fine-grained control over how tokens are normalized. PostgreSQL provides several predefined dictionaries for linguistic support, available for many languages, and English is one of them.

PostgreSQL have two special data type tsvector and tsquery representing preprocessed documents and processed queries with support of boolean operators respectively. These data types are vector representation of documents and queries like in vector space model. There are some functions to convert documents or queries into these data types. to_tsvector is used to transform a document to tsvector data type while to_tsquery and plainto_tsquery are used for converting a query to the proper tsquery data type. All of these transformation functions take a language specific configuration parameter. Full text searching is done using the match operator @@, which returns true if a tsvector (document) matches a tsquery (query).

1) Indexing: Five different PostgreSQL tables with two columns (docno, contents) are created. An additional tsvector type column named ts_col is added to the table.

ALTER TABLE docs ADD COLUMN ts_col tsvector;

For out-of-the-box option, documents are inserted into the tables without any preprocessing. Then ts_col column is populated from contents column by invoking to_tsvector function with the configuration parameter for the English language that comes with out-of-the-box settings of PostgreSQL.

UPDATE docs SET ts_col = to_tsvector('english', contents);

For the remaining stemming options, documents are first analyzed by respective Lucene Analyzer and then inserted into their tables. The ts_col column is populated with the output of to_tsvector function, but this time using the simple template parameter which behaves like no language is specified because data in the table are already preprocessed.

UPDATE docs SET ts_col = to_tsvector ('pg_catalog.simple', contents);

PostgreSQL offers two kinds of indices that can be used to speed up full text searches.

- GiST (Generalized Search Tree) based index
- GIN (Generalized Inverted Index) based index

[16] http://dev.mysql.com/doc/refman/5.5/en/fulltext-query-expansion.html

[17] http://www.postgresql.org/docs/8.4/static/textsearch-dictionaries.html

TABLE III
POSTGRESQL DOCUMENT LENGTH NORMALIZATION OPTIONS

Mode	Meaning
0	(the default) ignores the document length
1	divides the rank by 1 + the logarithm of the document length
2	divides the rank by the document length
4	divides the rank by the mean harmonic distance between extents (this is implemented only by ts_rank_cd)
8	divides the rank by the number of unique words in document
16	divides the rank by 1 + the logarithm of the number of unique words in document
32	divides the rank by itself + 1

Since GIN index is best for static data and searches are about three times faster than GiST, GIN index was created to speed up the search as follows:

CREATE INDEX text_index ON docs USING gin (ts_col);

2) Searching: PostgreSQL provides two functions to_tsquery and plainto_tsquery for converting a query to the tsquery data type. plainto_tsquery transforms unformatted text querytext to tsquery by parsing and normalizing text, then inserting the & (AND) Boolean operator between surviving words. to_tsquery creates a tsquery value from querytext, which must consist of single tokens separated by the Boolean operators & (AND), | (OR) and ! (NOT). Note that to_tsquery with AND operator is identical to plainto_tsquery.

To rank search results, PostgreSQL provides two predefined ranking functions to calculate similarity between a tsvector (document) and a tsquery (query). These are standard ranking function (ts_rank) and cover density ranking function (ts_rank_cd) [18]. While the ts_rank does not consider term position proximity, the ts_rand_cd ranking function punishes documents where the search terms are further apart. PostgreSQL's both ranking functions do not use any global information (inverse document frequency); therefore indices are not mandatory but can be used to speed up full text searching.

Both ranking functions take an integer normalization option[18] that specifies whether and how document length normalization will be done. Table III shows the document length normalization options and their effect on ranking mechanism that PosgreSQL supports with the out-of-the-box settings.

Note that mode 32 is used just to scale ranks between zero and one. The ordering of search results does not change in this mode.

In out-of-the-box option to obtain pure OR queries, description parts of topics are tokenized at white spaces and then OR operator ("|") is inserted between each token. English language configuration is used in this mode. SQL queries are submitted as follows:

SELECT docno, *ranking_function*(ts_col, query,
normalization) AS rank FROM docs,
to_tsquery('english','<topic>') query WHERE query @@
ts_col ORDER BY rank DESC LIMIT 1000;

In remaining stemming options to obtain pure OR queries, description parts of topics are first passed through respective Lucene Analyzer and then OR operator ("|") is inserted between surviving words. Simple template configuration is used because queries are already analyzed. SQL queries are submitted as follows:

SELECT docno, *ranking_function*(ts_col, query,
normalization) AS rank FROM docs,
to_tsquery('pg_catalog.simple','<topic>') query WHERE
query @@ ts_col ORDER BY rank DESC LIMIT 1000;

IV. EXPERIMENTAL RESULTS

Since the overview of TREC-4 paper presents the precision/recall curves for the groups with the highest non-interpolated average precision (**MAP**) and the runs are ranked by the average precision, in this work the same metric is used in global evaluation. While evaluating each system in itself we also presented precision at 5 (**P@5**) and precision at 10 (**P@10**) values as well as search time (**sec/q**) per query. Additionally results of citri2 run of Royal Melbourne Institute of Technology [19] (one of the best TREC-4 automatic adhoc) were included in experimental result for comparison.

The evaluation measures presented in this paper are calculated by using Chris Buckley's `trec_eval`[19] package (version 8.1) which is the standard tool used by the TREC community for evaluating an adhoc retrieval run, given the results file and a standard set of judged results.

In our calculations a cut-off level of 1000 is used, which defines the retrieved set as the top 1000 documents in the ranked list which is similar to official TREC usage:

trec_eval -c -M1000 official_qrels submitted_results

All retrieval systems are designed in a similar fashion to standard TREC-type adhoc runs that retrieve maximum 1000 documents per topic.

Topics created for the TREC-4 adhoc task consist of only one field (description). Therefore we ran retrieval experiments over each index, by using description-only queries. In this work completely automatic query construction is used. Description only queries had on average 16 terms with stop words, 9 terms without stop words.

A. Lucene

Lucene's scoring[20] mechanism uses both the Vector Space Model and the Boolean Model. The Boolean model is used to first filter the documents to be used in score calculation. Lucene's scoring algorithm implements cosine similarity between tf-idf weighted documents and queries. It adds several factors to cosine similarity including document length normalization. Default length normalization function divides the score by square root of the number of words in the document.

[18] http://www.postgresql.org/docs/8.4/static/textsearch-controls.html

[19] http://trec.nist.gov/trec_eval/

[20] http://lucene.apache.org/java/3_0_1/scoring.html

TABLE IV
LUCENE SEARCH QUALITY COMPARISON

Run	MAP	P@5	P@10	sec/q
StopAnalyzer OR operator	**0.1645**	**0.4939**	**0.4163**	0.0452
StopAnalyzer AND operator	0.0011	0.0490	0.0286	0.0067

TABLE V
MYSQL SEARCH QUALITY COMPARISON

Run	MAP	P@5	P@10	sec/q
natural language mode	**0.1182**	**0.3388**	**0.3204**	1.2244
boolean mode AND operator	0.0023	0.0612	0.0449	0.0816
boolean mode OR operator	0.0318	0.1429	0.1041	3.4693
with query expansion limit 3	0.0249	0.0939	0.0755	8.1428
with query expansion limit 5	0.0405	0.1426	0.1122	10.4285
with query expansion limit 10	0.0326	0.1102	0.0939	14.7346
with query expansion limit 15	0.0229	0.0735	0.0633	18.0612
with query expansion limit 20	0.0331	0.1265	0.0898	20.6530

Details of the Lucene scoring can be found in chapter 3.3 of [15].

Lucene allows selecting between two Boolean operators (AND, OR) when performing search. Search quality results of these two operators are given in Table IV. Result set of OR operator is superset of the result set of AND operator. In other words, result set of OR operator already contains result set of AND operator. Moreover Lucene gives higher scores to the documents that contain more query terms. This fact implies that highest ranked documents will usually have the most ORed query terms among documents returned. OR operator used in remaining Lucene runs since it yields better results than AND operator.

B. MySQL

Total eight runs performed (with the out-of-the-box settings) to determine which type of MySQL full-text search is superior. Search quality results of our MySQL runs are given in Table V. It includes different boolean operators and search options described in section III-C2.

Boolean mode with pure required and pure optional queries performed badly due to its simplistic ranking mechanism which is described in section III-C2.

Blind relevance feedback (BRF) is used in TREC competitions and usually improves performance in TREC adhoc tasks. For example Cornell SMART system [20] at TREC 4 applied BRF (with good success) by adding the most frequently occurring 50 single terms and 10 phrases from the top 20 documents to initial query. However in MySQL it didn't perform as expected. To investigate this behavior we ran experiments with query expansion with five different ft_query_expansion_limit values. However results were still lower than natural language mode. We compared individual MAP values of natural language mode run and best with query expansion mode (limit 5) run and found that in 4 topics query expansion performed better. MySQL Reference Manual does not explain details of the query expansion mechanism but we suspect that this is due to appending whole document text to the initial query. Appending whole document to the initial query increases noise significantly and returns nonrelevant documents.

It is easy to understand TREC-like runs; natural language mode is more suitable due to its sophisticated ranking algorithm and the 50% threshold limitation. Query terms that occur in half of the documents in the collection have no distinctive property. Such words alone would return at least half of the documents in the collection. Natural language mode that yields best results in terms of retrieval quality is selected as best representative of MySQL.

C. PostgreSQL

In PostgreSQL, both matching operator (@@) and ranking functions takes same two parameters, tsquery and tsvector. We observed that when AND is used as matching operator, some topics did not return any documents while the others returned a few documents. Pure AND queries returned 13 documents per topic on the average. This behavior of AND operator yield very poor retrieval quality. Therefore - as described in section III-D2 - initially we used OR for both matching operator and ranking functions in our runs. By doing so, we obtained better results than AND operator.

When we examined our submitted qrels file, we observed that many documents ranked exactly with the same float value for a particular topic. Further investigations revealed that one of the ranking functions (ts_rank) does not play well with OR operator. As it can be understood from first two rows of Table VI ts_rank does not take account into how many ORed terms match when all query terms occur in a document. When it is used with AND operator, this behavior reverses and score increases with the number of matched terms. Interestingly ts_rank with AND operator (last row), does not yield zero for the document that does not contain all of the query terms. On the other hand ts_rank_cd function yields zero in this scenario. Also score produced by ts_rank_cd with OR operator increases as the number of the query terms found in the specified document increases.

After these observations we concluded that it is more convenient to use ts_rank_cd with OR operator while rank_cd with AND operator in our score calculations. We keep using OR operator for matching function in our remaining runs. Note that if we were to use ts_rank_cd with AND operator, the documents that do not contain all of the query terms would get a rank of zero. To obtain pure AND tsquery for use with ts_rank function, topics are fed into plainto_tsquery function. SQL sentence of runs using ts_rank is modified as follows:

SELECT docno, ts_rank(ts_col,
plainto_tsquery('english','<topic>'), *normalization*) AS
rank FROM docs WHERE to_tsquery('english','<topic>')
@@ ts_col ORDER BY rank DESC LIMIT 1000;

Total eleven different runs performed (with the out-of-the-box settings) to determine which ranking function and doc-

TABLE VI
POSTGRESQL RANKING FUNCTIONS WITH BOOLEAN OPERATORS

SQL: SELECT	ts_rank	ts_rank_cd
to_tsvector('t1 t2 t3'), to_tsquery('t1 | t2')	0.0607	0.2
to_tsvector('t1 t2 t3'), to_tsquery('t1 | t2 | t3')	0.0607	0.3
to_tsvector('t1 t2 t3'), to_tsquery('t1 | t2 | t4')	0.0405	0.2
to_tsvector('t1 t2 t3'), to_tsquery('t1 & t2')	0.0991	0.1
to_tsvector('t1 t2 t3'), to_tsquery('t1 & t2 & t3')	0.2683	0.1
to_tsvector('t1 t2 t3'), to_tsquery('t1 & t2 & t4')	0.0991	0

TABLE VII
POSTGRESQL SEARCH QUALITY COMPARISON

Run	MAP	P@5	P@10	sec/q
ts_rank mode 0	0.0818	0.1878	0.1755	19.0408
ts_rank mode 1	**0.1071**	**0.3102**	**0.2898**	19.5918
ts_rank mode 2	0.0441	0.1102	0.1184	19.5306
ts_rank mode 8	0.0505	0.1143	0.1265	19.0204
ts_rank mode 16	0.0987	0.2612	0.2714	19.0408
ts_rank_cd mode 0	0.0065	0.0163	0.0122	21.3673
ts_rank_cd mode 1	0.0111	0.0204	0.0163	21.6122
ts_rank_cd mode 2	0.0267	0.0735	0.0837	21.5510
ts_rank_cd mode 4	0.0465	0.0980	0.0735	21.0816
ts_rank_cd mode 8	0.0169	0.0245	0.0306	21.0612
ts_rank_cd mode 16	0.0094	0.0122	0.0143	21.1224

TABLE VIII
MEAN AVERAGE PRECISION (MAP) VALUES

	ootb	Lovins	Porter	Porter2	KStem
MySQL	0.1182	0.1152	0.1314	0.1325	0.1394
PostgreSQL	0.1071	0.0973	0.1004	0.1010	0.1094
Lucene	0.1645	0.1726	0.1931	0.1941	**0.2012**

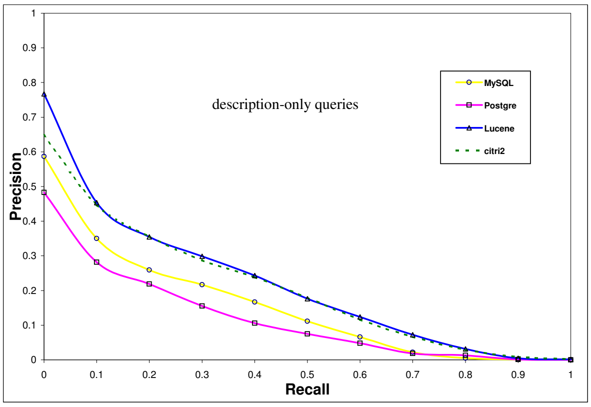


Fig. 1. Interpolated precision - recall graph

ument length normalization combination yields best results. Search quality results of PostgreSQL runs are given in Table VII. It includes six different document length normalization options and two ranking functions described in section III-D2. Note that option 4 is supported only by ts_rank_cd and option 32 is just a cosmetic change so that it is not included in our runs.

In all of runs ts_rank performed better than ts_rank_cd in terms of both search quality and search time. ts_rank_cd yielded highest MAP value with mode 4 which is implemented just for it. Among two predefined ranking functions and six length normalization options, standard ranking (ts_rank) function with the normalization option 1, yielded highest MAP value. Therefore this combination is selected as best representative of PostgreSQL.

D. Global Evaluation

In this section we compare best representative of each system. In Table VIII, the retrieval qualities (in terms of mean average precision) of each system are compared. Each system performed its best with KStemmer while overall best performing one is Lucene (with KStemmer). It is observed that among four different stemming methods, best performing is the KStemmer for the English language. Also English (porter2) stemming algorithm performed slightly better than the original porter stemming algorithm in all systems. KStemmer with Lucene performed slightly (2.8%) better than citri2 [19] which was at the seventh seat at TREC-4 automatic adhoc competition with the MAP value of 0.1956. PosgreSQL's poor performance can be explained by lack of inverse document frequency component in its ranking mechanism.

Among the several metrics eleven point precision recall curves of each system are presented in the TREC-4 overview paper [11]. Therefore the curves of each system (with KStemmer) and citri2 are plotted on Figure 1 to compare information retrieval systems visually. Lucene whose curve is the closest to the upper right-hand corner of the graph (where recall and precision are maximized) indicates the best performance.

Although main focus of this study is text retrieval quality benchmarking, as a side note, indexing times of each system are presented on Figure 2. In our experiments relational database settings are optimized for static data. Note that in PostgreSQL, GIN index is recommended[21] for static data since lookups are about three times faster than GiST, on the other hand GIN index takes about three times longer to build than GiST.

Average searching times of each system's different runs are depicted separately in **sec/q** column of tables in previous subsections. Relational Databases' searching and indexing speed are much slower than Lucene. With the response time under 50 milliseconds per query, Lucene would perfectly satisfy online web users.

V. OBSERVATIONS

Many Relational Databases have full text search functionalities but all have different syntax so there is no real standard between different vendors. Details of their inner algorithms are not well-documented.

All three systems use document length normalization in order to prevent long documents taking over. Lucene and

[21] http://www.postgresql.org/docs/8.4/static/textsearch-indexes.html

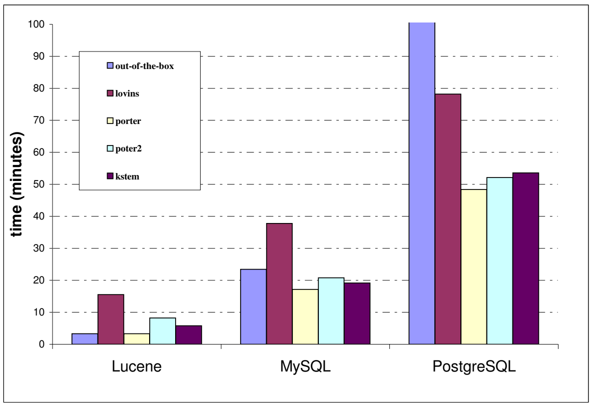


Fig. 2. Average Indexing Times

MySQL uses idf component in their score calculation while PostgreSQL does not. MySQL's full text search is limited to only MyISAM tables and has a few tuning parameters. PostgreSQL has more configurable parameters through dictionaries that provide stemming, synonym expansion, stop word removal etc. Available PosrgreSQL dictionaries are Stop Words, Simple, Synonym, Thesaurus, Ispell and Snowball. PosgreSQL's full text search, similar to Boolean mode in MySQL, does not require full text indices. In PostgreSQL, it is possible to immediately see the output of full text related functions and debug this way. For example executing SELECT to_tsvector('english', 'Testing the English configuration'); displays `'configur':4 'english':3 'test':1`

PostgreSQL and Lucene have highlighting feature that can generate snippets where query terms are highlighted. Generally users like to see which part of the document matches their queries.

VI. Conclusion

Our work in comparing the performances of relational databases and information retrieval libraries showed that; for TREC-4 adhoc collection, Lucene produced the best results in terms of efficiency and effectiveness. Lucene's out-of-the-box search quality reached to top six for TREC-4 adhoc evaluation. Although relational databases provide easy to use full text search capabilities that do not require an additional system installation and maintenance, without linguistic preprocessing their search quality is quite low. Due to the impractical response and indexing times, open source relational databases are unsuitable to be installed as a full text search solution for high traffic web applications.

References

[1] C. Middleton and R. Baeza-yates, "A comparison of open source search engines," Dec. 03 2008. [Online]. Available: http://wrg.upf.edu/WRG/dctos/Middleton-Baeza.pdf

[2] A. Arslan and O. Yilmazel, "A comparison of relational databases and information retrieval libraries on turkish text retrieval," in *Natural Language Processing and Knowledge Engineering, 2008. NLP-KE '08. International Conference on*, 19-22 2008, pp. 1 –8.

[3] K. Mahesh, J. Kud, and P. Dixon, "Oracle at trec8: A lexical approach," in *Text REtrieval Conference (TREC) TREC-8 Proceedings*. Department of Commerce, National Institute of Standards and Technology, 1999, pp. 207–??, nIST Special Publication 500-246: The Eighth Text REtrieval Conference (TREC 8). [Online]. Available: http://trec.nist.gov/pubs/trec8/papers/orcl99man.pdf

[4] S. Alpha, P. Dixon, C. Liao, and C. Yang, "Oracle at TREC 10: Filtering and question-answering," in *Text REtrieval Conference (TREC) TREC 2001 Proceedings*. Department of Commerce, National Institute of Standards and Technology, 2001, pp. 423–??, nIST Special Publication 500-250: The Tenth Text REtrieval Conference (TREC 2001). [Online]. Available: http://trec.nist.gov/pubs/trec10/papers/orcltrec10.pdf

[5] A. Maier and D. E. Simmen, "DB2 optimization in support of full text search," *IEEE Data Eng. Bull*, vol. 24, no. 4, pp. 3–6, 2001. [Online]. Available: http://sites.computer.org/debull/A01DEC-CD.pdf

[6] J. R. Hamilton and T. K. Nayak, "Microsoft SQL server full-text search," *IEEE Data Eng. Bull*, vol. 24, no. 4, pp. 7–10, 2001. [Online]. Available: http://sites.computer.org/debull/A01DEC-CD.pdf

[7] P. Dixon, "Basics of oracle text retrieval," *IEEE Data Eng. Bull*, vol. 24, no. 4, pp. 11–14, 2001. [Online]. Available: http://sites.computer.org/debull/A01DEC-CD.pdf

[8] M. Krellenstein, "Search engine versus dbms," Lucid Imagination. [Online]. Available: http://www.lucidimagination.com/Community/Hear-from-the-Experts/Articles/Search-Engine-versus-DBMS

[9] D. Smiley, "Text search, your database or solr," Packt Publishing, Nov. 2009. [Online]. Available: http://www.packtpub.com/article/text-search-your-database-or-solr

[10] Y. Jing, C. Zhang, and X. Wang, "An empirical study on performance comparison of lucene and relational database," in *Communication Software and Networks, 2009. ICCSN '09. International Conference on*, 27-28 2009, pp. 336 –340.

[11] D. Harman, "Overview of the fourth text retrieval conf. (TREC-4)," *Proceedings of the Fourth Text REtrieval Conference (TREC-4)*, 1996. [Online]. Available: http://trec.nist.gov/pubs/trec4/overview.ps.gz

[12] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[13] J. B. Lovins, "Development of a stemming algorithm," *Mechanical Translation and Computational Linguistics*, vol. 11, no. 1-2, pp. 22–31, 1968.

[14] R. Krovetz, "Viewing morphology as an inference process," in *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 1993, pp. 191–202. [Online]. Available: http://ciir.cs.umass.edu/pubfiles/ir-35.pdf

[15] O. G. Erik Hatcher and M. McCandless, *Lucene In Action*, 2nd ed. Manning Publications, 2010.

[16] *MySQL Internals Manual*, MySQL AB, Inc. [Online]. Available: http://forge.mysql.com/wiki/MySQL_Internals_Algorithms\#Full-text_Search

[17] S. Golubchik, "Mysql fulltext search," MySQL AB, Nov. 2004. [Online]. Available: http://forge.mysql.com/w/images/c/c5/Fulltext.pdf

[18] C. L. A. Clarke, G. V. Cormack, and E. A. Tudhope, "Relevance ranking for one to three term queries," *Inf. Process. Manage.*, vol. 36, no. 2, pp. 291–311, 2000.

[19] R. Wilkinson, J. Zobel, and R. Sacks-Davis, "Similarity measures for short queries," in *Text REtrieval Conference (TREC) TREC-4 Proceedings*. Department of Commerce, National Institute of Standards and Technology, 1995, pp. 277–285, nIST Special Publication 500-236: The Fourth Text REtrieval Conference (TREC-4). [Online]. Available: http://trec.nist.gov/pubs/trec4/papers/citri.ps.gz

[20] C. Buckley, A. Singhal, M. Mitra, and G. Salton, "New retrieval approaches using SMART: TREC 4," in *NIST Special Publication 500-236: The Fourth Text REtrieval Conference (TREC-4)*, D. Harman, Ed. Department of Commerce, National Institute of Standards and Technology, Nov. 1995. [Online]. Available: http://trec.nist.gov/pubs/trec4/papers/Cornell_trec4.ps.gz

ISBN 978-83-60810-27-9
ISSN 1896-7094

Parallel, Massive Processing in SuperMatrix—a General Tool for Distributional Semantic Analysis of Corpus

Bartosz Broda, Damian Jaworski, Maciej Piasecki
Institute of Informatics, Wrocław University of Technology, Poland
Email: {bartosz.broda, maciej.piasecki}@pwr.wroc.pl

***Abstract*—The paper presents an extended version of the SuperMatrix system — a general tool supporting automatic acquisition of lexical semantic relations from corpora. Extensions focus mainly on parallel processing of massive amounts of data. The construction of the system is discussed. Three distributed parts of the system are presented, i.e., distributed construction of co-incidence matrices from corpora, computation of similarity matrix and parallel solving of synonymy tests. An evaluation of a proposed approach to parallel processing is shown. Parallelization of similarity matrix computation demonstrates almost linear speedup. The smallest improvements were achieved for construction of matrices, as this process is mostly bound by reading huge amounts of data. Also, a few areas in which functionality of SuperMatrix was improved are described.**

I. Introduction

HUGE text corpora are now relatively easy to be collected, e.g., on the basis of the Web. They can be a valuable, direct source of linguistic data, e.g., use examples, but also a basis for the automated extraction of linguistic knowledge. Such applications as automatic induction of morphological descriptions or extraction of syntactic subcategorisation frames are well known. However, recently we can also observe applications of automated extraction methods in the lexical semantics, too. Construction of a large scale language resource describing lexical semantics, e.g. a wordnet[1], is a very laborious process, in which skilled lexicographers must be involved. Such a process can be supported by the automatic extraction of semantic relations among words, or multi-word units in general, as well, as semantic restrictions imposed by particular words on their occurrence contexts. The extracted knowledge can be consulted by linguists during the construction of a resource or can be used for suggesting to the lexicographers some new elements of a semantic lexicon, e.g., lexical semantic relations between lexical units or groups of lexical units (i.e., wordnet synsets) in a wordnet, cf [2].

There are two main paradigms of automatic extraction of instances of lexical semantic relations, e.g., [3]:

- *pattern-based*,
- and *clustering-based*.

The clustering-based paradigm is also called *distributional paradigm*, as it originates directly from the *Distributional Hypothesis* formulated by Harris [4].

According to the pattern based approaches there are some lexico-syntactic patterns, which combine two words and mark the word pair as an instance of some lexical semantic relation, e.g., hypernymy [5]. So every single occurrence of a precise pattern is treated as evidence for the semantic association of the given words. The pattern-based methods can utilise individual occurrences of word pairs in a corpus, but are error prone in the same time, as they depend on individual occurrences. This drawback can be corrected by taking into account statistics of word co-occurrences with different patterns across the whole corpus, cf Espresso [3] and Estratto [6] methods.

Statistical analysis of co-occurrences can lead to the automatic construction of a Measure of Semantic Relatedness (MSR) (also called Measure of Semantic Similarity) which is modelled as a function:

$$MSR : W \times W \to R \tag{1}$$

where W is a set of words and R is the set of real numbers.

MSR construction is based on the analysis of the similarity of distributions of some words across different lexico-syntactic or even semantic contexts as evidence for their close semantic relation.

There are plenty of methods of the automatic extraction of MSRs. Most of them start with processing a corpus and constructing a co-incidence matrix describing co-occurrences of LUs (rows) and lexico-syntactic contexts (columns). They differ in three aspects: definitions of contexts, transformations of the raw frequencies and calculation of the final measure value. A context can be a whole document, but much better results can be obtained using a more precise description of the context. So, a word occurrence is mostly described by other words occurring in its context and, possibly, syntactic relations linking them to it, e.g., a particular noun can be characterised by adjectives such that they occur in its context and syntactically modify it. A *feature* is a particular type of co-occurrence describing the context, and a lexico-syntactic feature is a pair: a word and a syntactic relation, e.g., ⟨*fast*, `modifier`⟩. The initial value of the features are the frequencies of their co-occurrence with the words that are described. However, these raw values are mostly transformed later in order to emphasise the semantic information they express in relation to words being described.

[1] By wordnet we mean here an electronic thesaurus of a structure following the main lines of the Princeton WordNet thesaurus [1].

978-83-60810-27-9/09/$25.00 © 2010 IEEE

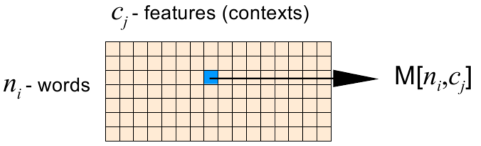


Fig. 1. Schematic view of co-occurrence matrix.

Several algorithms for collecting corpus frequencies, transforming feature value and calculating semantic relatedness of words on the basis of their feature vectors were implemented in the *SuperMatrix* system [7] – a general tool for the extraction of lexical semantic knowledge on the basis of distributional analysis of corpora. The increasing size of corpora processed by SuperMatrix and the resulting huge amount of the extracted data resulted in very long processing time in practical applications of SuperMatrix. From the very beginning a typical practice was to divide manually huge processing tasks into several parts processed independently: several copies of SuperMatrix were run in parallel, on different computers, and next the partial results were merged using manually controlled functions of SuperMatrix. The general requirement for the computing time is unavoidable, if one wants to work with huge corpora, but the total time of task completion can be reduced by extending SuperMatrix with elements of the parallel architecture.

The goal of the paper is to present a version of SuperMatrix, which is extended with elements of the parallel processing architecture, that is aimed towards better support for efficient distributional analysis of massive corpora. Moreover, several extensions and improvements concerning extraction and clustering algorithms of SuperMatrix are also presented.

In the following sections, first we will briefly describe the SuperMatrix system as a tool supporting semantic analysis, i.e., its basic algorithms and functionality. Next we will identify these algorithms that can be successfully transformed into their parallel versions. We will present the distributed, parallel architecture of the new version of SuperMatrix. Finally, we will report on the experiments performed, practical applications and plans for further extensions.

II. SuperMatrix

SuperMatrix is a collection of libraries for programmers and end-user tools for creating, storing and manipulating co-incidence matrices that describe distributional patterns of words. A co-incidence matrix is a matrix, in which rows correspond to words being described and columns to features, see Fig. 1. Initial values of cells are set to frequencies with which features occurred in the contexts of the subsequent words. The system consists of the four main modules,[2] namely:

- *Matrices* – a library for storing and accessing matrices. SuperMatrix supports a few formats of storing matrices, but the most important one are sparse matrices.
- *Comparator* – a library enabling computation of similarity between rows of matrices (i.e., between one word and multi-word lexical units) using different MSRs. Comparator was also designed in a modular manner. The computation of a similarity matrix was decomposed into a few steps, i.e., global filtering of features, local selection of features, weighting of features and comparing feature vectors. For every stage there are several algorithms implemented in SuperMatrix.
- *Matrix tools*, including (but not limited to) tools for: constructing co-incidence matrices (collecting raw frequencies and transforming feature values), MSRs, joining matrices and analysing matrix content, e.g., manually browsing selected rows and columns from any matrix – including transformed and weighted matrices.
- Clustering – a package consisting of several clustering algorithms, e.g., implementation of Clustering by Committee [8] (modified and extended [9]), RObust Clustering using linKs [10] and Growing Hierarchical Self-Organising Maps [11], and also an interface to the CLUTO package [12].

III. Distributed SuperMatrix

There are two main problems of processing huge corpora: long processing time and huge amount of memory required. Fortunately, corpus processing has in general a parallel nature and the majority of matrix operations can be easily decomposed into operations performed on matrix parts. In the following subsection we will discuss parallel algorithms introduced into extended SuperMatrix, and in the next subsection we will present the contemporary, improved functionality of the system.

A. Parallel Algorithms

We have identified three main areas of the SuperMatrix functionality, in which parallel algorithms can introduce improvements:

- co-incidence matrix construction,
- similarity matrix calculation,
- MSR evaluation by WordNet-based Synonymy Test (WBST)[3] [16].

Corpus processing is the fourth possible area, which is suitable for parallel processing, but SuperMatrix is mostly applied to text previously tagged morphosyntactically and text processing is reduced to the application of morphosyntactic constraints or reading annotation – both these elements are included in the matrix construction.

A natural way of task distribution among processing nodes during matrix construction seemed to be assignment of the subsequent parts of the corpus to the different nodes. However, this solution encounters several problems. First, the applied binary corpus format of Poliqarp is best suited for sequential corpus processing in our wrapping library. Secondly, definition of the matrix columns in the form of lexicalised morpho-syntactic constraints can consume huge amount of operating

[2] We present a brief description of a system, for detailed discussion see [7].

[3] The task in WBST test is to determine which word among a few choices given is synonymous to the given question word. There is a long tradition of evaluation of MSRs using synonymy tests, e.g., [13]–[15].

memory, as the constraints are kept in the compiled form due to efficiency reasons. Thus, a large constraint-based matrix of around 500 thousands of columns is too large to be kept in the memory[4], but we face the necessity to build such a matrix as a tool in the analysis of a huge Polish corpus of 930 million of words. Construction of a matrix including features based on the morpho-syntactic constraints requires huge amounts of memory, but it is also the most valuable type of a matrix in terms of the accuracy of the results generated and possible applications of SuperMatrix.

Instead of distributing data across processing nodes, we distribute processing. Each processing node is assigned a matrix with a subset of the original set of columns. The sub-matrices (column slices) are defined by the specification of the features sent by the central managing node. All nodes are run on the corpus in parallel. A corpus is stored on the shared resource, as this is the usual approach in computing clusters. Also, a corpus can be huge so keeping it copied in multiply places is not practical. Nevertheless, this is a potential bottleneck of the system. However, as our experiments with corpus copies placed on the processing nodes showed, the data transfer influences the system efficiency to a minor extent. At the end, the data stored in the sub-matrices are read sequentially by the central node and merged into the resulting matrix of possible huge dimensions, but typically very sparse. Constraints are kept in memory only for processing in the nodes, columns in the resulting matrix are described by labels.

The big advantage of the above model is the ability to construct word descriptions unlimited in the number of features used distributed across processing nodes. Scheduling is also simple in the case of a limited set of nodes – the central manager can gradually sent the subsequent sub-matrices to nodes which just finished their task.

In the case of the similarity matrix calculation, the smallest unit of processing which can be distributed independently of specifics of measure of semantic relatedness is the calculation of the similarity of two rows. As a typical task is computation of the k most semantically related words to the given one, the central manager assigns identifiers of rows to be processed to the processing nodes. The central node collects the computed values or ranking lists for the subsequent words.

In the parallel implementation of WBST-based evaluation of MSR, all processing nodes store the full list of question-answer (QA) pairs, the similarity matrix and the central manager assigns subsequent <test, MSR, matrix> tuples to free nodes. Testing results (the number of correct and incorrect answers, as well, as omitted QA pairs when one of the words was not described by a given matrix) are sent to the central node which calculates the final aggregated test statistics.

B. Improved Functionality

The functionality of SuperMatrix has been improved in a few areas. Aside from bugfixes and general code clean-up we have added a few new algorithms for construction of MSR or extended existing ones. We have added simple quantization of weighted values to mutual information based measures (Mutual Information of Lin's [17] measure and Pointwise Mutual Information [8]). Several different methods of weighting inside RWF function have been implemented using contingency table based definitions proposed by [18]. We have added RWF based on: mutual information, log likelihood and χ^2 measures of associations. Rank weight functions were also extended with a possibility of building non-linear rankings of features (e.g., exponential).

Treating co-occurrence data gathered using different lexico-semantic constraints (or different grammatical relations acquired from parser) in an uniform way may be sometimes inappropriate. Thus, we have added simple *ensemble-like* MSRs that works on different parts of matrices independently. This is one of the ongoing area of research within our team.

As it was mentioned above, SuperMatrix does data clustering via either using built-in algorithms (e.g., GHSOM or CBC) or interfacing with CLUTO. Both of these ways expose pretty advanced clustering algorithms for the user. However, there was no easy way to use some very simple algorithms. Thus we have implemented two baseline clustering algorithms: k-means and k-medoids. The latter uses classical *partition around medoids* approach. The implementation of those algorithms is a little bit different then classical ones, i.e., we perform clustering on the basis of similarity (any weighting scheme in SuperMatrix can be used, but similarity is restricted to cosine in current version). In terms of clustering we have improved our integration with CLUTO, i.e., for internal algorithms we use output format of CLUTO and we added tools for evaluation of clustering based on this format. We have implemented only a few evaluation measures, namely: purity, Rand Index, normalised mutual information and precision, recall and f-measure [19]. Having this tight integration with CLUTO enables us to use also SenseCluster [20] method of evaluation, which uses a little bit different definition for computing precision and recall.

We also work on the web service base access to SuperMatrix and part of the targeted functionality is already available[5]. This work is done as a part of the CLARIN project. Currently we have most of the technical bits of SOA-based communication with SuperMatrix worked out and a user can query a few pre-build matrices for similarity between words.

We have added an interface to work with Doug Rohde's SVDLIBC library[6] — a cleaned up version of Berry's SVDPACKC [21].

Also a few technical improvements have been made in SuperMatrix. The performance of (sequential) parts of MSR computation have been greatly improved (by a factor of 2-10 depending on complexity of MSR, the more complex the higher the speedup) by allowing user to weight whole matrix, before computations (which can be undesirable in a few cases,

[4] On the processing node with 4 GB of RAM only construction of a matrix with 76,000 columns could be started, but was not completed.

[5] http://nlp.pwr.wroc.pl/clarin/ws/supermatrix/

[6] http://tedlab.mit.edu/ dr/SVDLIBC/

e.g., some MSRs use unweighted data in the later stages of processing, or raw frequencies are required in different part of user's application). We have improved a parser of MSR request in a way that every part of MSR can be now specified via a character string without resorting to assembling MSR inside the code. In previous version of SuperMatrix only predefined combinations of parameters were available in this way.

IV. Experiments

All experiments were aimed at comparing the processing time of the single-node version of SuperMatrix with the new parallel version run in the distributed network of processing nodes. For the needs of semi-automated development of plWordNet (a wordnet for Polish) [2] a corpus of the size 900 million words[7] is processed, around 350,000 lexicalised constraints (based on adjectives, nouns and verbs) were identified as delivering statistically meaningful information and the description is built for around 25,000 nouns (some of them are multiword expressions)[8]. Concerning these numbers, performing the repeated tests on the corpus of this size would consume the power of our computing cluster to the very large extent, while the cluster is being extensively used in everyday practice of the works on preparing semantic resources for plWordNet. Also, it is not possible to fit a matrix of this size in the memory which is needed for comparison with performance of a run on a single node – only up to 40–60 thousands of column constraints, depending on the complexity of the constraints, can be stored in 4GB RAM. Thus all experiments were performed on smaller matrices or on partially completed tasks.

All experiments were performed on the computing cluster consisting of 6 computing nodes. Each node is equipped with two dual core processors Intel(R) Xeon(R) CPU 5160 3GHz. 4 processes can be run in parallel on each node that gives maximally 24 virtual processing nodes. Three of the nodes have 5 GB RAM, three 6 GB RAM – this is a strong limitation for SuperMatrix, concerning a single processing node. The nodes are connected by the InfiniBand network of 96Gbit/s bandwidth. The cluster runs under CentOs 5.3. All parallel algorithms of the extended SuperMatrix were written in C++ and the inter-process communication using Message Passing Interface (MPI) library.

Aside from time of execution we mention also *speedup* $S(p) = \frac{T_s}{T_p(p)}$, where p is a number of processors, Ts is a time of execution of best sequential algorithm solving the problem and $T_p(p)$ is time of execution of parallel algorithm using p processors. In practice $T_s = T_p(1)$ for convenience [24].

TABLE I
Matrix construction on the different number of processing nodes.

No of nodes	1	2	4	6	8	10	12	14
Time (min)	90.5	71.5	64.25	62	61.75	54.5	51.5	53
Speedup		1.27	1.41	1.46	1.47	1.66	1.76	1.71

A. Co-incidence Matrix Construction

A matrix was constructed on the basis of the IPI PAN Corpus (around 250 millions of words). Description of 25,000 nouns was based on constraints based only on adjectives that resulted in around 35,000 lexical constraints (columns). A constraint of this type recognises occurrence of the given adjective in text as modifying the given noun which is being described.

During the first experiment constraints were uniformly assigned to the processing nodes. The results for different number of nodes used are presented in Tab. I. The processing time decreases with the increasing number of nodes. However, the process saturates with 10 processing nodes, and starting with 12 nodes the speed of processing starts to increase. It is caused by the communication overhead: adding new nodes brings a relatively small increase of processing speed but increases the complexity of communication. The reason for this is that a significant part of the process has a sequential character, i.e., mainly reading the corpus.

In the second experiment, we compared the behaviour of the system in the environment of the limited memory capacity. The maximal number of the constraints stored was limited to 7800 only. A sequential non-parallel version was run 4 times and was compared with the parallel version run on the 4 nodes simultaneously. The results are, respectively: 169.5 minutes and 64.25 minutes. The total cost of processing 257 min. (4 times 64.25 min.) is much higher in the case of the parallel version, however we get the result 2.64 times faster.

B. Similarity Matrix Calculation

Experiments in this subsection were based on the co-incidence matrix of 25,000 rows (describing nouns) and 240,000 columns representing lexicalised constraints. Cosine measure was applied as row similarity measure, so no transformation took place during the similarity calculation.

In the first experiment, computations was repeated for several different numbers of words, in each case the goal was to find $k = 20$ most semantically related words to the given one. Experiment was repeated for the different numbers of processing nodes used. As complete would require a lot of processing time, the final results were estimated on the basis of the first hour of processing in each case. The results are presented in Tab. II.

The achieved speedup is almost linear – with the increasing number of nodes the speedup of the processing is increasing (see Fig. 2). The sequential part is performed on each node and encompasses reading the co-incidence matrix (from the shared hard drive) and calculating the similarity function. So

[7] A joint corpus consisting of 10 smaller, including the IPI PAN Corpus [22], the corpus of *Rzeczpospolita* [23] and several corpora built on the basis of the Web content.

[8] The set includes words already included in plWordNet and new ones, as the aim of the process is to extract from the corpus information concerning semantic associations among words and next use this information in generating suggestions concerning the placement of the new words in the plWordNet structure.

TABLE II
TIME (MIN) REQUIRED FOR THE SIMILARITY MATRIX CALCULATION ON THE DIFFERENT NUMBER OF PROCESSING NODE – "ND." AND FOR DIFFERENT SIZE OF THE PROBLEM.

	No of words						
Nd.	200	500	1000	2000	5000	10000	25000
1	107	267	476	1120.5	2719	4486.5	14760
2	57.25	138	302	553.75	1323.5	3102	6402
4	34.25	74	154.5	306.5	727	1488.25	3682
6	23	45	93.5	180	453.5	911	2318
8	20	37.5	78.75	152.5	387	759.5	1987
10	16.75	28.5	60	115.25	288	581	1408.5
12	13.75	23	47.5	88	225.5	445	1104.5
14	13.25	19.5	40.25	75	186.5	373.25	927
20	12.5	16.75	33.25	70.25	157	307	781

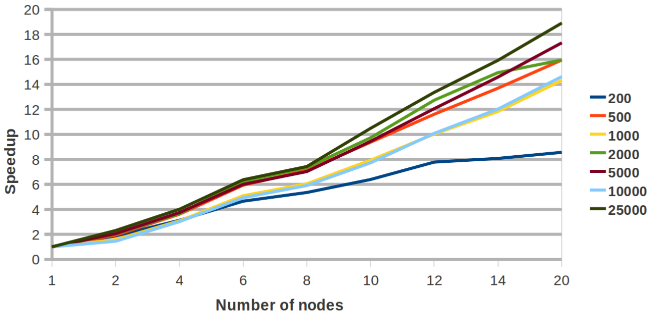


Fig. 2. Speedup of similarity matrix calculation in relation for different size of the problem

the parallel part of the process increases with the increasing number of words.

A typical task is calculation of a list of the k most semantically related words to each described word. Such a list is used next, e.g., as a knowledge source in a *WordNet Weaver* system that supports extension of a Polish wordnet in a semiautomatic way.

During the second experiment, we tested the influence on the speedup of the number of k most semantically related words calculated for each word and sent to the central manager. Our aim was to analyse consequence of the expected increasing communication complexity with the increasing length of lists returned to the central manager. The results for the two values of k: 20 – a typical one, and 25,000 – the full list, are presented in Tab. III. The lists were computed for a sample of 500 words (nouns).

In the case of 1 and 2 nodes, the processing time was estimated on the basis of part of the process. If size of the results had had an influence on the communication, the time required for computing a list for $k = 25000$ would have been larger than for $k20$, but it is much smaller in fact! It is caused by the lack of sorting in the case the whole list of 25,000 similarity values is returned. According to the assumed algorithm, if k is smaller than the number of words a list must be sorted and only the k top are returned.

Both experiments showed that with the increasing complexity of the problem, but also with increasing number of processing time the system expresses very positive efficiency.

TABLE IV
TIME (MIN) USED REQUIRED FOR PERFORMING WBST TESTS FOR DIFFERENT CO-INCIDENCE MATRICES AND MEASURES OF SEMANTIC RELATEDNESS. THE VALUE OF SPEEDUP IS GIVEN IN THE BRACKETS.

MSRs	5		10	
Matrices	1	2	1	2
Nodes				
1	21	51	66.5	157.5
2	13 (1.61)	26 (1.96)	35 (1.9)	82.5 (1.91)
3	10.2 (2.05)	22 (2.32)	28 (2.37)	65 (2.42)
4	10 (2.1)	19.5 (2.61)	24.2 (2.74)	55.5 (2.84)
5	6 (3.5)	15 (3.4)	18 (3.69)	42.2 (3.73)
8	6 (3.5)	11.2 (4.53)	16.2 (4.09)	28.5 (5.53)
10	6 (3.5)	8 (6.35)	16.2 (4.09)	22.5 (7)

These results have a very practical meaning, as row similarity matrices are comptuted very often in practice, and in the case of a non-parallel version of SuperMatrix their calculation was a serious bottleneck for more complex process of the extraction of lexico-semantic relations.

C. *Measure of Semantic Relatedness Evaluation*

For the needs of the experiment a WBST test was randomly generated from plWordNet. The test consisted of around 15,000 of question-answers (QA) pairs. Two large co-incidence matrices were collected. For the experiment we selected also 10 MSRs based on different algorithms of different complexity. All MSRs were calculated during experiment for the pairs of words from the test. Matrices were stored on the processing nodes.

Two versions of the WBST-based evaluation were compared. The old one, which is parallel (thread-based parallelism) but not distributed. It can assign different <test, MSR, matrix> tuples to different cores of the single machine. The second one of extended SuperMatrix is capable of distributing subsets of <test, MSR, matrix> tuples across processing nodes.

The parallel version was tested for 5 and 10 different MSRs to be tested and 1 and 2 co-incidence matrices on up to 10 processing nodes. The old non-distributed version was tested for the same combinations of MSRs and matrices but only up to 4 cores – the maximal number in the PC used for the test. The results of the experiment are presented in Tab. IV.

We expected that the WBST performance would be increasing with the increasing number of processing but the speed will not rise linearly in relation to the size of the problem. In our expectations we took into account different complexity of the used MSR algorithm and different properties of the two co-incidence matrices applied. WBST parts, i.e., subsets of QA pairs are different from the computation point of view – they differ only in their size. In order to solve one QA pair, the system has to compare four row vectors. The achieved results are generally compatible with our initial expectations. The only difference is that in the case of two matrices processed the speedup grows almost linearly, while in the case of one matrix saturates around 5 nodes.

TABLE III
TIME (MIN) REQUIRED FOR COMPUTING LIST OF THE k MOST SEMANTICALLY RELATED WORDS TO THE GIVEN ONE.

	Number of Nodes								
	1	2	4	6	8	10	12	14	20
k=20	267	138	74	45	37.5	28.5	23	19.5	16.75
k=25000	215.5	117.25	75.5	45.25	37.75	29.25	23.25	19.75	16.75

V. RELATED WORKS

There are a few systems that are similar in functionality to SuperMatrix, but process data in a sequential manner. Reimplementation of LSA [13] is possible with the combination of *MC Toolkit* [25] and SVDPACKC [21]. There are a few problem with this approach, e.g., MC Toolkit supported only ASCII encoding and could only create a words by documents matrix. *Infomap NLP* package [26] supports similar functionality to MC Toolkit with SVD, but it was especially created for the extraction of word meanings from free text corpora. Infomap NLP package has been abandoned in favour of a new system called *Semantic Vectors* (SV) [27]. Main differences with Infomap NLP are: usage of random projection instead of SVD for dimensionality reduction and the implementation, which was written completely in Java, using Apache Lucene as a document-indexing engine.

Distribution of computation in natural language processing has a long tradition in domain of Information Retrieval (IR). However, in IR the main approaches focuses on reducing volumes of fetching, sending and processing of (inverted) index, cf [28], [29].

Pantel et al. present work similar to subset of ours — scaling similarity calculation to very large datasets [30]. However, the focus in [30] was put on distribution of only the similarity function, like cosine or Dice. In our approach we parallelized all the computations (matrix construction, feature selection and filtering, weighting and similarity calculation). Pantel et al. use a approach based on MapReduce framework which is popular in many application in NLP, e.g., clustering [31].

VI. CONCLUSIONS

SuperMatrix appeared to be a very valuable system during the semi-automated construction of plWordNet, see the detailed report on techniques applied in [2]. However, the growing size of both: corpus and plWordNet, caused that the further usage of a single-machine version of SuperMatrix started to be difficult. Work on transforming the system into a system based on the parallel architecture have been initiated.

In SuperMatrix, the basic data structure is a matrix, the majority of operations have a sequential character in relation to processing rows or columns of the matrix. Thus, the main concern was in which areas the effort on introducing parallel and distributed processing will be most effective in terms of the reduced time of processing or the increased scale of problems processed. The latter was even a more severe limitation than the former.

Three main areas were identified for transformation into parallel computation, namely: co-incidence matrix construction, similarity matrix calculation and MSR evaluation by WBST. The first two are the most costly processes in terms of computation complexity and both were bottlenecks for larger problems: the former because of the memory usage, and the latter due to time complexity. Less attention was given so far to the feature-oriented transformations of the matrix, as they are applied to the already collected frequencies and sparse matrices. Matrix-oriented transformations, like SVD, form a separate class of problems, SuperMatrix depends in this area on the third-party solutions, so this important issue was left for further development. Corpus preprocessing, e.g., by a morpho-syntactic tagger, is also an area suitable for parallel processing. However there is an ongoing work on some solutions in this phase in another project, cf. [32], so it was not solved here – SuperMatrix is able to co-operate with some language tools, e.g., the TaKIPI tagger, but from the point of view of parallel SuperMatrix architecture we assume that the input corpus is already pre-processed. Moreover, the morpho-syntactic constraints, which are heavily applied during matrix construction are in fact a language tool similar to a chunker (a kind of a parser).

All introduced solutions appeared to be effective in performed tests and in practice. In the case of the matrix construction the decrease of processing time saturates with the increasing number of processors, but even more important profit is that parallel architecture enables construction of matrices of practically unlimited size in a convenient way, i.e., without human guiding the system through the process. This is important feature, because the construction of large amount of submatrices independently and summing them in correct order is long, laborious and error prone process when done manually. The use of 500,000 features and even more is justified as distribution of words in the natural language is very sparse and different subsets of lexicon can be described by different subsets of lexico-syntactic features, e.g., by different specific adjectives. The present version of SuperMatrix is prepared for this.

The applied technical solution based on the MPI, makes possible to use the system not only on a specialised computing cluster by also on a network of computers or even utilise multi-core architecture of a modern PC.

We plan to make SuperMatrix more flexible with respect to the application to different languages and different corpus annotation schemes. The present version works for Polish and English, but some elements are tightened to some specific language tools.

Possible application areas for SuperMatrix include extraction of MSRs from corpora and semantic correction of handwritten text. This system can be also used to perform unsupervised word sense disambiguation, named entity disambiguation, sentiment analysis, document indexing, clustering and retrieval and search engine construction. Last but not least, we suspect that SuperMatrix can be used outside the domain of natural language processing, i.e., everywhere were an object can be represented as a sparse feature vector (especially in high dimensional space).

Acknowledgment

Work financed by Innovative Economy Programme project POIG.01.01.02-14-013/09. Work supported by the DCS-Lab, which is operated by the Department of Distributed Computer Systems (DDCS) at the Institute of Informatics, Wroclaw University of Technology.

References

[1] C. Fellbaum, Ed., *WordNet — An Electronic Lexical Database*. The MIT Press, 1998.

[2] M. Piasecki, S. Szpakowicz, and B. Broda, *A Wordnet from the Ground Up*. Oficyna wydawnicza Politechniki Wroclawskiej, 2009.

[3] P. Pantel and M. Pennacchiotti, "Espresso: Leveraging generic patterns for automatically harvesting semantic relations." ACL, 2006, pp. 113–120. [Online]. Available: http://www.aclweb.org/anthology/P/P06/P06-1015

[4] Z. S. Harris, *Mathematical Structures of Language*. New York: Interscience Publishers, 1968.

[5] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora." in *Proceeedings of COLING-92*. Nantes, France: The Association for Computer Linguistics, 1992, pp. 539–545.

[6] R. Kurc and M. Piasecki, "Automatic Acquisition of Wordnet Relations by the Morpho-Syntactic Patterns Extracted from the Corpora in Polish," in *Proceedings of the International Multiconference on Computer Science and Information Technology – Third International Symposium Advances in Artificial Intelligence and Applications*, 2008, pp. 181–188.

[7] B. Broda and M. Piasecki, "SuperMatrix: a General tool for lexical semantic knowledge acquisition," *Speech and Language Technology*, vol. 11, pp. 239–254, 2008.

[8] P. Pantel, "Clustering by committee," Ph.D. dissertation, Edmonton, Alta., Canada, Canada, 2003, adviser-Dekang Lin.

[9] B. Broda and M. Piasecki, "Experiments in documents clustering for the automatic acquisition of lexical semantic networks for polish," in *Proceedings of the 16th International Conference Intelligent Information Systems*, 2008, to appear.

[10] S. Guha, R. Rastogi, and K. Shim, "Rock: A robust clustering algorithm for categorical attributes," *Information Systems*, vol. 25, no. 5, pp. 345–366, 2000. [Online]. Available: citeseer.ist.psu.edu/guha00rock.html

[11] A. Rauber, D. Merkl, and M. Dittenbach, "The growing hierarchical self-organizing maps: exploratory analysis of high-dimensional data," 2002.

[12] G. Karypis, "CLUTO - a clustering toolkit," Tech. Rep. #02-017, nov 2003.

[13] T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition," *Psychological Review*, vol. 104, no. 2, pp. 211–240, 1997.

[14] P. D. Turney, "Mining the Web for synonyms: PMI-IR versus LSA on TOEFL," in *Proceedings of the Twelfth European Conference on Machine Learning*. Berlin: Springer-Verlag, 2001, pp. 491–502.

[15] M. Piasecki, S. Szpakowicz, and B. Broda, "Extended similarity test for the evaluation of semantic similarity functions," in *Proceedings of the 3rd Language and Technology Conference, October 5–7, 2007, Poznań, Poland*, Z. Vetulani, Ed. Poznań: Wydawnictwo Poznańskie Sp. z o.o., 2007, pp. 104–108.

[16] D. Freitag, M. Blume, J. Byrnes, E. Chow, S. Kapadia, R. Rohwer, and Z. Wang, "New experiments in distributional representations of synonymy." in *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*. Ann Arbor, Michigan: Association for Computational Linguistics, June 2005, pp. 25–32.

[17] D. Lin, "Automatic retrieval and clustering of similar words," in *COLING 1998*. ACL, 1998, pp. 768–774. [Online]. Available: http://acl.ldc.upenn.edu/P/P98/P98-2127.pdf

[18] S. Evert, "The statistics of word cooccurrences: word pairs and collocations," *Unpublished doctoral dissertation, Institut f "ur maschinelle Sprachverarbeitung, Universit "at Stuttgart*, 2004.

[19] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[20] A. Purandare and T. Pedersen, "Senseclusters - finding clusters that represent word senses," in *HLT-NAACL 2004: Demonstration Papers*, D. M. Susan Dumais and S. Roukos, Eds. Boston, Massachusetts, USA: Association for Computational Linguistics, May 2 - May 7 2004, pp. 26–29.

[21] M. Berry, "Large scale singular value computations." *International Journal of Supercomputer Applications*, vol. 6, no. 1, pp. 13–49, 1992.

[22] A. Przepiórkowski, *The IPI PAN Corpus: Preliminary version*. Warsaw: Institute of Computer Science, Polish Academy of Sciences, 2004.

[23] "Korpus rzeczpospolitej," [on-line] http://www.cs.put.poznan.pl/dweiss/rzeczpospolita.

[24] B. Wilkinson and M. Allen, *Parallel programming: techniques and applications using networked workstations and parallel computers*. Prentice Hall, 1999.

[25] I. S. Dhillon and D. S. Modha, "Concept decompositions for large sparse text data using clustering," *Machine Learning*, vol. 42, no. 1, pp. 143–175, Jan 2001.

[26] D. Widdows, "Unsupervised methods for developing taxonomies by combining syntactic and statistical information," in *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 197–204.

[27] D. Widdows and K. Ferraro, "Semantic vectors: a scalable open source package and online technology management application," in *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, E. L. R. A. (ELRA), Ed., Marrakech, Morocco, may 2008.

[28] A. Moffat, W. Webber, J. Zobel, and R. Baeza-Yates, "A pipelined architecture for distributed text query evaluation," *Information Retrieval*, vol. 10, no. 3, pp. 205–231, 2007.

[29] F. Silvestri, S. Orlando, and R. Perego, "WINGS: a parallel indexer for Web contents," *Computational Science-ICCS 2004*, pp. 263–270, 2004.

[30] P. Pantel, E. Crestan, A. Borkovsky, A. Popescu, and V. Vyas, "Web-scale distributional similarity and entity set expansion," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*. Association for Computational Linguistics, 2009, pp. 938–947.

[31] J. Uszkoreit and T. Brants, "Distributed word clustering for large scale class-based language modeling in machine translation," *Proceedings of ACL-08: HLT*, pp. 755–762, 2008.

[32] B. Broda, M. Marcińczuk, and M. Piasecki, "Building a node of the accessible language technology infrastructure," in *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odjik, S. Piperidis, M. Rosner, and D. Tapias, Eds. Valletta, Malta: European Language Resources Association (ELRA), may 2010.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 381–388

ISBN 978-83-60810-27-9
ISSN 1896-7094

Development of a Voice Control Interface for Navigating Robots and Evaluation in Outdoor Environments

Ravi Coote
Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE
Neuenahrer Str. 20, 53343 Wachtberg, Germany
ravi.coote@fkie.fraunhofer.de

***Abstract*—In this paper the development of a prototypic mobile voice control for navigating autonomous robots within a multi robot system is described. As basis for the voice control a hidden markov model based speech recognizer with a very low vocabulary of 30 words is utilized. It is investigated how many training samples for a markov model are required for a normal operation of speaker-dependent speech recognition. Therefore, hidden markov models were developed successively in parallel with an own training data corpus containing finally 2290 utterances from 12 speakers. Within the successive development of acoustical models and training corpus, the work revealed details about how many speakers are necessary to achieve an acceptable degree of speaker independence. We focused on an evaluation of the speech recognizer in adverse outdoor environments. The evaluation ranges from almost calm conditions of about 39 dB up to very adverse noise conditions of 120 dB. It is investigated whether a small vocabulary attenuates the noise vulnerability and in how far an increase of speaking volume can compensate noises of different intensity. The voice control was tested in outdoor environments and aspects of its usage are described.**

I. Introduction

IN HUMAN-MACHINE scenarios, e.g., where the user does not have his hands free to type in commands, or where the user is handicapped, the ability to control a system by voice can be considered. For those purposes usually small vocabularies are sufficient. In calm acoustical environments, e.g., in flats, low vocabulary speech recognition performs almost perfectly. Unfortunately in outside-scenarios or in in-vehicle-scenarios the acoustical environment can be very adverse.

It is not clear in how far a small vocabulary can compensate such bad acoustical conditions in order to maintain an acceptable word recognition rate (WRR) of 95%. Furthermore, it is an open question in which noise scenarios an increase of the speaking volume can maintain this recognition rate.

To this end, a low vocabulary speech recognizer was developed and evaluated under several adverse conditions. The evaluation ranged from almost calm conditions of about 39 dB up to very adverse noise conditions of 120 dB. For the training of models we constructed an acoustic corpus containing 2290 hand labeled German utterances from 12 people with different accents and relevant issues in corpus construction are described. For the unit of speech that has to be acoustically modeled by hidden markov models (HMMs), the word was chosen. Suitable numbers of gaussian mixture components were specified for speaker-dependent and speaker-independent training. The successive development of the acoustic models revealed insights in how many training samples are required per model and how many speakers are needed for speaker independent speech recognition. The core speech recognizer was connected to the software framework of the robots by means of suitable software libraries as will be explained in Section II. Finally, the speech recognizer was integrated into a voice control application for navigating robots within a multi robot system and issues in operating the voice control in outside environments are described.

A. Related work and Goals

In various studies experimental speech communication with robots has already been developed and successfully used (see, e.g., [1], [2]). However, in these works no studies were conducted regarding the performance of speech recognition when used in different noise scenarios. Therefore, we developed the voice control application with the aim to give answers to the following questions:

1) How strong is the effect of street noise, crowd noise, and in-vehicle noise of different degrees on the performance of speech recognition? Can an increase of speaking volume improve recognition rate and can a vocabulary with less than 50 words compensate such noise?
2) How many speakers are necessary to achieve an acceptable level of small-vocabulary speaker independence?
3) Does direct voice input suit to perform spatial navigation tasks?

The rest of the paper is structured as follows. Next, Section II describes groundwork and conceptual considerations for the voice control. The successive development of the application is illustrated in Section III. The speech recognizers performance is evaluated in Section IV and Section V concludes with a discussion.

II. Conception

This section describes basic conditions and conceptual considerations on which basis the voice control was developed.

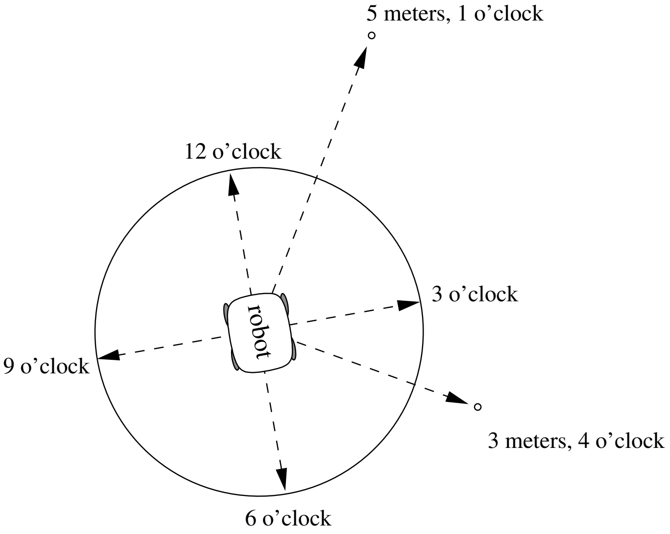


Fig. 1. Specifying target coordinates in a robot-centered 2-dimensional area

A. Navigation by the use of voice

The goal was to enable a navigation of robots onto a two-dimensional arbitrary ground, i.e., to move the robots backwards and forwards, and to rotate and stop. To allow for such a control, one single command was defined to consist of the specification of a direction and a distance. The values of distance and direction are given in *meters*[1] and *o'clock*. For instance, if the user commands robot 1 to go five meters forwards, the command to be uttered is formed as *"robot 1: 5 meters, 12 o'clock"*. If robot 1 should drive five meters to the right, the command is *"robot 1: 5 meters, 3 o'clock"*. The values for the parameters distance and direction are always specified relative to the robot (see also Figure 1).

B. Reused tools, software and platforms

In the following, toolkits and robot platforms utilized for the implementation of the voice control application are described.

Robots software and hardware: For the operation of the voice control we used robots of ATRV series from Real World Interface[2]. An ATRV-robot is a four-wheeled mobile platform equipped with sonar sensors and wireless ethernet communications. The ATRV-robots employ the software robot framework *RoSe* [3], [4], which serves as framework for control and communication among roboters. A C++ application is embedded in this framework and is called a *RoSe service*. The framework provides methods that allow RoSe services to communicate with each other via wireless link. A relevant service for the voice control is the *collision avoidance service* [5]. To this service a target coordinate in a two-dimensional robot-centered coordinate system can be passed. The service computes a path to the target coordinate which prevents to collide with obstacles. In order to put the robot in motion the collision avoidance service instructs a further RoSe service which is responsible for control of the robot's motor.

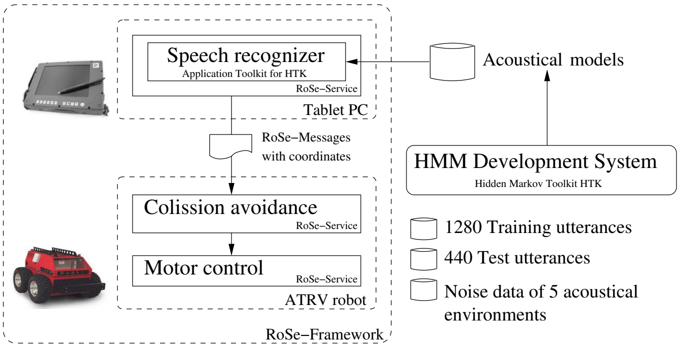


Fig. 2. Several software components arranged into an overall system for development, evaluation, and tests in outdoor environments

Toolkits for speech recognition: For development of acoustic models and evaluation under noise, the hidden markov Toolkit (HTK, [6]) was used. For embedding the developed acoustic models into a C++-application of the robots framework the *Application Toolkit for HTK* (ATK, [7]) was used. ATK is designed to create experimental applications based on HTK. It consists of a C++ layer which directly accesses the HTK library modules. ATK enables acoustic models that have been developed by HTK to be reused and integrate them into arbitrary C++-applications wrapped by ATK.

Composition of the voice control application: A combination of the software tools and hardware platforms described above composed the voice control application as follows. Acoustic models were trained with a development system based on HTK. These models were loaded into ATK. A RoSe service was written which utilizes the ATK libraries and, in particular, markov recognition algorithms and thus represents a RoSe service for voice control. Since the voice control appears as RoSe service, it is able to send messages with target coordinates to the collision avoidance service (see Figure 2).

C. Determining the speech recognizers application scope

This section describes the determination of adequate parameters for defining the speech recognizers application scope.

Vocabulary size: A small vocabulary of around 30 words was chosen to cover the required words for navigating the robot like, e.g., "robot", "meter", "o'clock" and several numbers ranging from "one" to "twelve".

Degree of speaker independence: Investigations dealing with speaker independence indicate different numbers of speakers required to achieve an acceptable degree of speaker independence [8], [9], [10]. According to *Lee* [9], at least 100 male speakers are in the training set as a minimum requirement for speaker independence. Furthermore, *Kubala* [10] shows that with 12 carefully selected speakers the same degree of speaker independence as a reference system can be obtained which was trained with 100 speakers. Thus, in this work utterances of 12 speakers of our research group were used as data basis. According to the statement of *Kubala* [10], it

[1] All commands have to be uttered in German. But for better readability they are written in English throughout the paper.

[2] formerly RWI, now iRobot, http://www.irobot.com

is assumed that even with this little number of speakers an approximately similar degree of speaker independence can be achieved as with a system that is trained with great speaker numbers.

D. Conception of Acoustic Modeling

The parameters that define the structure of the hidden markov models and the feature extraction are as follows.

Unit of speech: As unit of speech that has to be modeled by hidden markov models the word was chosen.

Number of states: The number of states per word-model was chosen dependent from words number of phonems but an extra state was added for the closure phase of plosives to model their non-stationarity more adequately. The number of states of the HMM was chosen to be linear to the number of phonemes in the corresponding word. This was intended to ensure that the phonetic structure of words is identical with the states of the HMMs. Furthermore, Bakis models [11] were used in which each of the next state may be skipped. This was intended, to take care of articulatory phenomena like vowel reduction.

Number of gaussian mixture densities: In literature there is no way known how to calculate the correct number of gaussian mixture densities. Accordingly, the number of gaussian mixture densities was kept variable. In development of acoustic models various numbers of gaussian mixture densities were tested in order to determine a suitable value.

Feature extraction: Mel Frequency Cepstral Coefficients (MFCC) were used to simulate a frequency sensitivity that is similar to that of the human ear.

E. Conception of the Acoustic Corpus

The overall speech corpus was recorded with a *Sennheiser PC 30* microphone.

Utterances to be spoken: It was determined to use speech samples out of continuously spoken utterances. Phenomena of coarticulation as they will occur in the operation of the speech are aimed to be included into the models. For instance, a sentence that had to be recorded looked like the following.

```
robot one drive ten meters twelve o'clock
```

Amount of utterances to be spoken: As requirement for HMM training for each model that should be trained at least 10, but better 50 or 100 samples should be available [8]. With a vocabulary of about 30 words, it should be sufficient to take 60 training utterances as basis to achieve a set of 20 references per word. Thus, in this work 60 training utterances were specified as minimum.

Manual annotation of utterances: Training procedures for hidden markov models require model specific pre-annotated audio data. For good results, at least an initial annotation for the models should be provided [12], [8]. For that reason parts of the corpus were manually segmented on word level. For the utterances for which no segmentation has taken place, a complete orthographic annotation was required instead.

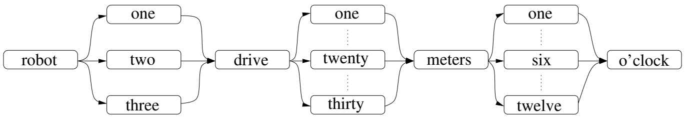


Fig. 3. Grammar network to cover navigation commands

III. Development and Implementation

In this section the development of the speech recognizer, its connection to the robots framework, and its integration onto a Tablet PC is described.

A. Realization of the Grammar

To allow for the requested operation the following grammar has been constructed. A command basically consists of a citation of the robot which is to execute a command, e.g., "robot two", and the actual statement, e.g., "drive ten meters towards twelve o'clock". Figure 3 depicts the grammar where silence models are omitted for better clarity.

B. Development of the Acoustic Corpus

In order to create the corpus, speaker utterances for training and for testing were recorded, afterwords post-processed, and finally partly annotated on word level with *Praat* [13].

Recording: The recordings have taken place in a carpeted large room with curtains. In total 220 training and 50 test utterances were recorded from the author. From each of the 12 speakers, 72 training and 50 test utterances were recorded. The whole corpus consists of 2290 utterances which includes the training part of 1850 utterances and the test part of 440 utterances. Issues related to the recording procedure were as follows. In order to maintain an adequate recording level and to avoid overmodulation, the distance to the mouth was readjusted for each speaker. For very loud or very soft voices the recording volume of the sound device had to be adjusted. Increasing the recording volume had to be done carefully in order to avoid too much inclusion of ambient noise into the signal. Sometimes it was difficult to maintain the same mean energy due to a movement of the microphone or a change of speaking volume. To ensure a flawless corpus, it was necessary to review the recorded utterances and, if some utterances were faulty, to capture those again.

Post-processing: The recorded utterances had needed to be post-processed such that only those audio data was included in the speech signals that were specified by our orthographic annotation. Thus, the utterances were freed from previous and following silence with standard sound-editing software. Random reviewing of temporal and spectral variation has taken place at approximately one third of the statements. Attention in inspecting the utterances was paid to modulation issues like insufficient modulation or overmodulation.

Manual annotation: All 240 utterances of the author were entirely manually annotated considering to use them for speaker-depended training. 10 of the 70 utterances of remaining speakers have been annotated manually aiming to use them for speaker-independent training. An automatic conversion

TABLE I
NUMBER OF UTTERANCES FOR TRAINING AND EVALUATION

Subject	Sex	Accent	Training	Segmented	Test
AK	male	Franconian	72	10	50
AT	male	Russian	72	10	10
BB	male	High German	72	10	50
DS	male	High German	72	10	50
FH	male	High German	72	10	50
FS	male	High German	0	0	10
HLW	male	High German	72	10	50
HM	male	Arabic	0	0	10
HN	male	High German	72	10	50
MS	male	High German	0	0	10
RC	male	High German	220	220	50
SR	male	High German	72	10	50
TB	male	High German	72	10	50
TR	male	High German	72	10	50
			1850	320	440

TABLE II
MODEL VERSIONS FOR SPEAKER-DEPENDENT TRAINING

Mixtures	References per model	WRR
4	≥ 5	97.56
4	≥ 10	98.00
4	≥ 21	94.17
4	≥ 21	99.04
4	≥ 23	100.00

TABLE III
MODEL VERSIONS FOR SPEAKER-INDEPENDENT SPEECH RECOGNITION

Mixtures	Training Speakers	Test Speakers	WRR
4	RC	SR	75.00
4	RC,SR	HN,DS	65.00
4	RC,SR,HN,DS	BB,TR,FS,MS,HLW	76.00
2	RC,SR,HN,DS,HLW	BB,TR,FS,MS	52.00
3	RC,SR,HN,DS,HLW	BB,TR,FS,MS	90.00
3	RC,SR,HN,DS,HLW,TB	BB,TR,FS,MS	90.00
3	RC,SR,HN,DS,HLW,TB,FH	BB,TR,FS,MS,HM,AT	93.34

of annotations from the Praat format into HTK format has been realized with linux scripts. Naturally appearing speaking sounds like those of smacking and exhalation were separately annotated intending to train particular models for them later on. When smacking sounds merged with subsequent verbal sounds those were included into the annotation boundaries of the whole word. Small pauses between words were handled such that the word boundary was set in the middle of the pause. Regarding the annotation of silence, it had to be ensured that the duration of the intervals was consistently the same. Therefore, we chose an interval of 110 to 130 ms for silence annotations.

C. Development of acoustic models

In literature there was no precise information found about how much data is necessary for training of acoustic models. For this reason, the development of acoustic models was keeping pace with the creation of the acoustic body. In each development step, a model version was created. Below the structure and development of acoustic models is described.

Creating the HMM structure: The number of model states N_s was initially selected for each model by $N_s = N \cdot (P + 3)$ with the word's number of phonemes P, and a weight factor N. A small number of 3 states were provided for the word's beginning and the word's end. After the results for $N > 1$ were significantly worse, N was set to 1. In determining the number of phonemes of a word an extra state was provided for closure phases of plosives. For instance, for the German word 'roboter' 11 states were provided where two states are considered to model the closure phases of /b/ and /t/.

Training of HMMs: The training of models was carried out in two main steps. First, a speaker-dependent model was developed and optimized for high detection rates. Second, the speaker-dependent model was gradually extended to other speakers to achieve a high degree of speaker independence. To keep the development cycle as small as possible, the entire exercise was automated with linux scripts for which HMM parameters could be specified. The varied parameters of the training were *number of gaussian mixture components* and *number of training reestimations*. Tables II and III show the development of the speaker-dependent and speaker-independent models. As expected, it was observed that the degree of speaker independence increased for each additional speaker in the training set.

D. Creating the link to the robot framework

After development of the acoustic models was finished, a RoSe service has been written in C++ in which ATK connects the acoustic models with the RoSe-framework (see Fig. 2). ATK provides methods for starting the speech recognition process and returns the word chain that is assumed to be uttered. From the recognized word chain values for distance and direction are extracted by regular expressions. If the distance r in meters and the angle α is given in clock the target coordinate (x, y) was determined by $x = r \cdot sin(\frac{\alpha \cdot \pi}{6})$ and $y = r \cdot cos(\frac{\alpha \cdot \pi}{6})$. The target coordinate is then sent via a RoSe-message to the RoSe service for collision avoidance which is responsible for further activation of the robot's motor.

E. Integration onto a Tablet PC and outdoor tests

In the integration and test of the speech recognizer on a Tablet PC, it was observed that the sensitivity of the microphone had to be adjusted such that less ambient noise was included in the signal. If the recording level was set too high, the detection rate fell off dramatically. This was noted especially for operation in a outdoor environments when the microphone was adjusted too sensitive because even little noise was included in the signal.

IV. EVALUATION

The overall development of the voice control described above has already revealed some details about the degree of

TABLE IV
RESULTS OF THE LEAVE-ONE-OUT-TEST FOR MEASURING THE DEGREE OF SPEAKER-INDEPENDENCE, 1 MIXTURE, REESTIMATIONS 1-10

Training data	Test data	WRR
HN,HLW,AK,BB,FH,TB,SR,AT,DS,TR	RA	98%
HLW,AK,BB,FH,TB,SR,AT,DS,TR,RA	HN	100%
AK,BB,FH,TB,SR,AT,DS,TR,RA,HN	HLW	100%
BB,FH,TB,SR,AT,DS,TR,RA,HN,HLW	AK	84%
FH,TB,SR,AT,DS,TR,RA,HN,HLW,AK	BB	100%
TB,SR,AT,DS,TR,RA,HN,HLW,AK,BB	FH	98%
SR,AT,DS,TR,RA,HN,HLW,AK,BB,FH	TB	98%
AT,DS,TR,RA,HN,HLW,AK,BB,FH,TB	SR	96%
DS,TR,RA,HN,HLW,AK,BB,FH,TB,SR	AT	100%
RA,HN,HLW,AK,BB,FH,TB,SR,AT,DS,TR	FS	90%
RA,HN,HLW,AK,BB,FH,TB,SR,AT,DS,TR	HM	100%
RA,HN,HLW,AK,BB,FH,TB,SR,AT,DS,TR	MS	100%
	Mean	97%

TABLE V
AVERAGE SOUND LEVELS OF MALE SPEAKERS IN 1 M DISTANCE OF SPEAKERS MOUTH FOR SPECIFIED SPEAKING STYLES; P = PRIVATE FIELD, FROM LAZARUS [16], SUPPLEMENTED WITH SPECIFICATIONS AT SPEAKERS MOUTH

Speaking style	1 m distance	3,125 cm distance
whispering	36 dB	66 dB
softly speaking	42 dB	72 dB
relaxed speaking (p)	48 dB	78 dB
relaxed, normal (p) speaking	54 dB	84 dB
normal, raised (p) speaking	60 dB	90 dB
raised speaking	66 dB	96 dB
speaking loudly	72 dB	102 dB
speaking very loudly	78 dB	108 dB
screaming	84 dB	114 dB
screaming maximally	90 dB	120 dB
screaming maximally (single cases)	96 dB	126 dB

speaker independence and vulnerability to noise. In this section these features are examined in more detail.

A. Speaker independent recognition

The speaker-independent training has shown that the degree of speaker independence increased with the number of speakers in the training set. For evaluation of the degree of speaker independence, only those speakers had to be used who were not in the training set of acoustic models. By repeatedly leaving out a speaker in the respective training set and testing only this specific speaker, i.e., performing a *leave-one-out test*, one measurement of the degree of speaker independence for this specific speaker could be achieved. Since a total number of 11 speakers were available, the exhaust test was repeated several times for each available speaker. Depending on the mean value of the results, the general degree of speaker independence was estimated. The test results are shown in Table IV.

B. Recognition in noise

In the following the speech recognizers evaluation under noise adversity of low and high degrees is described.

Acquisition of noise: For the environmental conditions *calm outdoor environment* and *busy street*, the noise was recorded with the same microphone used for creating the acoustic corpus. The adjustment of the recording volume was done manually such that it was initially set to zero and continuously increased up to a good modulation of amplitudes between values of 0.5 and -0.5. For every recording, the sound pressure level was measured in dB with a sound level meter. Noise of the high adversity environments *babble* and *track vehicles* was taken from the corpus *Noisex* [14].

Overlaying procedure: Both the speaker-dependent and speaker-independent models were evaluated. For speaker-dependent evaluation in noise, test utterances of speaker RC were used. For speaker-independent evaluation in noise, test utterances of all speakers were used. Furthermore, noise adversity was simulated by artificially superimposing the clear commands from the test corpus with the software tool *FaNT* (Filtering and Noise Adding Tool, see [15]). The operation of FaNT requires SNR values to be specified which represents the intensity with which the noise superimposes the speech signal. The ratio of signal to noise depends on the sound level of speech and on the sound level of the noise. Sound levels of noise were measured in case of recording and are specified by *Noisex* [14] in case of noise corpus usage. Sound levels of speaking styles are taken from Lazarus et al. [16]. They provide average sound levels of different speaking styles, i.e., whispering, speaking softly, and relaxed speaking. Therefore, men produce by whispering at a distance of one meter a sound pressure level of 36 dB. By screaming, up to 96 dB can be achieved in some cases. For a summary of the data collected see Table V. Thus, utterances were superimposed at several SNRs in the range of small SNRs where the noise was barely noticeable up to large SNRs where the speech was hardly intelligible. In particular, superimposition ranged from -5 dB to 50 dB SNRs in 1dB steps. For the various noise scenarios, the test corpus was superimposed several times and new modified testing corpuses were created. The speech recognizer was then scheduled on the created corpora and word recognition rates were logged. This overlaying procedure has been used for noise evaluation of speech recognition in several works, e.g., [17], [18].

Overlaying Results: The sounds of a *calm outdoor environment* were obtained by recording at an average sound pressure of 39 dB. It can be seen that in a quite acoustic environment with a noise level of 40 dB, it is sufficient to speak softly to achieve very good recognition rates of at least 95% (see Figures 4 and 5). The sounds of a *busy street* were obtained by recording in a distance of 5 meters at an average sound pressure of 61 dB. The experiment showed that no good word recognition rates were possible when speaking relaxed. But by increasing the speaking volume good detection rates above 90% were achieved (see Figures 6 and 7). The *babble sounds* came from a large hall in which 100 people spoke to each other producing an average sound level of 88 dB. Hence, by

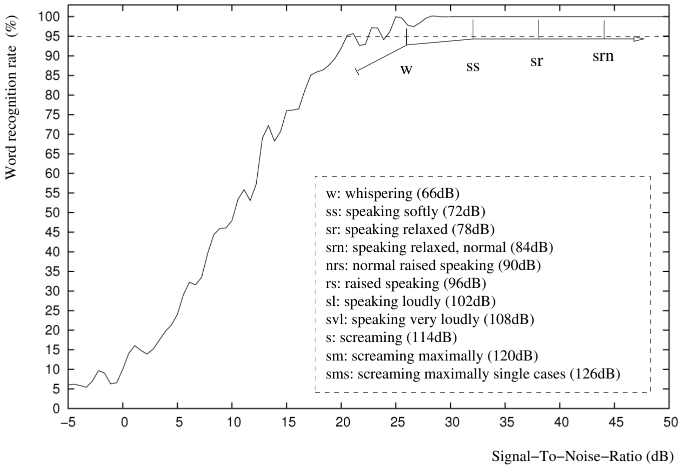


Fig. 4. Speaker-dependent recognition in 40 dB noise of a calm outdoor environment

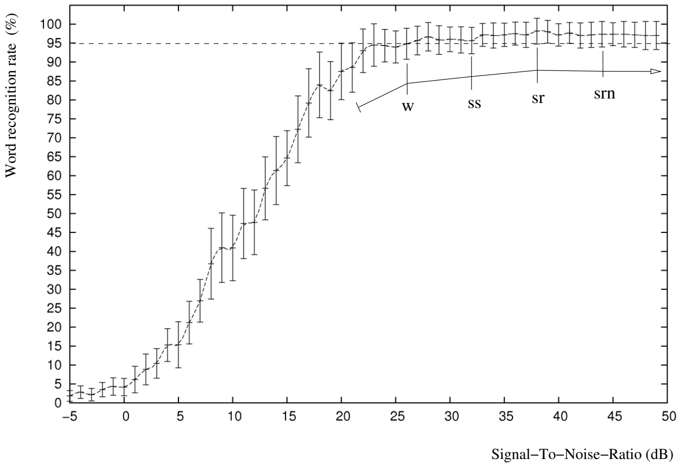


Fig. 5. Speaker-independent recognition in 40 dB noise of a calm outdoor environment

speaking very loud with 108 dB effecting an SNR of 20 dB, it could still be possible to achieve acceptable word recognition rates of around 90% (see Figures 8 and 9). The in-vehicle sounds came from *track vehicle 1* driving at a speed of 30 km/h producing an in-vehicle sound level of 100 dB. Good detection rates were virtually not able to be achieved. With a maximum achievable speaking volume of 120 dB, it would be theoretically possible even to achieve 80% recognition rate (see Figures 10 and 11). The in-vehicle sounds came from *track vehicle 2* driving at a speed of 70 km/h. The sound power level was specified with 114 dB. The results of the experiment may be taken from Figures 12 and 13. For those background noises virtually no satisfactory recognition rates were achieved.

V. Discussion and Conclusions

This section describes finally which conclusions were made and how this work can be used as a basis for further developments.

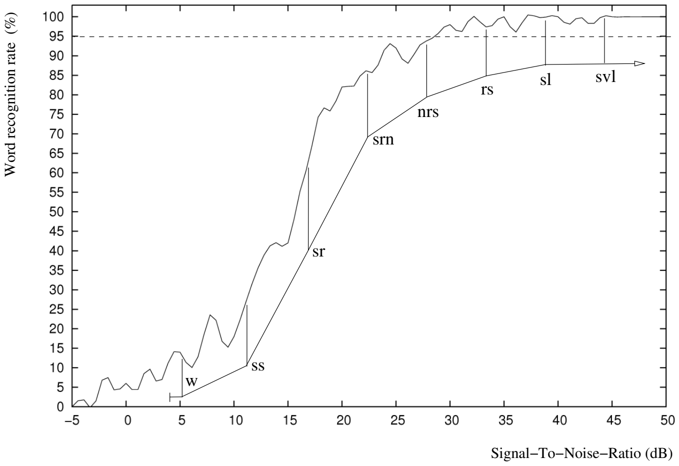


Fig. 6. Speaker-dependent recognition in 61 dB noise of a busy street in 5 meters distance

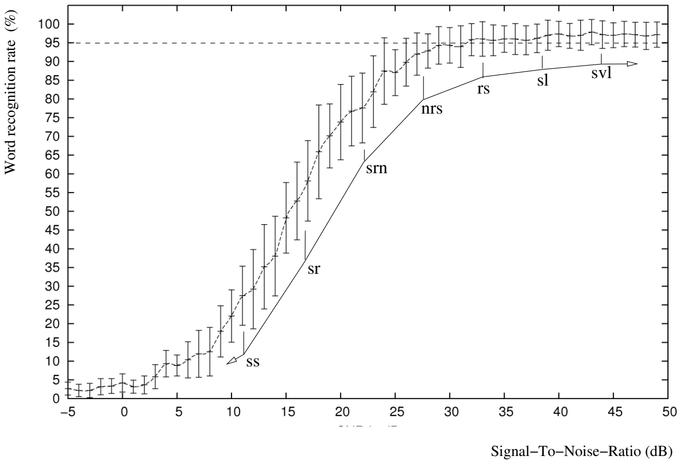


Fig. 7. Speaker-independent recognition in 61 dB noise of a busy street in 5 meters distance

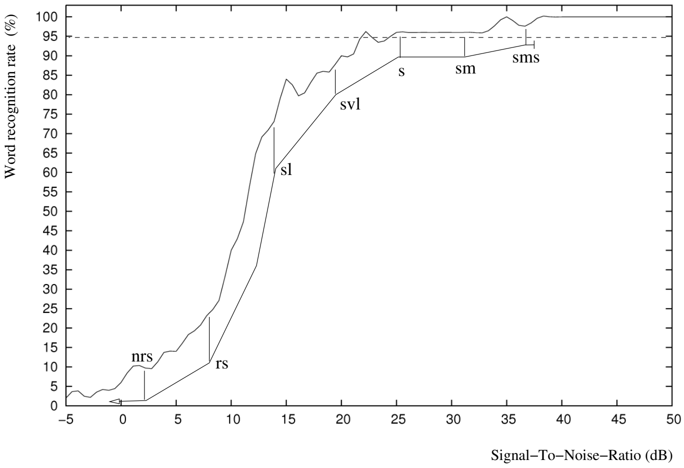


Fig. 8. Speaker-dependent recognition 88 dB noise of a crowd of 100 people in a large room

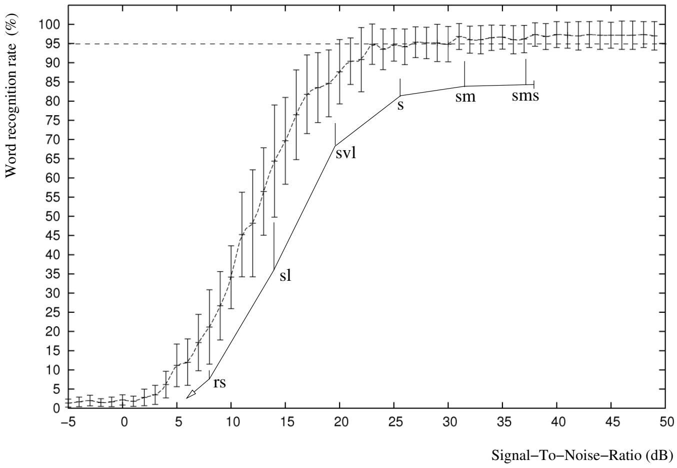


Fig. 9. Speaker-independent recognition 88 dB noise of a crowd of 100 people in a large room

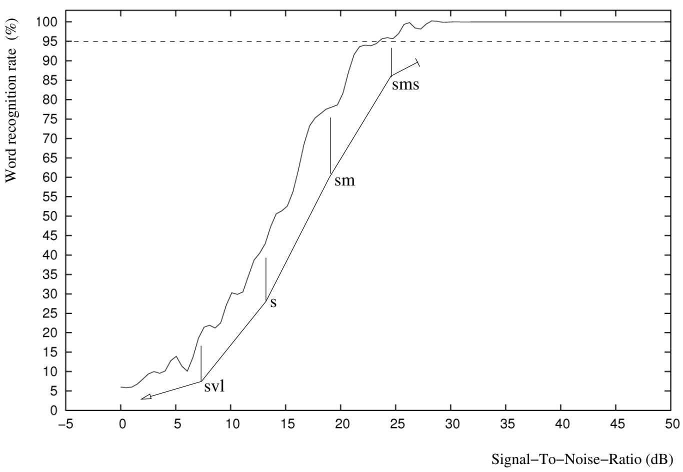


Fig. 10. Speaker-dependent recognition in 100 dB noise from within a track vehicle driving with 30km/h

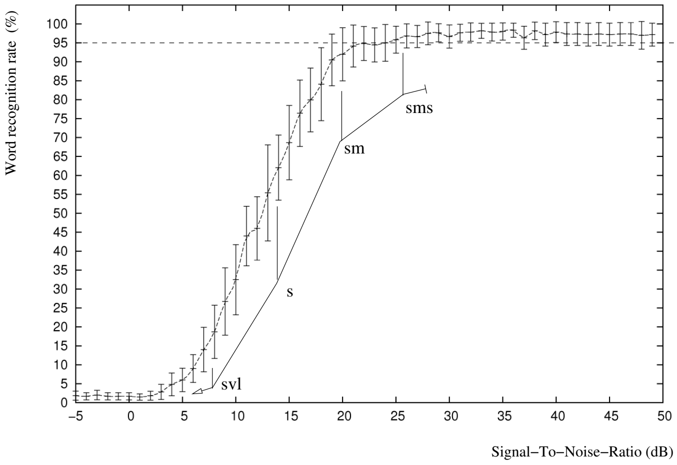


Fig. 11. Speaker-independent recognition in 100 dB noise from within a track vehicle driving with 30km/h

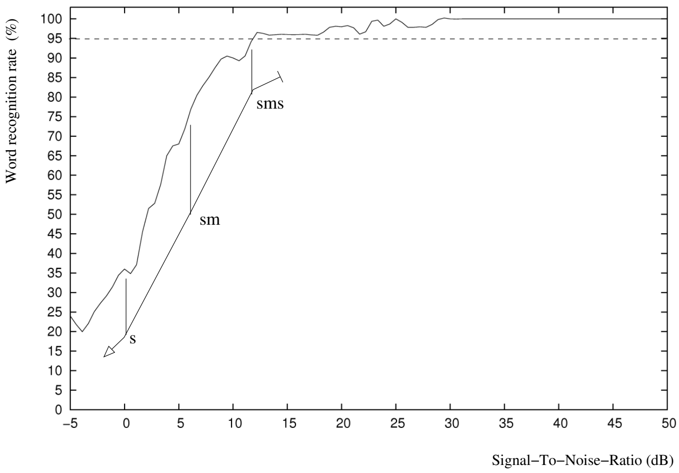


Fig. 12. Speaker-dependent recognition in 114 dB noise from within a track vehicle driving with 70km/h

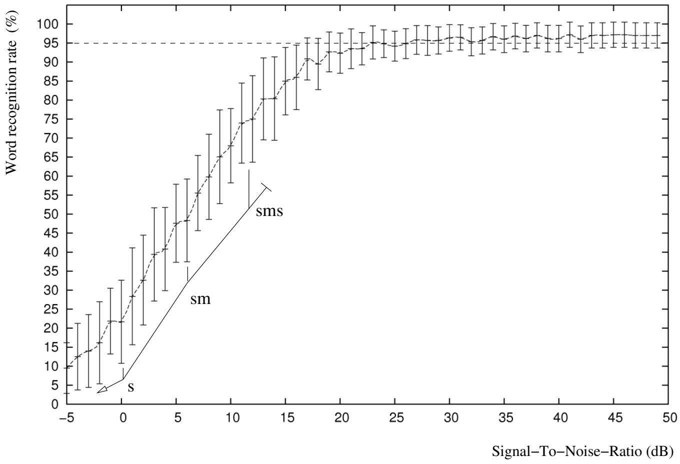


Fig. 13. Speaker-independent recognition in 114 dB noise from within a track vehicle driving with 70km/h

A. Acoustic modeling and degree of speaker independence

The training of acoustic models in Section III-C proved, that a number of around 20 samples per hidden markov model can be sufficient for normal operation of a HMM based speech recognizer. In the evaluation of speaker independence, it was shown that in a speech recognizer with a small vocabulary and a small number of speakers of around 10, a relatively high degree of speaker independence can be achieved.

B. Noise vulnerability

The evaluation had shown that when using a speech recognizer with vocabulary of solely 30 words in adverse environments of around 60 dB noise can be tackled by speaking with a raised voice in order to achieve a recognition rate of about 95%. In environments with extreme noise conditions from 80 dB up to 150 dB, the speech recognizer can no longer be used satisfactorily due to detection rates of around 80%. Here, a method for compensation of noise should be taken into account in conception of the speech recognizer. It could be summarized that keeping a speech recognizers vocabulary small does not compensate adverse noise of high sound levels above 100 dB.

However, it should be noted that when speaking louder, not only SNR increases but also the vocal tract changes its shape and produces frequencies for which the acoustic models were not trained. The results obtained here are subject to some inaccuracy since the identified speech sound levels were calculated directly at the mouth and thus only represent an estimate. Furthermore, many microphones feature polar patterns so that the noise received on the side of the microphone is attenuated more than those which enter head-on. It can be concluded that slightly higher SNRs and better recognition rates could be achieved. The comparison of the speaker-specific and speaker-independent results suggests that models that were trained by much more data and therefore having a greater generality, do not necessarily offer worse results than models that were only trained by a speaker and are much more specific.

Regarding possible further developments noise sensitivity can be reduced based on several methods. Besides conventional signal filtering methods human strategies for speech understanding in noise can be employed. A recent survey about findings of human strategies for noise compensation can be found in *Loizou* [17].

C. Voice as input mode for navigation tasks

The usage of speech as an input mode for the control of robots must be carefully planned. In this work, speech which represents a verbal mean is used to perform a spatial continuous operation task. The objective called to enable a discrete navigation of the robots on an arbitrary two-dimensional ground. In considering how this task could be completed by means of speech, the continuous task was transformed into a discrete task by instructing the user to specify a target coordinate in a two-dimensional system that the user must consider first. The spatial thinking user has to transform his intention to move first into a verbal command which causes an additional cognitive load and costs time.

Regarding possible further developments, a holistic designed interface could take the requirements of the operation task and the expectations of the operator into account. The development should take place through an iterative design-implementation approach and the human-robot interface in field evaluation should be kept at pace with development. The result should be an effective human-robot interface that allows the user, even under extreme conditions like stress and noise, for a consistent, effective, and fast control of the robot. For control by voice discrete operating activities are suitable. It would be conceivable to raise the navigation commands for navigation to a higher level of abstraction. For example, the user could navigate the robots as follows:

- "Drive back to command center", or
- "Drive to robot group A, drive to robot group B"

Furthermore, semi-autonomous functions of the robot could be controlled. Examples of such instructions are:

- "Follow robot A",
- "Explore area, radius 50 m", or
- "Search for intruders".

References

[1] S. Yamamoto, K. Nakadai, J. Valin, J. Rouat, F. Michaud, K. Komatani, T. Ogata, and H. G. Okuno, "Making A Robot Recognize Three Simultaneous Sentences in Real-Time," *Proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005.

[2] O. Majdalawieh, J. Gu, and M. Meng, "An HTK-Developed Hidden Markov Model for a Voice-Controlled Robotic System," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.

[3] A. Tiderko and T. Bachran, "A Service Oriented Framework for Wireless Communication in Mobile Multi Robot Systems," in *ROBOCOMM 2007: First International Conference on Robot Communication and Coordination*, Athen, 2007.

[4] ——, "A Framework for Multicast Communication over Unreliable Networks in Multi Robot Systems," *Proceedings of Towards Autonomous Robotic Systems*, 2007.

[5] F. Höller, *Personenbegleitung mit mobilen Robotern: Lokale Navigation mit probabilistischen Roadmapverfahren, FKIE-Bericht. Nr. 134.* Forschungsgesellschaft für angewandte Naturwissenschaften, Wachtberg, FKIE, 2007.

[6] S. Young, "The HTK Hidden Markov Model Toolkit: Design and Philosophy," Cambridge University (UK), Department of Engineering, Tech. Rep. 153, 1993.

[7] ——, "ATK - Application Toolkit for HTK," University of Cambridge, 2007.

[8] E. Schukat-Talamazzini, *Automatische Spracherkennung: Grundlagen, statistische Modelle und effiziente Algorithmen.* F. Vieweg, 1995.

[9] K.-F. Lee, *Automatic Speech Recognition: The Development of the SPHINX Recognition System (The Springer International Series in Engineering and Computer Science)*, 1st ed. Springer, 10 1988.

[10] F. Kubala and R. Schwartz, "A New Paradigm For Speaker-independent Training," in *ICASSP '91: Proceedings of the Acoustics, Speech, and Signal Processing. ICASSP-91.* Washington, DC, USA: IEEE Computer Society, 1991, pp. 833–836.

[11] R. Bakis, "Continuous speech recognition via centisecond acoustic states," *The Journal of the Acoustical Society of America*, vol. 59, 1976.

[12] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[13] P. Boersma and D. Weenink. (2008) Praat: Doing Phonetics by Computer (Version 5.0.32). [Online]. Available: http://www.praat.org

[14] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. Vol. 12, pp. 247–251, 1993.

[15] H. G. Hirsch. (2005) F a N T - Filtering and Noise Adding Tool. Hochschule Niederrhein. [Online]. Available: http://dnt.kr.hs-niederrhein.de

[16] H. Lazarus, A. C. Sust, R. Steckel, and K. P. Kulka M., *Akustische Grundlagen sprachlicher Kommunikation.* Springer, 2007.

[17] P. C. Loizou, "Comparison of Speech Enhancement Algorithms," in *Speech Enhancement: Theory and Practice (Signal Processing and Communications).* CRC, 2007, ch. 11, pp. 545 – 555.

[18] H. Hirsch and D. Pearce, "The Aurora Experimental Framework for the Performance Evaluation of Speech Recognition Systems under Noisy Conditions," *ISCA ITRW ASR2000 Automatic Speech Recognition: Challenges for the Next Millennium*, 2000.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 389–396

ISBN 978-83-60810-27-9
ISSN 1896-7094

The Role of the Newly Introduced Word Types in the Translations of Novels

Maria Csernoch
University of Debrecen,
Egyetem tér 1. Debrecen, H-4029, Hungary
Email: csernoch.maria@inf.unideb.hu

***Abstract*—The project detailed in the article is able to find the vocabulary rich segments of novels in different languages. The method used here takes into account the frequency of the words of the text, and based on this information we are able to create artificial texts with the same parameters. Since the original and the artificial texts share parameters they are comparable and we can find those segments of the original text which are richer in vocabulary then it is expected as compared to a random selection of the words. The advantages of finding these vocabulary rich segments of the text, beyond that they give an insight of the development of the vocabulary of a novel, is that in any translation, adaptation process it is a great advantage being familiar with these sections of the text.**

I. Introduction

THE PRIMARY goal of this project was to find a dynamic first-order statistical model [5] with the help of which we are able to find the vocabulary rich segments of the texts written in natural languages. By vocabulary rich segments we understand those sections of the text where the number of the newly introduced words is higher than it is predicted by the model. These sections of the text make their stand out by being richer in vocabulary than the main body of the text.

After finding such a model, and – based on the model – the text slices corresponding to the vocabulary rich segment of the text the question arose, what could be the reasons for their presence. I found several previously published works which venture to give explanations for this phenomenon but these opinions are subjective, mainly based on the intuitions of the reader. I was interested in finding more objective explanations, which are based on quantitative data, for what makes the author of a text to increase the number of newly introduced words so intensively that the process makes qualitative changes in the text.

Finally, my aim was to explore how the proportion of newly introduced words varies in different adaptations of the same text. The term ‘adaptation’ is used here to include both intralingual and interlingual adaptations, whether involving reduction in text size or not. This means that beyond the theoretical significance that we were able to find a method with the help of which we can separate the vocabulary rich segments from the main body of the text, the practical usage of the method is also remarkable. One of the advantages of this statistical analysis is that it is invariant to the fact that translators are free to use words, expressions, structures, different kind of techniques well known in translation studies. The only expectation here is that in a faithful translation the text segments in the target language should be as rich in vocabulary as they are in the original text. Both translators and critics are offered the location and the intensity of the vocabulary rich segments of a text. On the one hand, prior to the translators’ effective work becoming familiar with the vocabulary rich text segments of the original work might help the translator(s) to pay more attention to these unique slices of text. They are unique in the sense that the author of the original text used richer vocabulary to call the readers’ attention to them. As a result, we would be able to reach a more faithful translation. On the other hand, being familiar with the vocabulary rich text segments of both the original and the translated texts means that a new set of parameters is added to the tools of translation criticism. Here we would be able to provide more objective analysis of the translation(s), we would be able to tell how close a translation is to the original text in vocabulary, or decide which translation is closer to the original text.

II. Methods

A. Newly introduced words (word types and lemmas)

The analyses of texts highly depend on the notion of newly introduced word types. First the definition of newly introduce word types should be given.

Newly introduced word types have meaning in a closed text, at a certain point of that particular text. When the first appearance of the word is detected at a certain point of the text, the word is considered as newly introduced in that text.

Beyond that newly introduced word types are understood within a closed environment, unlike other word concepts newly introduced word type is relative. The title being newly introduced is only temporary, and the word is in the range of this concept until its second appearance in the text. The definition of newly introduced words contain that hapax legomena of the text are those words which never loose this title.

B. Dynamic first-order statistical models

To be able to follow the flow of a text instead of the previously used static models a dynamic model should have been created. The advantage of the dynamic model to the static models is that it gives data not only in general but in

978-83-60810-27-9/09/$25.00 © 2010 IEEE

the comparison to the original text it is able to follow the changes of the text.

The model belongs to the category of first-order statistical models [5] because it takes into consideration only the frequency of word types of the original text. A first-order statistical model can be created with the urn model. The urn model for word frequency distributions compared the use of words to the sampling of marbles, balls from an urn [3]. The essence of urn model is the following. Consider an urn containing marbles of various colors. Each color corresponds with a marble type. A particular color may be extremely rare, or it may be represented by a great many individual marbles, the marble tokens. We randomly draw N marbles from the urn, assuming that the outcome of a given trial is completely independent from the outcome of any other trial [17], [19], [27], [1], [2], [3]. In Baayen's opinion [2], [3]the urn model is responsible for the overestimation bias, which means that using a model, where the polynomial distribution of words are assumed, produces a larger size of vocabulary, specially in the first half of the text than the observed vocabulary.

To decrease the difference between the observed and the predicted vocabulary instead of assuming the polynomial distribution of the words, the hypergeometrical distribution of them should be applied. The difference between the two models seems to be minor, but assuming the hypergeometrical distribution of the words the overestimation is usually not longer than a couple of thousand words at the very beginning of the texts and later on the observed vocabulary fluctuate around the expected vocabulary.

If we assume that the word types of the texts can be modeled by the balls with different colors of an urn, and from each color as many balls are stored as the frequency of the word type than for polynomial distributions the balls are randomly selected from the urn and are returned after taking notes of its color [3], [7], [8]. The selection of balls is continued until we reach the number of tokens of the original text.

With the assumption of the polynomial distribution of the words, however, there is no clear guarantee that when reaching the desired number of tokens the number of different word types of the observed text equals the number of the different word types of the model.

The hypergeometric distribution of the words, on the other hand, can be modeled by an urn where the balls are not returned after inspecting its color. With word types the algorithm is the following. The number and the frequencies of the different word types of the original text are counted and then all the found word types are stored in a one-dimensional array as many times as their frequencies indicate. The size of the array equals the size of the text that is the number of the tokens in it. The random selection of a word type beyond inspecting it means the erasing of it from the array. Using this method the algorithm might slow down towards the end of the process, because as we advance in selecting the words the place of the erased words are reselected more often than at the beginning. To speed up the algorithm after selecting and deleting the word from the array the cell was not left on its original location, rather, the array was compressed by moving the elements forward. In the next step a new word was selected from this, one cell shorter, array. As a result, even to the longest novels I ever met the selection was carried out within two minutes. The great advantage of the hypergeometric urn model to the polynomial urn model that the number and frequency of the word types in the model equals to that of the original text. All this was carried out with DyMoCASAT (Dynamic Model for Computer Aided Analyses of Texts), a program designed for these special analyses).

C. Texts and their different adaptations

In this project primarily English and Hungarian novels were compared to their different adaptations. The selected adaptations were the human foreign language translations to English, French, German, and Hungarian, and the machine translations to English and Hungarian languages. Here the only publicly available English–Hungarian translator program [20] was used to create the translations and then the same method with DyMoCASAT was applied to these texts. Beyond these inter-language adaptations the lemmatized and non-lemmatized versions and finally the condensed versions of the texts were compared.

Beyond the comparison of the original texts to their adaptations different translations to the same language are also compared to each other. This kind of comparison might reveal differences in connection with changes in vocabulary. We would be able to find those sections of the texts where the translation is richer or shallower in vocabulary then the original text or another translation. To find explanation for the presence and hiatus of the vocabulary rich segments of the texts the appearance of hapax legomena was also tested. The method for the distribution of hapax legomena were similar to that of the word types.

At this stage of the project the different adaptations of The Jungle Books (Rudyard Kipling), The Da Vinci Code (Dan Brown), The Adventures of Tom Sawyer (Mark Twain), The Adventures of Robinson Crusoe (Daniel Defoe), Alice’s Adventures in Wonderland, Through the Looking-Glass and What Alice Found There (Lewis Carroll), Sorstalanság (Kertész Imre) are analyzed, along with some novel without their translations. However, I have to note that the selected texts and the advancing on the texts highly depend on the availability of the printed and the digitized form of them. Most of these texts are manually scanned and digitized because their availability differs due to various reasons. To get comparable results and make the texts readable for DyMoCASAT, processable for the lemmatizer programs the texts should be converted into plain text with well defined borders of paragraphs.

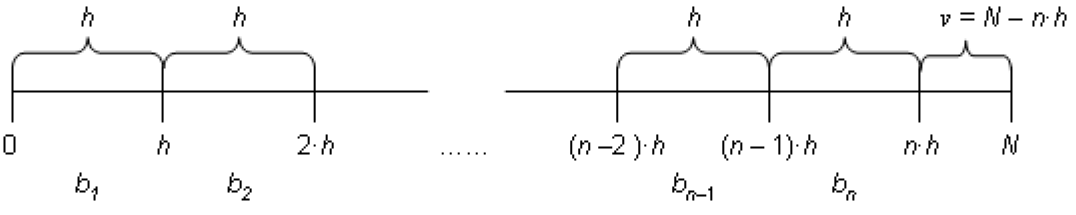


Fig 1. An *N* token-long text is divided into *h*-token-long blocks. The length of the blocks is usually one-hundred tokens. This choice of length is proved to be a good selection because it is longer than the average length of sentences, so syntactic constrains should be taken into account.

III. Results

A. Determining the number of newly introduced word types of a text

To carry out the analysis DyMoCASAT is used [7], [9], [10]. The first step in this process is the analysis of the text. At this stage the number of the tokens (*N*) and then the number of different words (*V*(*N*)) are counted in the text. In the second step the text is divided into intervals of equal length (*h* is the length of an interval, which is usually one-hundred token-long), called blocks.

In each block the different words are identified and the number of their occurrence is counted. The amount of data gained from the text with this method required a time and space consuming data storage. We had to be prepared for further analysis of these data, and had to find a data structure which can be searched with reasonable speed. The solution for this problem is a theoretical three-dimensional matrix. The matrix is theoretical because in practice the data stored in a number of text files. The number of files equals the number of different initial characters found in the text. Each file is identified by these initial characters. This is the first dimension of the matrix. Within each file as many different words are stored as found in the text starting with the same initial character. This is the second dimension of the matrix. Following the paragraph of the word the numbers of occurrence of the word in the blocks are stored. To be able to store numbers greater than ten with only one digits we had to find a numerical system greater than ten. Working with one-hundred token-long blocks even the hexadecimal numerical system seemed small, so we decided on using a system with twenty-seven digits. In this system the numbers are replaced with the letters of the English alphabet and the zero with a character from outside of the alphabet. The last character of the paragraphs indicates the last block where the selected word was found. This is the third dimension of the matrix.

The advantage of storing the occurrence of the words in this theoretical matrix lies again in the numbers. Let us denote the number of different initial characters as k, considering the most frequent initial character as m, and the maximum number of blocks as n. In a real three-dimensional matrix this amount of data requires a k × m × n celled storage place. In practice, however, we might have characters which do not start words, the number of paragraphs in each file differs based on the frequency on the initial character, and finally, the length of the paragraphs also differs based on the last occurrence of the word.

With this method all the information about the appearance of the words is stored, so in case the whole text would be restored within the limit of the length of the blocks.

To each block the number of newly introduced words, the number of hapax legomena, the number of different words can be assigned depending on the goal of the analysis.

Our primary aim was to follow the changes of the newly introduced word types, so the number of these words in each block has to be counted (Fig. 2, upper left panel). In general, the number of newly introduced word types follow a monotonic decay. However, we can find sections where this monotonic decay is reversed and a sudden increase then a sudden drop can be detected in the number of the newly introduced words. We are interested in finding the location of these sections and finding reasons for their presence. By mapping the number of newly introduced words, where the domain of the graph is the blocks of the text and the range is the number of the newly introduced words to each block, the protuberances of the graph suggest the location of these unique sessions of the text. Suggestions however, are closer to subjective judgments than objective facts, so we had to find a more reliable analysis of the graphs than just looking at them. The first step following the mapping of the original data is to rule out the accidental rises of the graph. This can be done by the smoothing of the graph (Fig. 2, middle left panel). As the result of the smoothing only the secondary protuberances of the graph are left, which we are interested in. These secondary protuberances can be grouped into two sets. Those which stand for significant changes create the first group, those which not belong to the second. To be able to distinguish the two sets of the protuberances the following method was invented.

IV. Determining the significant changes in newly introduced words

After collecting the data of the selected text using the same program (DyMoCASAT) first-order statistical models can be built to the text. As it was detailed in Mesthods, the assumption of hypergeometric distribution of word provides more reliable data than the polynomial distribution of them. Based on this assumption DyMoCASAT was extended to be

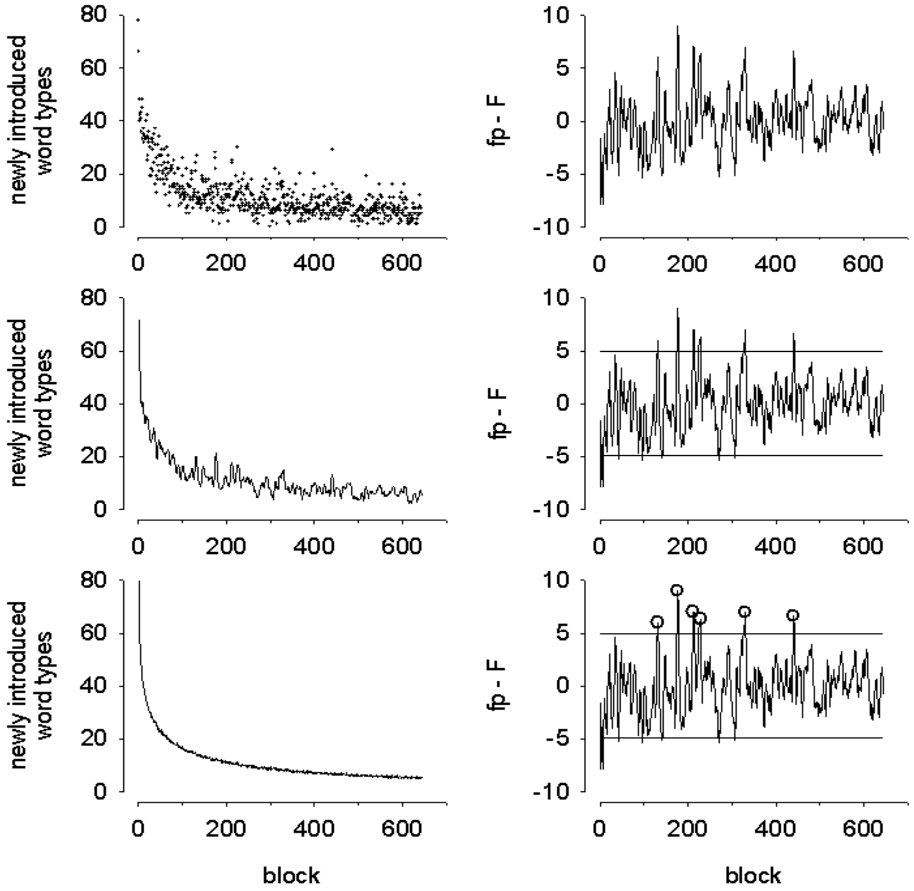


Fig 2. The major steps in the process of analyzing the introduction of word types and lemmas with DyMoCASAT in the condensed English version of The Da Vinci Code. First, the newly introduced words were counted in each block (left, upper), then this function was smoothed (left, middle). In the third step 100 artificial texts were created and the number of newly introduced words was averaged for each corresponding block (left, lower). The difference between the averaged artificial and the smoothed function was calculated (right, upper). The level of significance was determined as M + 2SD (right, middle). Those protuberances were considered significant which exceed the significance level (right, lower).

able to create artificial text. The main characteristic of the artificial text that it carries as many words as the original text with exactly the same frequency. The artificial text is a gibberish, but that is the consequence of the random selection of the words [7]– [9]. To build this model a random selection of words was carried out, the outcome of which was an artificial text.

After creating an artificial text with the same parameters as the original text, the same analysis can be carried out to this text. Using DyMoCASAT, we again are able to create the theoretical three-dimensional matrix, storing all the data considering the appearance of the words. In theory these sets of data are comparable. The random selection, however, by its very nature might cause unpredictable changes in the number of the newly introduced words [27]. To rule this possibility out, a hundred artificial texts were created and averaged ($F(n)$; Fig. 2, left lower graph).

The next step in the process was to determine the differences between the artificial texts and the original ($fp(n)$–$F(n)$; Figure 2 right, upper graph). The mean (M) and the standard deviation (SD) of the differences of $fp(n) - F(n)$ were counted (Fig. 2, right middle graph). The values considered distinguishable are those which exceed the $M + 2SD$ value (Fig. 2, right lower graph). In earlier studies these sudden increases were shown to mark 'longish' inlays in the text and, furthermore, their positions and distribution were found to be unique to the given text [6], [10]).

The result of the analysis is always a graph (Fig. 2, right upper graph – Fig. 2, right lower graph). The flow of this graph is unpredictable, since there is no clear evidence on what it is that makes a writer to increase the number of newly introduced words, and at what point in the story. The corresponding graphs for such pieces of literature always provide protuberances in the graphs.

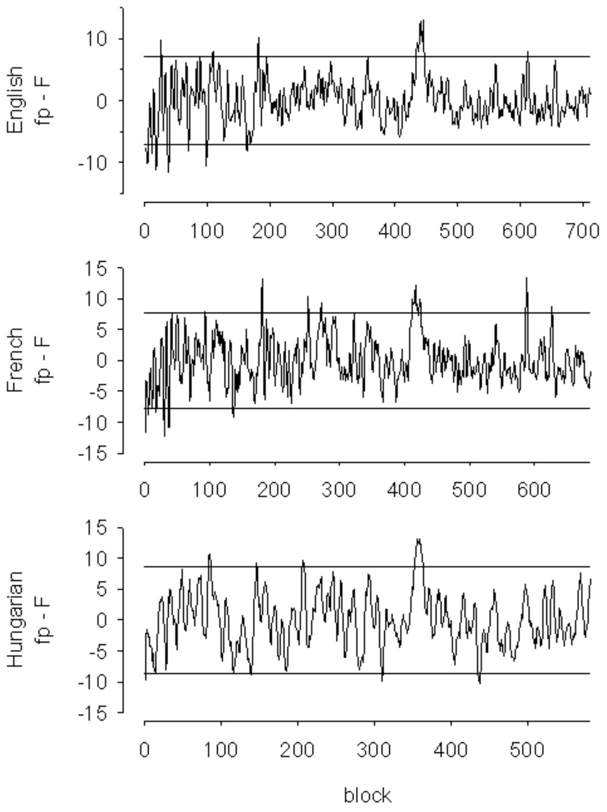


Fig 3. The newly introduced word types of The Adventures of Tom Sawyer and its French and Hungarian translations.

Most of the readers believe that significant changes in the number of newly introduced word types are in close connection with the chapter boundaries. However, in Genette's [16] opinion these vocabulary rich segments of the texts can appear anywhere in the text and carry 'functionally useless' information. This means that if we leave them out the text would still be understandable.

During the proofreading of the texts it was found that these vocabulary rich segments of the text mainly stand for longish descriptions of characters, settings, historical events, and changes in style or language. These vocabulary rich segments of the text are presented on the graph of the newly introduced word types as protuberances. The protuberances have two parameters which are able to describe them. These are the length and the intensity of the protuberances. The length of the protuberances are the number blocks or tokens through which the number of newly introduced word types are over the significance level. The intensity of a protuberance gives the number of newly introduced word types over the significance level. Considering all these we can see that the graph of the newly introduced word types are unique to each text, by them we are able to identify the texts to which they belong. They are like the graphs of sound files. The analyses did not prove that these vocabulary rich segments are expected at the chapter boundaries. They can appear anywhere in the text and the most robust protuberances are proved to be those which are due to changes in style regardless of these boundaries.

In Fig 3 the newly introduced word types of The Adventures of Tom Sawyer are mapped in the original English text, and in the French and Hungarian translations. First the problem of different number of tokens had to be solved ($N_{English} = 711$, $N_{French} = 685$, $N_{Hungarian} = 581$). By the normalization of the domain of the graphs the texts are comparable regardless of their lengths. It is obvious from the graphs that the most robust protuberance is between blocks 434 and 446 in the English text. This protuberance stands for a change in style, the students' writings for their school leaving exam, while the others, smaller both in length and intensively, stand for descriptions. In both translations the change in style is followed remarkably well, while the descriptions are not necessarily. We can find missing protuberances in both languages, and the other way around, protuberances of the translations which were not significant in the original text.

The theory that the protuberances appear at chapter-boundaries was proved wrong by comparing the original texts and their translations where the chapter boundaries of the translated texts are changed as compared to the original text. The vocabulary rich segments of the original and the translated texts stand for the same text segments regardless of the old and new chapter-boundaries.

A. Languages of the texts, word types vs. lemmas

The other question arose in connection with the analysis of the newly introduced word types, and consequently with the vocabulary rich segments of the texts that whether this feature is language independent or not. To test this first DyMoCASAT should be able to read texts of different languages. The primary language of the program is English. However, the program, through its menu, offers the users the opportunity to create their own alphabet, upload this alphabet, and analyze texts in languages different from English. When texts in languages so different in nature are compared the question arose whether the word types are satisfactory enough for such analyses, especially in the agglutinating Hungarian language, due to the fact that affixes attached to the lemmas might increase the number of word types without semantic background. Lemmatization were carried out to English [23], [24] and Hungarian [21], [22] texts. In both languages lemmatizer programs were used to carry out the lemmatization, then the results of the programs were mended to gain comparable data to word types. In this form of the text the lemma and its part of speech tag were concatenated and stored as a single word.

The pattern formed by the newly introduced word types and lemmas show great similarities in both languages [11]. There might be detectable differences between the length and intensity of the vocabulary rich segments. However in general, the vocabulary rich segments of the texts both in the lemmatized and non-lemmatized texts are at the same locations. The only difference found between the English and the Hungarian texts was that at the beginning of the Hungarian non-lemmatized texts the word types might hide significant protuberances by forcing their peaks below the significance level.

B. *Texts and their foreign languages translations*

Analysis of the introduction of new words (either word types or lemmas) in the foreign language translations of a text reveals a novel feature of the translated text. Since texts contain several untranslatable elements, and elements which the translator is not willing to reproduce for various reasons, their replacement with other lexical elements is acceptable [4], [26], [25]. To accept this freedom in the process of translation we can accept the result of the translator program with its serious problems with forming sentences, and finding the suitable words and expressions. From the point of view of translation theory it should be interesting to know how changes in the vocabulary of translations follow the changes of the vocabulary in the original text. With the method described above we can decide whether the translation is as rich in vocabulary as the original text. Our goal is not judging the translation, but checking how faithfully the translation follows the original text in vocabulary. This means that our goal is to highlight those segments of the text which are richer in vocabulary than the main body of the text, and in an already existing translation reveal the shallower and richer text segments.

On the thoroughly analyzed texts (listed in MethodsI) we were able to tell for each translation how faithfully it followed the vocabulary changes of the original texts. On the single texts we were able to find their vocabulary rich segments of them, which might be used for further analyses when their translations are available. However, the found and presented vocabulary rich segments by themselves give useful information for the translators previous to the translation process [12]- [14].

In Fig. 4 the comparison of three different German translations of The Jungle Book to the original text clearly show the differences in the changes of vocabulary. The details of the comparison of these texts reveals the differences of the three translations.

Three different German translations were found for the analysis: Mikush's adaptation from 1951 (from now on Mikush, 1951), and two translations from 1987 from Haef and Harranth (from now on Haefs, 1987, and Harranth, 1987). Among the three texts there is only one which is a full text (Haefs, 1981), unfortunately, the others are cropped to some extent. To be able to make comparison of the three texts they all should have been cropped to the shortest text (Harranth, 1987).

In Fig. 4 the newly introduced word types of the six-story-long English and German translations are mapped. In the order of appearance, the first significant protuberance of the English text stands for the King's Palace in Kaa's Hunting, the second is for the fights of Sea Catch for their territory, the third is the introduction of Rikki-Tikki-Tavi, while the fourth is for the story of Toomai.In Harranth, 1987 all four protuberances are present. However, in this translation new protuberances appeared. One appeared between the original first and second, and stand for the text segment in Tiger-Tiger when Mowgli went to the village. The peak is not too wide, but it is clearly there. The other new protuberance is between the original second and third and represents the search for the Sea Cow in The White Seal. These two new protuberances mean that we were able to find text segments in this newer German translation which are richer in vocabulary than the original English text. Haefs, 1987 has only three significant protuberances: at the description of King's Palace, Rikki-Tikki-Tavi, and Toomai of the Elephants. However, a closer analysis reveals other remarkable similarities. The missing protuberance from The White Seal is clearly there, but its peak is just below the significance level. Interestingly, the second peak of Harranth, 1987 is also detectable in this version.

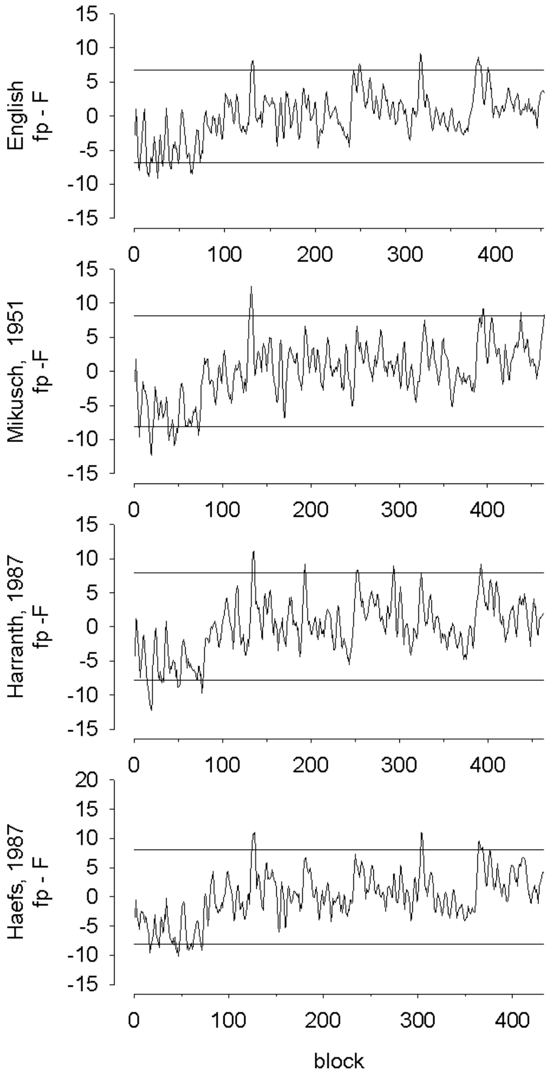


Fig 4 The newly introduced word types of The Jungle Book and its three German translations.

Finally, Mikusch, 1951 seems to be the furthest away in vocabulary from the original English text. While the first peak for the King's Palace is clearly there, the next two, although detectable, did not reach the significance level. Finally, the second of the text matches the fourth of the English text. In addition to these, new peaks appeared, which are unique to this translation: Kala Nag's rush and the song following the story, Shiv and the Grasshopper.

In general, Mikusch, 1951 was found somewhat arbitrary compared to the original English text. The other two translations were able to reproduce the vocabulary rich text segments of the original text much better. Harranth, 1987 goes ever beyond, the translator generated more vocabulary rich text segments than the original text has. Haefs, 1987 seems to be the closest to the original English text.

C. Texts and their different adaptations

Beyond the foreign language translations of novels the analyses of other adaptations of the texts might reveal characteristics which should have be taken care of during the adaptations process. In this project the condensed versions of novels were analyzed and compared to the original texts [12], [13]. Being familiar with the vocabulary rich segments of the texts might help the translators on the decisions which segments of the texts should be left out or shortened. These decisions highly depend on the target group. So being familiar with the advance of the vocabulary of a text might be a great advantage compared to subjective decisions.

On the other hand, the method proved to work on the decision in the direct origin of a second level adaptation of a text. This means that by deciding whether the analysis of the newly introduced word types we were able to show the condensed Hungarian translation of The Da Vinci Code originated from the full-length Hungarian translation or the condensed English adaptation. By the analysis of the advance of the vocabulary we were able to show that the condensed Hungarian text is derived from the condensed English text.

D. The comparison of the results of the analysis with the reviews

For the German translations of The Jungle Book we were able to find previously published reviews [15], [18]. Considering the vocabulary of the novels are these reviews in accordance with the results of our statistical analysis. These reviews state that the Mikush, 1951 translation carries vocabulary which were not meant in the original text [18]. The closest to the original is Haefs, 1987, while Harranth, 1987 gives a good approximation, but a special interpretation of the text resulting in a vocabulary which gives back the original vocabulary rich segments, but beyond that created new segments which are richer in vocabulary than the original text [15].

This example clearly show that being familiar with the position of the vocabulary rich segments of a the texts in advance to the translation process gives the translator information which segments of the texts requires greater attention because it is further away from the random selection of words than the rest of the text. Gives help in creating condensed versions of the text by finding the 'functionally useless' section of the text.

E. Hapax legomena in the texts

It was found that there is a strong correlation between the appearance of the newly introduced word types and the appearance of the hapax legomena, which means in general that if there is rise or a fall in the number of newly introduced word types the same true to the number of hapax legomena. However, rarely there are segments which carry a high number of newly introduced word types and fewer hapax legomena. These words are naturally reused in a later section of the text, and using Genette's expression [16], are less 'useless'. They have function in the text, their specialty is that they were just introduced in that particular block. Those segments which carry both high number of newly introduced word types and hapax legomena are the real 'functionally useless' segments of the text.

V. Summary

The method presented in this article is able to provide those segments of the texts which are richer in vocabulary than the robust part of the text. These segments carry information which is not bound strongly to the flow of the text. They usually give additional information about the settings, the historical background, the characters, or due to severe changes in style. These segments make the novels unique, but if we leave them out the text still would be completely understandable, so they are referred to as 'functionally useless' segments.

To find these vocabulary rich segments of the texts might help us in the analyses of the texts. Beyond that, it was proved that these objective data are able to provide preliminary information for translators by showing the vocabulary rich segments of the texts. Being aware in advance to the translation, or any adaptation of the texts of the position and the intensity of these vocabulary rich segments the translator might pay more attention to them.

VI. References

[1] R. H. Baayen, "The Randomness Assumption in Word Frequency Statistics," In Perissinotto, G. (ed), *Research in Humanities Computing* vol. 5. Oxford: Oxford University Press, 1996, pp. 17–31.

[2] R. H. Baayen, "The Effect of Lexical Specialization on the Growth Curve of the Vocabulary," *Computational Linguistics* vol. 22, pp. 455–480.

[3] R. H. Baayen, *Word Frequency Distributions*. Kluwer Academic Publishers, Dordrecht, Netherlands, 2001.

[4] I. Bart, and K. Klaudy, (ed.) *A fordítás tudománya*. Tankönyvkiadó, Budapest, 1985.

[5] R. Beaugrande, de and W. Dressler, *Introduction to text linguistics*. Bevezetés a szövegnyelvészetbe. Siptár, Péter. (trans.) (2000) Corvina, Budapest, 1981.

[6] M. Csernoch, "Természetes nyelvi szövegek összehasonlítása elsőrendű statisztikai modellekkel," *Publicationes Universitatis Miskolcinensis, Sectio Philosophica, Tomus X. – Fasciculus 3*. Miskolc 2005.

[7] M. Csernoch, "The introduction of word types and lemmas in novels, short stories and their translations," http://www.allc-ach2006.colloques.paris-sorbone.fr/DHs.pdf *Digital Humanities 2006. The First International Conference of the Alliance of Digital Humanities Organisations*. (5–9 July 2006, Paris), 2006.

[8] M. Csernoch, "Frequency-based Dynamic Models for the Analysis of English and Hungarian Literary Works and Coursebooks for English as a Second Language," *Teaching Mathematics and Computer Science*. Debrecen, Hungary, 2006, pp. 53–70.

[9] M. Csernoch, "Seasonalities in the Introduction of Word-types in Literary Works," *Publicationes Universitatis Miskolcinensis, Sectio Philosophica, Tomus XI. – Fasciculus 3*. Miskolc 2006–2007, 11–34.

[10] M. Csernoch, "Dinamikusan kezelhető statisztikai modellek irodalmi művek szóalakjainak vizsgálatára," *Alkalmazott Matematikai Lapok* vol. 24 (2007), 2007, pp. 57–77.

[11] M. Csernoch, “Newly introduced word-types and lemmas in Dan Brown’s The Da Vinci code and its translations,” *Across Languages and Cultures* vol. 8 (2), 2007, pp. 195–220.

[12] M. Csernoch, “Condensed versions of literary works,” In *When grammar minds language and literature*. University of Debrecen, 2007d, pp. 107–118.

[13] M. Csernoch, “Újonnan bevezetett szóalakok és lemmák Dan Brown The Da Vinci Code című művében és fordításaiban,” *Fordítástudomány*. 10, 2008, pp. 18–41.

[14] M. Csernoch, A novel way for the comparative analysis of adaptations based on vocabulary rich text segments: the assessment of Dan Brown’s The Da Vinci Code and its translations. *Digital Humanities 2008*. pp. 95–96. http://www.ekl.oulu.fi/dh2008/Digital%20Humanities%202008%20Book%20of%20Abstracts.pdf

[15] B. Danken, “Kiplings unsterblicher Klassiker.“ *DIE ZEIT*, 06.11.1987 Nr. 46, 1987.

[16] G. Genette, *Narrative Discours*e. Cornell University Press, Ithaca, New York, 1995, pp. 165.

[17] B. Hajtman, *Bevezetés a matematikai statisztikába*. Akadémiai Kiadó Budapest, 1971.

[18] W. Harranth, “Das Dschungelbuch. Nachwort,” Aus dem Englischen von Wolf Harranth (1987). Cecilie Dressler Verlag GmbH & KG, Hamburg, 2004, pp. 217–219.

[19] Gy. Meszéna, and M. Ziermann, *Valószínűség elmélet és matematikai statisztika*. Közgazdasági és Jogi Könyvkiadó, Budapest, 1981.

[20] MorphoWord http://www.morphologic.hu/index.php?option=com_virtuemart&Itemid=320&flypage=shop.flypageTab&page=shop.product_details&product_id=133&lang=en (June 1, 2010)

[21] Cs. Oravecz, and P. Dienes, “Large scale morphosyntactic annotation of the Hungarian National Corpus,” In Béla Hollósi and Judit Kiss-Gulyás (eds) *Studies in Linguistics*, vol. VI., Debrecen, 2002, pp. 277–298.

[22] Cs. Oravecz, and P. Dienes, “Efficient Stochastic Part-of-Speech tagging for Hungarian” *In Proceedings of the Third International Conference on Language Resources and Evaluation*, Las Palmas, 2002, pp. 710–717.

[23] P. Rayson, Matrix: “A statistical method and software tool for linguistic analysis through corpus comparison,” PhD thesis, Lancaster University, 2003.

[24] P. Rayson, “Wmatrix: a web-based corpus processing environmen,”. Computing Department, Lancaster University. http://www.comp.lancs.ac.uk/ucrel/wmatrix/, 2005.

[25] J. Ribycki, “Burrowing into Translation: Character Idiolects in Henryk Sienkiewicz’s Trilogy and its Two English Translations,” *Conference Abstract, The 16th Joint International Conference of the Association for Literary and Linguistic Computing and the Association for Computers and the Humanities* Göteborg University, Sweden, 2003.

[26] F. S. Simigné, *A fordítás mint közvetítés*. STÚDIÓ Rendezvények és Nyelvtanfolyamok, Miskolc, 2006.

[27] Gy. Solt, *Valószínűségszámítás*. Műszaki Könyvkiadó, Budapest, 1971.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 397–401

ISBN 978-83-60810-27-9
ISSN 1896-7094

SyMGiza++: A Tool for Parallel Computation of Symmetrized Word Alignment Models

Marcin Junczys-Dowmunt
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
ul. Umultowska 87, 61-614 Poznań, Poland
Email: junczys@amu.edu.pl

Arkadiusz Szał
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
ul. Umultowska 87, 61-614 Poznań, Poland
Email: arekszal@amu.edu.pl

***Abstract*—SyMGiza++ — a tool that computes symmetric word alignment models with the capability to take advantage of multi-processor systems — is presented. A series of fairly simple modifications to the original IBM/Giza++ word alignment models allows to update the symmetrized models between each iteration of the original training algorithms. We achieve a relative alignment quality improvement of more than 17% compared to Giza++ and MGiza++ on the standard Canadian Hansards task, while maintaining the speed improvements provided by MGiza++'s capability of parallel computations.**

I. Introduction

WORD alignment is a key component of the training procedure for statistical machine translation systems. The classic tool used for this task is Giza++ [1] which is an implementation of the so-called IBM Models 1-5 [2], the HMM model by [3] and its extension by [1], and Model 6 [1].

All these models are asymmetric, i.e. for a chosen translation direction, they allow for many-to-one alignments, but not for one-to-many alignments. Training two models in opposite directions and symmetrizing the resulting word alignments is commonly employed to improve alignment quality and to allow for more natural alignments. The two alignment models are trained fully independently from each other. Symmetrization is then performed as a post-processing step. Previous work [4], [5] has shown that the introduction of symmetry during training results in better alignment quality than post-training symmetrization.

The approaches from [4], [5] as well as our method still require the computation of two directed models which use common information during the training. Employing a multi-processor system for the parallel computation of theses models is a natural choice. However, Giza++ was designed to be single-process and single-thread. MGiza++ [6] is an extension of Giza++ which allows to start multiple threads on a single computer.

We therefore choose to extend MGiza++ with the capability to symmetrized word alignments models to tackle both problems in one stroke. The resulting tool SyMGiza++ is described in this work. The paper will be organized as follows: Section 2 provides a short overview of Giza++ and MGiza++ and the above mentioned methods of symmetrized alignment model training. In Sec. 3 we give a formal description of our modifications introduced into the classical word alignment models implemented in Giza++ and MGiza++. The evaluation methodology and results are provided in Sec. 4. Finally, conclusions are presented in Sec. 5.

II. Previous Work

A. *Giza++ and MGiza++*

Giza++ implements maximum likelihood estimators for several statistical alignment models, including Model 1 through 5 described by [2], a HMM alignment model by [3] and Model 6 from [1]. The EM [7] algorithm is employed for the estimation of the parameters of the models. During the EM algorithm two steps are applied in each iteration: in the first step, the E-step, the previously computed model or a model with initial values is applied to the data. The expected counts for specific parameters are collected using the probabilities of this model. In the second step, the M-step, these expected counts are taken as fact and used to estimate the probabilities of the next model. A correct implementation of the E-step requires to sum over all possible alignments for one sentence pair. This can be done efficiently for Model 1 and 2, and using the Baum-Welch algorithm also for the HMM alignment model [1].

For Models 3 through 6, a complete enumeration of alignments cannot be accomplished in a reasonable time. This can be approximated by using only a subset of highly scored alignments. In [2] it has been suggested to use only the alignment with the maximum probability, the so-called Viterbi alignment. Another approach resorts to the generation of a set of high probability alignments obtained by making small changes to the Viterbi alignment. [8] proposed to use the neighbour alignments of the Viterbi alignment.

MGiza++ [6] is a multi-threaded word alignment tool that utilizes multiple threads to speed up the time-consuming word alignment process. The implementation of the word alignment models is based on Giza++ and shares large portions of source code with Giza++. The main differences rely on multiple thread management and the synchronization of the counts collecting process. Similarly, our tool in turn incorporates large portions of the MGiza++ source code extending MGiza++'s capabilities of using multiple processors with the ability to compute symmetrized word alignment models in a multiprocessor environment. Since the multiprocessing aspect is mainly

978-83-60810-27-9/09/$25.00 © 2010 IEEE

a feature of the original MGiza++, we will not discuss it in this paper and refer the reader to the original paper on MGiza++ [6].

B. Symmetrized Word Alignment Models

The posteriori symmetrization of word alignments has been introduced by [1]. This method does not compute symmetrized word alignment models during the training procedure, but uses heuristic combination methods after the training. We described it in more detail in Sec. III-E. The best results of [1] for the Hansards task are 9.4% AER (using Model 4 in the last training iterations) and 8.7% AER (using the more sophisticated Model 6).

[4] improve the IBM alignment models, as well as the Hidden-Markov alignment model using a symmetric lexicon model. This symmetrization takes not only the standard translation direction from source to target into account, but also the inverse translation direction from target to source. In addition to the symmetrization, a smoothed lexicon model is used. The performance of the models is evaluated for Canadian Hansards task, where they achieve an improvement of more than 30% relative to unidirectional training with Giza++ (7.5% AER) is achieved.

In [9], the symmetrization is performed after training IBM and HMM alignment models in both directions. Using these models, local costs of aligning a source word and a target word in each sentence pair are estimated and graph algorithms are used to determine the symmetric alignment with minimal total costs. The automatic alignments created in this way are evaluated on the German–English Verbmobil task and the French–English Canadian Hansards task (6.6% AER).

Another unsupervised approach to symmetric word alignment is presented by [5]. Two simple asymmetric models are trained jointly to maximize a combination of data likelihood and agreement between the models. The authors restrict their experiments to IBM Models 1 and 2 and a new jointly trained HMM alignment model. They report an AER of 4.9% — a 29% reduction over symmetrized IBM model 4 predictions — for the Canadian Hansards task.

III. SyMGiza++ — Symmetrized MGiza++

In this section we will describe our modifications to the well known alignment models from [2] and [1].

We do not introduce changes to the main parameter estimation procedure. Instead, we modify the counting phase of each model to adopt information provided by both directed models simultaneously. The parameter combination step is executed in the main thread. In the following subsections, the formal aspects of the parameter combination will be outlined separately for each model. The notation has been adopted from [2] and we refer the reader to this work for details on the original models that will not be repeated in this paper.

A. Model 1

Model 1 is the first of the IBM models described extensively by [2] which have been implemented accurately in Giza++ and MGiza++.

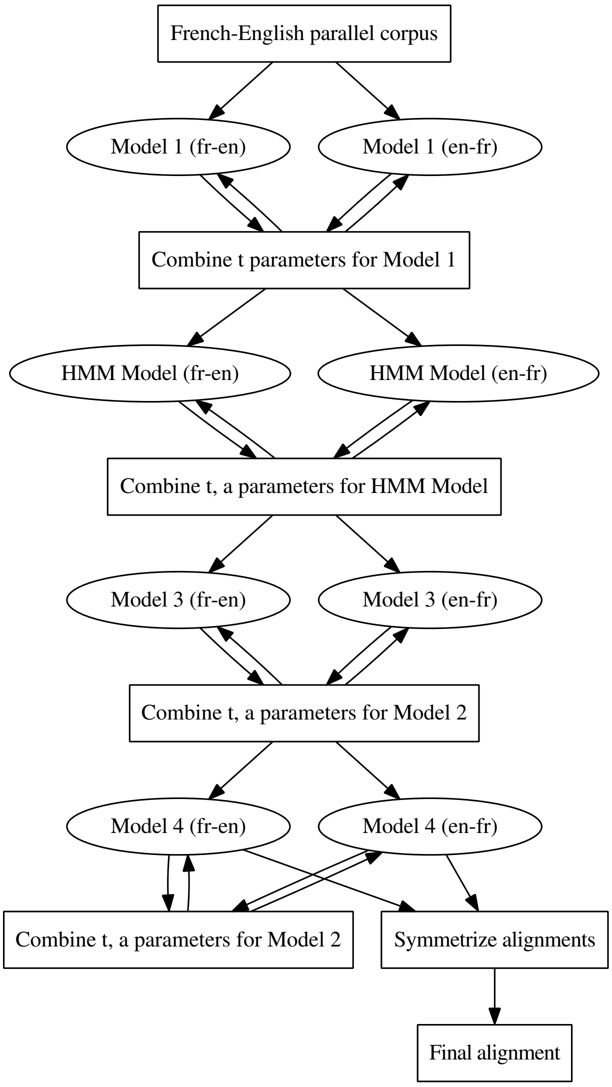

French-English parallel corpus
Model 1 (fr-en)
Model 1 (en-fr)
Combine t parameters for Model 1
HMM Model (fr-en)
HMM Model (en-fr)
Combine t, a parameters for HMM Model
Model 3 (fr-en)
Model 3 (en-fr)
Combine t, a parameters for Model 2
Model 4 (fr-en)
Model 4 (en-fr)
Combine t, a parameters for Model 2
Symmetrize alignments
Final alignment

Fig. 1. General training scheme for SyMGiza++

In order to distinguish between the parameters of the two simultaneously computed alignment models we will use α and β as subscripts for the parameters of the first and second model respectively. For our English-French training corpus we compute the following two models:

$$Pr_\alpha(\mathbf{f}|\mathbf{e}) = \frac{\epsilon(m|l)}{(l+1)^m} \sum_{\mathbf{a}} \prod_{j=1}^{m} t_\alpha(f_j|e_{a_j}) \tag{1}$$

$$Pr_\beta(\mathbf{e}|\mathbf{f}) = \frac{\epsilon(l|m)}{(m+1)^l} \sum_{\mathbf{b}} \prod_{i=1}^{l} t_\beta(e_i|f_{b_i}) \tag{2}$$

where l and m are the lengths of the French sentence $\mathbf{f}$ and the English sentence $\mathbf{e}$ respectively, $\mathbf{a}$ and $\mathbf{b}$ are the directed alignments between the sentences and t_α and t_β the directed *translation probabilities* between the French and English words f and e. Due to the simplicity of this model, it is straightforward to introduce our changes in the counting method used during the E-step of the EM-algorithm. The only free parameters of Model 1 are the translation probabilities t_α and t_β which are estimated by:

$$t_\alpha(f|e) = \frac{\sum_{s=1}^{S} c(f|e; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}{\sum_{f'} \sum_{s=1}^{S} c(f'|e; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}, \tag{3}$$

where S is the number of sentences in the parallel training corpus. $c(f|e; \mathbf{f}, \mathbf{e})$ is the expected count of times the words f and e form translations in the given sentences $\mathbf{f}$ and $\mathbf{e}$, in the inverted model $c(e|f; \mathbf{e}, \mathbf{f})$ is used.

In the original model, the expected counts $c(f|e; \mathbf{f}, \mathbf{e})$ are calculated from the t values of the preceding iteration with the help of the following two formulas:

$$c(f|e; \mathbf{f}, \mathbf{e}) = \sum_{\mathbf{a}} Pr_\alpha(\mathbf{a}|\mathbf{f}, \mathbf{e}) \sum_{i,j} \delta(f, f_j)\delta(e, e_i), \tag{4}$$

and

$$Pr_\alpha(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \frac{\prod_{j=1}^{m} t_\alpha(f_j|e_{a_j})}{\sum_{\mathbf{a}} \prod_{j=1}^{m} t_\alpha(f_j|e_{a_j})}, \tag{5}$$

where δ is the Kronecker function[1]. Equations (3) and (4) are common for all models discussed in this section. Our modifications are restricted to (5) which is replaced by

$$\begin{aligned} Pr_\alpha(\mathbf{a}|\mathbf{f}, \mathbf{e}) &= \frac{\prod_{j=1}^{m} \bar{t}(f_j, e_{a_j})}{\sum_{\mathbf{a}} \prod_{j=1}^{m} \bar{t}(f_j, e_{a_j})} \\ &= \frac{\prod_{j=1}^{m} \left(t_\alpha(f_j|e_{a_j}) + t_\beta(e_{a_j}|f_j)\right)}{\sum_{\mathbf{a}} \prod_{j=1}^{m} \left(t_\alpha(f_j|e_{a_j}) + t_\beta(e_{a_j}|f_j)\right)} \end{aligned} \tag{6}$$

Here we see the only difference between the standard Model 1 and our symmetrized version. By taking into account the translation probabilities from the previous iteration of both directed models we inform each model about the estimates of its counterparts. The following intuition applies: a French word is a good translation of an English word, if the English word is a good translation of the French word as well. This cannot be easily captured in the directed models without breaking up its sound probabilistic interpretation, as it happens here. However, since we modify only the way expected counts are obtained, the requirement imposed by [2] that

$$\sum_f t(f|e) = 1$$

still applies. Our modifications do not interfere with the EM procedure. The parameters for the inverted model are obtained analogously.

It should be noted that most of the time — despite the symmetry of the sum $\tilde{t}_\alpha(f|e) + \tilde{t}_\beta(e|f)$ occurring in both counts — $c(f|e; \mathbf{f}, \mathbf{e})$ and $c(e|f; \mathbf{e}, \mathbf{f})$ will have different values for the same words and sentences. This is due to the differences in the alignment direction. Therefore $t_\alpha(f|e) \neq t_\beta(e|f)$ in the general case.

B. Model 2

Although it is common practice to replace Model 2 during the training procedure with the HMM Model described in the next subsection, we need to modify its counting procedure as well. Model 2 is used to score a subset of alignments during the training procedure of the more sophisticated Models 3 and 4 which — in contrast to the lower models — cannot efficiently enumerate all possible alignments.

Model 2 introduces a second type of free parameters: the *alignment probabilities* a. These a parameters capture the probability that given the lengths of both sentences, a French word at position j is aligned with an English word at position a_j. The complete model is given by [2] as:

$$Pr_\alpha(\mathbf{f}|\mathbf{e}) = \epsilon(m|l) \sum_{\mathbf{a}} \prod_{j=1}^{m} \left(t_\alpha(f_j|e_{a_j}) a_\alpha(a_j|j, m, l)\right) \tag{7}$$

The general scheme described in (3) and (4) for the estimation of t values is the same for Model 2 as for Model 1. The alignment probabilities are estimated similarly:

$$a_\alpha(i|j, m, l) = \frac{\sum_{s=1}^{S} c(i|j, m, l; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}{\sum_{i'} \sum_{s=1}^{S} c(i'|j, m, l; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}, \tag{8}$$

$$c(i|j, m, l; \mathbf{f}, \mathbf{e}) = \sum_{\mathbf{a}} Pr_\alpha(\mathbf{a}|\mathbf{f}, \mathbf{e}) \delta(i, a_j). \tag{9}$$

Again, we only modify $Pr(\mathbf{a}|\mathbf{f}, \mathbf{e})$ in (4) and (9) to obtain our symmetrized version of the alignment models:

$$Pr_\alpha(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \frac{\prod_{i=1}^{m} \left(\bar{t}(f_j, e_{a_j}) \bar{a}(a_j, j, m, l)\right)}{\sum_{\mathbf{a}} \prod_{j=1}^{m} \left(\bar{t}(f_j, e_{a_j}) \bar{a}(a_j, j, m, l)\right)} \tag{10}$$

where $\bar{t}(f, e)$ is defined as before for Model 1 and $\bar{a}(i, j, m, l) = a_\alpha(i|j, m, l) + a_\beta(j|i, l, m)$. The effect of information sharing between the two inverted models Pr_α and Pr_β is even increased for Model 2 since translation and alignment probabilities interact during the estimation of both types of parameters for the next iteration.

C. HMM Model

The HMM Alignment Model has been introduced by [3] and is used in the GIZA++ family of alignment tools as a replacement for the less effective Model 2. The HMM alignment model is given by the following formula which at first looks very similar to (7):

$$P_\alpha(\mathbf{f}|\mathbf{e}) = \epsilon(m|l) \sum_{\mathbf{a}} \prod_{j=1}^{m} \left(t_\alpha(f_j|e_{a_j}) a_\alpha(a_j|a_{j-1}, l)\right) \tag{11}$$

The alignment probabilities from Model 2, however, are replaced by a different type of alignment probabilities. Here the probability of alignment a_j for position j has a dependence on the previous alignment a_{j-1} which turns the alignment model into a first order Markov model. The counts for the new a parameter are defined as follows:

$$a_\alpha(i|i', l) = \frac{\sum_{s=1}^{S} c(i|i', l; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}{\sum_{i''} \sum_{s=1}^{S} c(i''|i', l; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}, \tag{12}$$

[1] $\delta(i, j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$.

$$c(i|i',l;\mathbf{f},\mathbf{e}) = \sum_{\mathbf{a}} Pr_\alpha(\mathbf{a}|\mathbf{f},\mathbf{e}) \sum_j \delta(i',a_{j-1})\delta(i,a_j) \quad (13)$$

The definition of the t parameter and corresponding counts remains the same as for Model 1 and 2. Like before we only have to modify the definition of $Pr(\mathbf{a}|\mathbf{f},\mathbf{e})$:

$$Pr_\alpha(\mathbf{a}|\mathbf{f},\mathbf{e}) = \frac{\prod_{j=1}^{m} t_\alpha(f_j|e_{a_j}) a_\alpha(a_j|a_{j-1},l)}{\sum_{\mathbf{a}} \prod_{j=1}^{m} t_\alpha(f_j|e_{a_j}) a_\alpha(a_j|a_{j-1},l)} \quad (14)$$

is replaced by

$$Pr_\alpha(\mathbf{a}|\mathbf{f},\mathbf{e}) = \frac{\prod_{i=1}^{m} \left(\bar{t}(f_j,e_{a_j}) a_\alpha(a_j|a_{j-1},l)\right)}{\sum_{\mathbf{a}} \prod_{j=1}^{m} \left(\bar{t}(f_j,e_{a_j}) a_\alpha(a_j|a_{j-1},l)\right)}. \quad (15)$$

$\bar{t}$ is defined as before for Model 1 and 2.

Here, the alignment probabilities a remain unchanged. For Model 2 we are able to find the symmetrically calculated a parameters just by swapping source and target values. Doing the same for the Markov model would change the interpretation of the alignment probabilities. We would require neighbouring source language words to be aligned only with neighbouring target language words which is to strong an assumption. Nevertheless, their values are still influenced by both models due to the appearance of $\bar{t}$ in the re-estimation.

D. Model 3 and 4

We already mentioned that the parameters specific for Models 3 and 4 are calculated from fractional counts collected over a subset of alignments that have been identified with the help of the Viterbi alignments calculated by Model 2. Therefore it is not necessary to revise the parameter estimation formulas for Model 3 and 4, instead we simply adopt the previous changes made for Model 2. This influences the parameters of Model 3 and 4 indirectly by choosing better informed Viterbi alignments during each iteration.

E. Final Symmetrization

Although the two directed models influence each other between each iteration, the two final alignments produced at the end of the training procedure differ due the restrictions imposed by the models. Alignments are directed and since alignments are functions, there are no one-to-many or many-to-many alignments for the respective directions. There are, however, many-to-one alignments. [1] have proposed a method for the symmetrization of alignment models, which they call *refined symmetrization* and which is reported to have a positive effect on alignment quality.

They first map each directed alignment into a set of alignment points and create a new alignment as the intersection of these two sets. Next, they iteratively add alignment points (i,j) from the union of the two sets to the newly created alignment occurring only in the first alignment or in the second alignment if neither f_j nor e_i has an alignment in the new alignment, or if both of the following conditions hold:

- The alignment (i,j) has a horizontal neighbour$(i-1,j)$, $(i+1,j)$ or a vertical neighbour $(i,j-1)$, $(i,j+1)$ that is already in the new alignment.
- Adding (i,j) to the new alignment does not created alignments with both horizontal and vertical neighbours.

This method is applied as the final step of our computation and will also be applied to the directed alignments created by Giza++ and MGiza++, our baseline systems.

TABLE I
RESULTS FOR THE HLT/NAACL 2003 TEST SET

Alignment Method	Time [m]	Prec [%]	Rec [%]	AER [%]
GIZA++ EN-FR	–	91.19	92.20	8.39
GIZA++ FR-EN	–	91.82	87.96	9.79
GIZA++ REFINED	457	93.24	92.59	7.02
MGIZA++ EN-FR	–	91.19	92.22	8.40
MGIZA++ FR-EN	–	91.84	87.96	9.78
MGIZA++ REFINED	306	93.25	92.60	7.01
SYMGIZA++	332	94.34	94.08	**5.76**

IV. EVALUATION

We compare three systems on the same training and test data: Giza++, MGiza++, and SyMGiza++. For the Giza++ and MGiza++ we run both directed models separately and in parallel and recombine the resulting final alignments with the refined method described in III-E. The tools from the Giza++ family are all run with the following training scheme: $5\times$Model 1, $5\times$HMM Model, $5\times$Model 3 and $5\times$Model 4. All experiments were performed on a test system with 4 CPUs and 8 GB RAM, we plan to increase the number CPUs in the future. Apart from Giza++ all tools make use of all available CPUs, the parallel computation of the alignment model with Giza++ can employ at most two CPUs.

A. Measures and Evaluation Data

The standard metric *Alignment Error Rate* (AER) proposed by [1] is used to evaluate the quality of the introduced input word alignments. AER is calculated as follows:

$$\text{Precision} = \frac{|A \cap P|}{|A|} \qquad \text{Recall} = \frac{|A \cap S|}{|S|}$$
$$\text{AER} = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|} \quad (16)$$

where P is the set of possible alignment points in the reference alignment, S is the set of sure alignments in the reference alignment ($S \subset P$), and A is the evaluated word alignment.

In order to obtain results that can be easily compared with the work summarized in II-B, we evaluated our system on the Canadian Hansards task made available during the HLT-NAACL 2003 workshop on "Building and Using Parallel Texts: Data Driven Machine Translation and Beyond" [10]. The training data comprises 1.1M sentences from the Canadian Hansards proceedings and a separate test set of 447 manually word-aligned sentences provided by [1].

B. Results

Our results — which comprise alignment quality and processing time — are summarized in Tab. I. Processing time is measured from the beginning of processing till the end of the symmetrization process.

It is not surprising that there are no significant differences between Giza++ and MGiza++ when AER is considered. MGiza++, however, is about 33% faster than the two Giza++ processes run in parallel. MGiza++ is also slightly faster than SyMGiza++. This delay of SymGiza++ is caused by the parameter recombination executed between each model iteration and by the idle time if one directed model has to wait for its counterpart. SyMGiza++ achieves the best AER results with a relative improvement of more than 17% compared to Giza++ and MGiza++.

In Sec. II-B we gave the results for a number of other symmetrization approaches. Although we use the same test set our results are not yet fully comparable to the results of other works. We tried but failed to reproduce the results from [5] using the BerkeleyAligner, for which the authors reported an AER of 4.9%. The results reported by [5] for their base line alignments produced with Giza++, on the other hand, are more or less identical to our results. This requires further investigation and we will give a more comprehensive comparison of our results and the results in the literature in an extended paper to come.

V. Conclusions

We have presented SyMGiza++, a tool that computes symmetric word alignment models with the capability to take advantage of multi-processor systems. Our fairly simple modification to the well-known IBM Models implemented in Giza++ and MGiza++ achieves quite impressive improvements for AER on the standard Canadian Hansards task. Our symmetrized models outperform post-training symmetrization methods. On a four processor system, SyMGiza++ is slightly slower than MGiza++, but significantly faster than Giza++ executed in two parallel processes.

Acknowledgments

This paper is based on research funded by the Polish Ministry of Science and Higher Education (Grant No. 003/R/T00/2008/05).

References

[1] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models." *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[2] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[3] S. Vogel, H. Ney, and C. Tillmann, "Hmm-based word alignment in statistical translation," in *Proceedings of ACL*, 1996, pp. 836–841.

[4] R. Zens, E. Matusov, and H. Ney, "Improved word alignment using a symmetric lexicon model," in *Proceedings of ACL-COLING*, 2004, p. 36.

[5] P. Liang, B. Taskar, and D. Klein, "Alignment by agreement," in *Proceedings of ACL-COLING*, 2006, pp. 104–111.

[6] Q. Gao and S. Vogel, "Parallel implementations of word alignment tool," in *Proceedings of SETQA-NLP*, 2008, pp. 49–57.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistcial Society, series B*, vol. 39, no. 1, pp. 1–38, 1977.

[8] Y. Al-Onaizan, J. Curin, M. Jahr, K. Knight, J. Lafferty, I. Melamed, F. Och, D. Purdy, N. Smith, and D. Yarowsky, "Statistical machine translation," JHU workshop, Tech. Rep., 1999. [Online]. Available: citeseer.ist.psu.edu/al-onaizan99statistical.html

[9] E. Matusov, R. Zens, and H. Ney, "Symmetric word alignments for statistical machine translation," in *Proceedings of ACL-COLING*, 2004, pp. 219–225.

[10] R. Mihalcea and T. Pedersen, "An evaluation exercise for word alignment," in *Proceedings of HLT-NAACL*, 2003, pp. 1–10.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Semi-Automatic Extension of Morphological Lexica

Tobias Kaufmann and Beat Pfister
Speech Processing Group
Swiss Federal Institute of Technology (ETH)
Zurich, Switzerland
Email: {kaufmann,pfister}@tik.ee.ethz.ch

***Abstract*—We present a tool that facilitates the efficient extension of morphological lexica. The tool exploits information from a morphological lexicon, a morphological grammar and a text corpus to guide the acquisition process. In particular, it employs statistical models to analyze out-of-vocabulary words and predict lexical information. These models do not require any additional labeled data for training. Furthermore, they are based on generic features that are not specific to any particular language. This paper describes the general design of the tool and evaluates the accuracy of its machine learning components.**

I. Introduction

MANY applications of natural language processing heavily rely on lexical resources. For highly inflected languages, such resources are typically represented as morphological lexica that contain stem entries, inflectional morphemes and derivational morphemes. The formation of words from morphemes is described by a morphological grammar.

In practical applications, lexical resources often need to be extended continually, either to include additional in-domain words or to expand the application to new domains. The manual extension of lexical resources is considered to be expensive. This is particularly true if lexical resources of different languages have to be maintained. In such cases, a trained native speaker may not be readily available.

One way to deal with this problem is to compromise on the quality of the lexicon. For example, out-of-vocabulary words can simply be added as full forms with some minimal specification, e.g. their part-of-speech. Such an approach is problematic for two reasons. First, the underspecification increases the ambiguity in later processing stages such as syntactic analysis or generation. Second, adding full forms is rather inefficient for highly inflected languages, where a single stem can account for many inflected forms and a potentially infinite number of compound words. The latter problem can be partially solved by stripping off the suffix of the out-of-vocabulary word (either manually or by means of heuristics) and allowing the resulting pseudo-stem to undergo arbitrary inflection. This, however, will again increase the ambiguity.

In this paper, we present a tool that facilitates the efficient extension of high-quality morphological lexica, even for a user with minimal knowledge of the underlying linguistic representations and conventions. The tool assists the user in creating the stem entry that is required to correctly analyze a given out-of-vocabulary word. The acquisition process is guided by statistical models whose training requires no additional data apart from a text corpus and some language-specific parameters. The models are based on a set of generic features and thus can be trained for arbitrary languages.

The paper is structured as follows. Section II describes the concepts underlying our tool and illustrates a typical user interaction. Section III is concerned with the statistical models, their training procedures and their accuracy. Related work is reviewed in Section IV and Section V concludes with a final discussion.

II. Tool Design

A. Basic Assumptions

Before describing the actual tool, we want to make certain assumptions explicit. Regarding the linguistic processing, we assume that the morphological grammar is a context-free grammar with atomic-valued attributes attached to the non-terminal symbols. In this setup, a stem entry consists of an orthographic representation of the stem, a preterminal symbol and a value for each attribute. The attributes of the preterminal symbol denote syntactic features (e.g. grammatical gender for German nouns) or inflectional information such as the inflectional class. For example, the stem entry for the German noun *Tür* (*door*) can be defined as follows:

```
NS(sk10,pk4,f) "tür+"
```

The first two attributes denote the inflectional classes for singular and plural inflection, and the third attribute indicates feminine gender. The orthographic representation of the stem is terminated by the morpheme boundary marker +.

Morphological processing may also involve a finite-state transducer (FST). FSTs essentially map the lexical form of a morpheme to one or more surface forms depending on the graphemic context. For example, the surface form of the German verb *handeln* (*to trade*) can be specified as `handel+`. The FST ensures that this lexical form has the surface realization *handl* if followed by an *e* but *handel* anywhere else. Our tool can deal with FSTs by transforming surface forms back to possible lexical forms.

A final assumption is that an out-of-vocabulary word contains exactly one unknown stem. To put it differently, our tool will always look for a single stem that allows to analyze a given out-of-vocabulary word. If the word contains two or more unknown stems, the acquisition will result in a pseudo-stem which is itself a compound.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

B. Acquisition Process

The acquisition process consists of three steps. In the first step, an out-of-vocabulary word is entered and automatically analyzed.

In the second step, the stem's preterminal symbol and lexical form are determined. To this end, the tool shows all possible stem hypotheses that would render the given out-of-vocabulary word analyzable if added to the lexicon. The visualization of the stem hypotheses reflects their relative probabilities, which allows the user to focus on the most likely hypotheses. The user can reduce the candidate set by adding constraints on various levels of linguistic abstraction. The second step is completed by selecting a stem hypothesis.

The morphosyntactic features of the stem entry are determined in the third step. The tool facilitates this task by providing a probability for each potential attribute value. Further, the tool offers different ways for verifying and correcting the predictions.

As soon as the selected feature values are confirmed, a new stem entry is generated and added to a special lexicon. The second and the third step will be illustrated and further detailed in the following sections.

C. Determining the Stem

Figure 1 shows the state of the dialog at the beginning of the second step. Panel 1 visualizes the out-of-vocabulary word under consideration (in the present example the German noun *Dressursport*, engl. *dressage sports*). The possible stem hypotheses are listed beneath this word in the order of their respective probability. The hypothesis on the top, the noun stem with lexical form `dress+`, is the most likely one. In order to help the user in verifying the top-ranked hypothesis, panel 3 lists the five most frequent words that can be analyzed with the respective stem.

In this example, the correct hypothesis (the noun stem `dressur+`) is ranked second. A user which is familiar with the given lexical representation can simply click on the correct stem and thereby proceed to the third step. If the user is less trained or if the correct stem is not among the best few hypotheses, the candidate set can be reduced in three ways.

1) Related words: Panel 5 lists words that can be analyzed with one of the available stem hypotheses *but not* with the top-ranked hypothesis. By clicking on a word, the candidate set is restricted to the stems which would allow to analyze that word. For example, clicking on *dressur* (*dressage*) results in a set of four stem hypotheses, the top-ranked hypothesis being the correct one. The word constraint is added to panel 4 and can be removed with another click. Note that this interaction requires no linguistic knowledge besides some sense of semantic relatedness.

The list of related words is constructed as follows. Starting with the second-ranked hypothesis, each stem hypothesis contributes up to three words that discriminate it from the top-ranked hypothesis. If there are more than three such words, only the three most frequent ones (in terms of corpus occurrences) are considered. Thus, the words at the top of the list correspond to the most probable stem hypotheses.

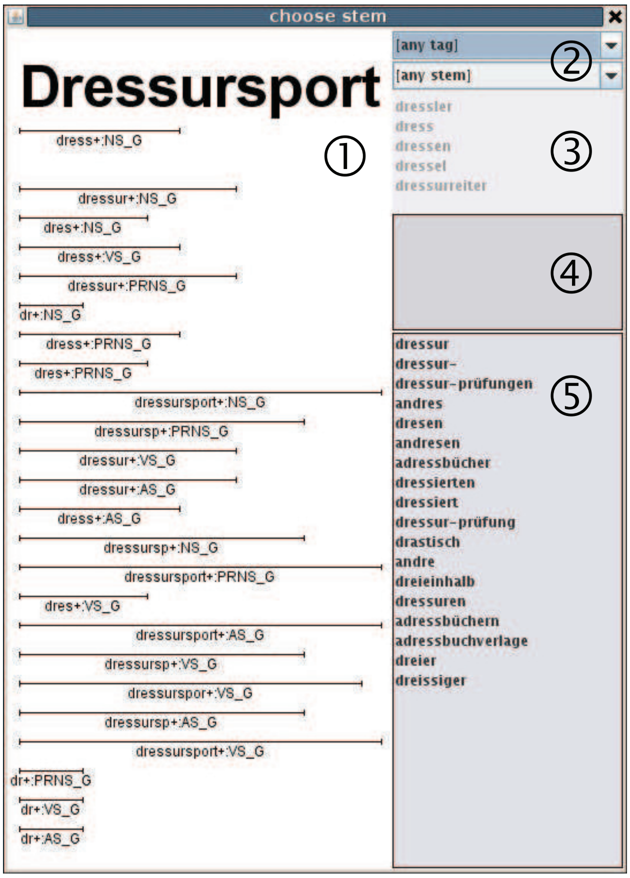
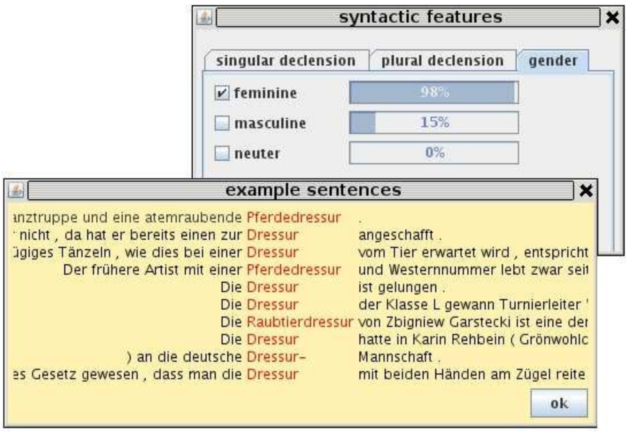


Fig. 1. The dialog for determining the preterminal and the lexical form of the stem. The stem hypotheses shown on the left are ordered by probability, the topmost hypothesis being the most likely one.

2) Stem span: By clicking on a letter of the out-of-vocabulary word in panel 1, the user can specify whether this particular letter should be included in the stem or not. As for the related words, the candidate set is reduced to include only stem hypotheses that fulfill this constraint. For example, clicking on the second letter *'r'* results in a set of 13 candidates. Again, the top-ranked hypothesis happens to be the correct one.

This kind of interaction requires the user to be familiar with the particular notion of a stem that is assumed in the morphological lexicon. This notion may not be intuitive. For example, the lexical stem may differ from the surface stem due to the FST, or parts of the "actual" stem may be represented by special morphemes in order to account for allomorphic variation.

3) Tag and preterminal: A final way of restricting the set of stem hypotheses is to specify the tag or the preterminal symbol in panel 2. The tag is the non-terminal symbol that represents the out-of-vocabulary word on the sentence level. Tags typically correspond to parts-of-speech such as noun, verb or adjective. Even though the tool is able to predict the



☑ feminine 98%
□ masculine 15%
□ neuter 0%

```
anztruppe und eine atemraubende Pferdedressur    .
 nicht , da hat er bereits einen zur Dressur     angeschafft .
ügiges Tänzeln , wie dies bei einer Dressur      vom Tier erwartet wird , entspricht
      Der frühere Artist mit einer Pferdedressur und Westernnummer lebt zwar sei
                                 Die Dressur     ist gelungen .
                                 Die Dressur     der Klasse L gewann Turnierleiter '
                                 Die Raubtierdressur von Zbigniew Garstecki ist eine der
                                 Die Dressur     hatte in Karin Rehbein ( Grönwohlc
                  ) an die deutsche Dressur-     Mannschaft .
es Gesetz gewesen , dass man die Dressur         mit beiden Händen am Zügel reite
```

Fig. 2. The dialog for determining the morphosyntactic features. The window in front presents corpus examples which indicate that the noun stem *dressur* has feminine gender.

part-of-speech of a word (see Section III-B), this information is currently not used to restrict the candidate sets. The preterminal symbol denotes the category of the stem, e.g. noun stem, verb stem or adjective stem.

D. Determining the Morphosyntactic Features

The dialog for determining the morphosyntactic features is shown in Figure 2. There is a tab for each morphosyntactic feature. A tab displays information for each possible value of its corresponding feature. In particular, it indicates the probability that a certain value is present in the given stem. These probabilities are computed by binary classifiers (one classifier per value) and thus do not sum to 1. The use of binary classifiers is motivated by the fact that values need not be mutually exclusive. For example, the German noun *filter* (engl. *filter*) can have either masculine or neuter gender.

Initially, the most probable value of each tab is checked. If the user agrees with this choice, the tool can directly generate the final stem entry. If not, the selection can be changed by checking or unchecking individual values. The validation and correction of the automatically predicted values is assisted in two ways:

First, the tool can present corpus examples which suggest that a particular value should be chosen. Figure 2 shows 10 corpus examples supporting the hypothesis that *Dressur* is feminine (which it is). The most informative example is shown on the top of the list. In this example, the best indicator for feminine gender seems to be the unambiguously feminine determiner *eine* followed by a word that ends in *-e*. The prototypical female determiner *die* is less discriminative as it can also appear with plural nouns of any gender.

Corpus examples tend to be less useful for verifying inflectional features. For instance, if the best predictor for some inflectional class is the grammatical gender, the tool will present corpus examples which indicate that gender. This information clearly doesn't help the verification. Thus, the tool can visualize the current inflectional paradigm as tables and lists of inflected forms. This layout of this visualization is defined as a grammar and language specific template that has to be created manually. The user can also introduce constraints by restricting the inflected forms for specific syntactic contexts. As a result, the incompatible values will be disabled in the dialog.

III. Machine Learning

The presented tool is based on machine learning components that compute the probability of a stem hypothesis and predict the morphosyntactic features. The underlying models are automatically created from a morphological lexicon, a morphological grammar, a normalized text corpus and some grammar-specific information. In the following, we will first comment on the latter two resources and then describe the main components. Finally, the accuracy of the models will be evaluated.

A. Resources

The most important additional resource is a sufficiently large text corpus. The corpus is assumed to be normalized such that there is exactly one token per line. A token is either a word or an interpunction symbol.

The tool also requires some explicit information about the given language and grammar. First of all, it needs a description of the stem morphemes that are to be acquired. Besides the preterminal symbol, a stem description specifies how each argument has to be handled: an argument can either be determined by the acquisition tool, set to a default value or remain unspecified. Next, the lexical form of a stem has to be described as a regular expression. The lexical form will typically be an arbitrary sequence of characters followed by a morpheme boundary symbol, e.g. `[a-zäöü]+[+]` in the case of our German grammar.

Finally, the explicit information also includes two sets of heuristics that allow for a very crude disambiguation. The first set relates a tag to its preferred heads. For example, German nouns can be derived from noun stems, adjective stems or verb stems, but the preferred head (in case of ambiguity) is the noun stem. The second set excludes certain tags for certain stems. For example, if an overly productive rule allows to create a proper name from each noun, it is hardly possible to train a part-of-speech tagger that can discriminate between these two. Thus, it can be declared that the proper name tag should be excluded if a word can be derived from a noun stem.

B. Part-of-Speech Tagging

Part-of-speech information can be applied on different levels of our approach. We have implemented an HMM part-of-speech tagger similar to the TnT tagger [1]. The tagger is based on a trigram sequence model and exploits suffix statistics for handling unknown words.

The tagger is trained on the text corpus by means of an expectation-maximization algorithm. For each analyzable word, the set of possible tags (i.e. non-terminal symbols) is defined by the grammar and the lexicon. We initially assume that the possible tags are uniformly distributed. The tag distributions of out-of-vocabulary words are estimated from the suffix statistics of the analyzable words. In each training iteration, the forward-backward algorithm is employed to compute the posterior tag probabilities of each word. This information is then used to re-estimate the model parameters. As the forward-backward algorithm is rather slow with logarithmic probabilities, we preferred a scaling approach to avoid underflow [2], [3].

The tagger is used to produce a tagged version of the text corpus. Further, the tag distribution of each word is computed as the average tag distribution for all corpus occurrences.

C. Predicting Stems

Predicting stem entries is considered to be a discriminative reranking task. First, we compute all stem hypotheses that would render the given out-of-vocabulary word analyzable. Next, each stem hypothesis is transformed into a stem entry with unspecified arguments. These additional entries allow to analyze the out-of-vocabulary word w, yielding a set of parse trees $\mathcal{T}(w) = \{t_1, t_2, ..., t_n\}$. A disambiguation model then assigns a conditional probability $P(t|\mathcal{T}(w))$ to each parse tree. The probability of a stem s is defined as

$$\tilde{P}(s|\mathcal{T}(w)) = \max_{t \in \mathcal{T}(w,s)} P(t|\mathcal{T}(w)), \tag{1}$$

where $\mathcal{T}(w,s)$ denotes the set of those parse trees that include the hypothetical stem entry s[1]. The distribution $P(t|\mathcal{T}(w))$ is described by a discriminative log-linear model [6], [7]:

$$P(t|\mathcal{T}(w)) = \frac{\exp(\sum_i \theta_i f_i(t))}{\sum\limits_{t' \in \mathcal{T}(w)} \exp(\sum\limits_i \theta_i f_i(t'))} \tag{3}$$

The real-valued features $f_i(t)$ represent pieces of evidence whose relative importance is expressed by the feature weights θ_i. In the present work, each feature $f_i(t)$ counts how often a certain linguistic event occurs in the parse tree t.

Table I shows the different types of linguistic events that were considered for stem prediction. Actual linguistic events are defined by replacing the arguments X, x, c, M, $i^\star$ or m_i with specific values. The character classes that are used to characterize stem hypotheses are automatically induced from the text corpus with a clustering algorithm [8]. Both 8 and 16 classes are considered in order to capture different levels of abstraction. The stem occurrence information is extracted by analyzing the 500 000 most frequent words in the corpus.

TABLE I
TYPES OF LINGUISTIC EVENTS FOR STEM PREDICTION

stem hypothesis (STH) features
STH has lexical form x
STH covers at most m surface characters ($1 \leq m \leq 10$)
STH contains the lexical character sequence x ($\lvert x\rvert \leq 4$)
STH contains the character class sequence x ($\lvert x\rvert \leq 4$)
STH has lexical prefix x ($\lvert x\rvert \leq 4$)
STH has lexical suffix x ($\lvert x\rvert \leq 4$)
STH has lexical prefix x and preterminal X ($\lvert x\rvert \leq 4$)
STH has lexical suffix x and preterminal X ($\lvert x\rvert \leq 4$)
STH has lexical prefix x and left surface context c ($\lvert x\rvert \leq 4$, $\lvert c\rvert \leq 3$)
STH has lexical suffix x and right surface context c ($\lvert x\rvert \leq 4$, $\lvert c\rvert \leq 3$)
parse tree features
tree contains terminal X
tree contains non-terminal X
tree contains rule X
tree contains lexicon entry X
tree contains morpheme X
tree contains morpheme with lexical form x
tree contains morpheme with surface form x
tree contains morpheme with lexical form x and preterminal X
tree contains morpheme with surface form x and preterminal X
morpheme has lex. prefix x and left surf. context c ($\lvert x\rvert \leq 4$, $\lvert c\rvert \leq 3$)
morpheme has lex. suffix x and right surf. context c ($\lvert x\rvert \leq 4$, $\lvert c\rvert \leq 3$)
capitalization features
word is capitalized and the parse tree's root non-terminal is X
word is not capitalized and the parse tree's root non-terminal is X
stem occurrence features
the corpus contains a word that can be decomposed into the morpheme sequence $m_1, ..., m_M$, where $m_{i^\star}$ is the stem hypothesis. Each morpheme m_i is specified by its preterminal symbol and (optionally) its lexical form.

Estimating the model parameters θ_i requires training examples, each example consisting of a correct parse tree and a set of incorrect parse trees. As the number of parse trees can be huge, a training example is generated by means of random subsampling [9]: we randomly choose one tree that contains the correct stem and 20 trees that contain any other stem. After generating 10 000 training examples for different words w, Charniak and Johnson's MaxEnt reranker [7] is used to compute the model parameters. Note that the potentially very large number of features requires some kind of feature selection. For this purpose, we only consider the first 2000 features that are chosen by a boosting learner [10].

At this point, we still require a list of out-of-vocabulary words for which the correct stem hypothesis is known. As no manually labeled data is available, the lexicon and the text corpus are used to artificially generate such data. This process is described in the following.

An in-vocabulary word w from the text corpus is morphologically analyzed and disambiguated by means of a simple heuristics: a parse tree is preferred if (in decreasing order of importance) its root non-terminal matches the most likely tag of w according to the tagger, if it contains less stems, if it contains more stems that are preferred heads of the root tag (see Section III-A), if it contains longer stems and if it is smaller in terms of tree nodes.

From the resulting unambiguous parse tree, a random stem is selected. This stem will be considered as the correct stem hypothesis. Next, the stem is removed from the lexicon, and

[1] The correct way of computing the stem probability may be considered to be

$$P(s|\mathcal{T}(w)) = \sum_{t \in \mathcal{T}(w,s)} P(t|\mathcal{T}(w)). \tag{2}$$

This probability can be efficiently computed from a packed parse forest representation [4] without explicitly enumerating all parse trees. However, we observed that the prediction of the correct stem hypothesis was significantly less accurate for this approach. In particular, we observed a strong bias towards stems that are more productive in generating parse trees (i.e. verb stems in our German grammar). We believe that this is due to the training procedure, which can be interpreted as maximizing some margin between the correct parse tree and the incorrect ones [5].

TABLE II
FEATURE TYPES FOR PREDICTING MORPHOSYNTACTIC FEATURES

word at position P is x $(-2 \leq P \leq 2)$
word at position P has suffix x $(\|x\| \leq 4,\ -2 \leq P \leq 2)$
stem contains the lexical character sequence x $(\|x\| \leq 4)$
tag at position P is X $(-2 \leq P \leq 2)$

TABLE III
STEM PREDICTION: EVALUATION

approach	*accuracy*
baseline	8%
no stem occurrence features, unconstrained	53%
no stem occurrence features, constrained	57%
stem occurrence features, unconstrained	68%
stem occurrence features, constrained	72%

so are all overlapping stems from alternative parse trees. In general, w will now be out-of-vocabulary, which allows to generate the stem hypotheses. Given the word w, the correct stem hypothesis and the set of incorrect stem hypotheses, a training example for the log-linear model can now be generated.

In order to reduce the mismatch between the training data and the actual out-of-vocabulary words, each (correct) stem occurs about equally often in the training data. This prevents the model from rote learning the properties of a few very frequent stems.

D. Predicting Morphosyntactic Features

Predicting a morphosyntactic feature involves one classifier for each possible value of the given feature. In our German system, for example, there is a classifier which predicts whether a given stem has feminine gender or not.

The decision of the classifier is based on a set of boolean features. Each feature can be thought of as scanning the text corpus and reporting whether the stem under consideration occurs in a specific context. For example, some feature may be true if and only if the corpus contains a word that can be analyzed with the given stem and is preceded by the word *eine*. The basic feature types are shown in Table II. The positions indicated by P are relative to the given stem occurrence, i.e. the word a position -1 immediately precedes the stem occurrence.

The actual classification process is as follows. First, a precomputed index is used to retrieve all corpus words which can be analysed as having the given stem as their rightmost stem. The rightmost stem is typically the head of a compound word. For languages with left-headed compounds (e.g. Vietnamese), the assumed head position can be changed accordingly. Next, a second index provides the positions where those words occur in the corpus. The aggregated value of a boolean feature is computed as the logical disjunction of the feature values for the different corpus occurrences.

The training of the classifiers is straight-forward: training examples can be generated by extracting the feature values for the known stems from the morphological lexicon, which also provide the correct label. As classifiers, we use maximum entropy models [11] with Gaussian priors for regularization [12]. We apply feature selection based on a mutual information criterion: the class label and the value of a particular feature can both be considered as random variables, where the random process consists of randomly choosing a stem occurrence in the corpus. The mutual information of these two random variables provides a measure of the feature's usefulness. For each classifier, we select the 500 features with the highest mutual information.

In addition to the basic features from Table II, we also use so-called joint features. A joint feature is an arbitrary conjunction of (possibly negated) basic features. For example, a joint feature might indicate that the word at position -2 is *die* and the word at position -1 does not end in *-e*. Joint features are created automatically by training an alternating decision tree classifier [13] to predict whether some morphosyntactic feature has a specific value or not. Rather than using information aggregated over the whole corpus, this classifier makes a prediction for a single stem occurrence. Each decision in the alternating decision tree depends on the value of a single basic feature, and a path from the root node to a so-called predictor node corresponds to a joint feature. We used JBoost [14] to create a decision tree from 10 000 labeled stem occurrences. JBoost is run for 100 training iterations which results in 200 joint features.

E. Evaluation

In order to evaluate the accuracy of stem prediction, we randomly chose 1000 out-of-vocabulary words from our German text corpus (220 million tokens of newspaper text). Words with typographic errors, foreign words, dialect words and closed-class words were removed from this set. Also, personal names and geographical names were not considered because they are massively underrepresented in our German lexicon. As it is assumed that compounds do not contain two or more unknown stems, such compounds were excluded as well. Finally, we removed compounds which our German grammar could not account for. For the remaining 530 words, the correct stems were determined manually.

Table III shows the resulting accuracies. The baseline is the expected accuracy when uniformly choosing a random stem hypothesis. The baseline is compared to four variants of our stem prediction algorithm. These variants differ in whether they make use of stem occurrence features (see last item of Table I) and whether tag constraints are applied.

With tag constraints, the best out-of-vocabulary stems are not directly determined according to $\tilde{P}(s|\mathcal{T}(w))$. Rather, the maximization in equation (1) is additionally restricted to parse trees whose root non-terminal matches the most likely tag of w according to the part-of-speech tagger. If the tagger does not make a prediction for w, the algorithm falls back on $\tilde{P}$. Unlike the disambiguation model for stem prediction, the tagger also considers the textual contexts of a word and thus provides additional disambiguating information.

TABLE IV
PREDICTION OF MORPHOSYNTACTIC FEATURES: EVALUATION

gender	*baseline*	*accuracy*
masculine	55%	91%
feminine	62%	96%
neuter	79%	93%
singular declension class	*baseline*	*accuracy*
none	85%	98%
class 1	64%	94%
class 2	73%	89%
class 3	82%	90%
class 4	94%	95%
plural declension class	*baseline*	*accuracy*
none	80%	92%
class 1	94%	98%
class 3	80%	97%
class 4	80%	96%
class 6	91%	98%
class 7	74%	94%

The results show that the presented approach to stem prediction is fairly accurate: the most likely stem hypothesis is correct in almost 3 out of 4 cases. They also suggest that judging stem hypotheses on the basis of corpus occurrence information (rather than just considering the word w in isolation) leads to a much higher accuracy. Using part-of-speech tagging to disambiguate parse trees seems to have a smaller impact. We did not use this information in our tool in order to keep the user interface intuitive.

The prediction of morphosyntactic features was evaluated on about 300 noun stems that were excluded from training. Table IV shows the results for the different values of gender, singular inflection class and plural inflection class. For brevity, the table lists only the values that are observed in at least 5% of the stems. For each value, the baseline accuracy and the classifier accuracy are provided. The baseline accuracy is achieved by assigning each example the majority class. It can be seen that the classifier accuracies are substantially higher than the baseline accuracies. Exceptions are very rare values, for which a negative decision is already very accurate.

IV. RELATED WORK

There are a number of approaches to extracting lexicon entries from text corpora [15], [16], [17], [18]. Here, a lexicon entry consists of a lemma or stem and an inflectional paradigm, though some approaches also consider syntactic features [16], [18]. All approaches explicitly describe word formation, either in the form of a morphological grammar [16] or some representation of inflectional paradigms [15], [17], [18].

The above approaches do not consider existing morphological lexica in the acquistion process. Rather, they employ heuristics which are mainly based on the presence or absence of inflected forms in the corpus. For example, Oliver et al. prefer a hypothetical lexicon entry if it accounts for more inflected forms observed in the corpus [18]. Adolphs additionally considers the coverage of the paradigm, i.e. the fraction of generated inflected forms that actually appear in the corpus [16]. Šnajder et al. compute the corpus frequencies of the generated forms and essentially use the sum of these frequencies as a measure of confidence. Finally, Forsberg relies on handcrafted constraints on the presence and absence of inflected forms [17].

Note that our approach also uses the presence of inflected form as evidence, both for predicting stems and for predicting morphosyntactic features. However, by exploiting the information in an existing morphological lexicon, we are able to compute a dedicated weight for each piece of evidence. Thus, inflected forms which are more reliable indicators have a higher influence on the prediction. The constraints used in Forsberg's approach [17] achieve a similar effect but have to be determined manually. Further, none of the above approaches attempts to analyze compounds: it is assumed that an unknown word is to be decomposed into a single stem and an inflectional ending.

Our approach to predicting morphosyntactic features is mainly related to some work on automatic lexical acquisition for precision grammars. Precision grammars require much richer lexical information than we are aiming at, e.g subcategorization frames. The most closely related work is that of Baldwin [19] and Nicholson et al. [20], who used k-nearest-neighbors classifiers to bootstrap lexical resources by means of corpus data. In contrast to the presented tool, they do not predict the stems of unknown words but assign lexical information to entire word forms. As in our approach, their features are based on word contexts, affixes and character n-grams. The former author additionally considers the output of a chunk parser and a dependency parser. We did not use such tools as they are not available for many languages.

V. DISCUSSION

We have presented a tool for the semi-automatic extension of morphological lexica. The tool facilitates the efficient acquisition of new stem entries from out-of-vocabulary words and ensures a high quality and consistency of the lexical resources.

1) *Efficiency*: Statistical models guide the user in the acquisition process and allow him to focus on the most promising hypotheses. The tool can also be handled by relatively untrained users: it employs concepts that are easily accessible to native speakers and its models capture some of the resource-specific "expert knowledge".
2) *Quality*: The user is offered different ways to verify his decisions, e.g. the predictions and the confidence of the statistical models, evidence from the text corpus and a visualization of the inflectional paradigm.
3) *Consistency*: Sometimes there are different possibilities to specify a stem entry. Sources of uncertainty may be the boundary between stem and affixes or the degree to which stems can include derivational morphemes. The statistical models can capture such tendencies in the original resources and may help to resolve ambiguities accordingly.

For our German resources, we have demonstrated that the machine learning components are able to predict lexical information with a reasonably high accuracy. We argue that the presented algorithms are applicable to a wide range of languages. Apart from German, the tool has already been deployed for Swedish and Finnish[2], the latter being an agglutinative language which is not part of the Indo-European family.

Acknowldegements

We thank SVOX AG for providing us with morphological grammars and lexica for different languages. This work has been supported by the Swiss authorities within the KTI framework (project number 8915.1 PFES-ES).

References

[1] T. Brants, "TnT: a statistical part-of-speech tagger," in *Proceedings of the sixth conference on Applied natural language processing*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 224–231.

[2] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[3] M. Johnson, "Why doesn't EM find good HMM POS-taggers," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007, pp. 296–305.

[4] M. Tomita, *Generalized LR parsing*. Springer, 1991.

[5] S. Riezler, T. King, R. Kaplan, R. Crouch, J. Maxwell III, and M. Johnson, "Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques," *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 271–278, 2002.

[6] M. Johnson, S. Geman, S. Canon, Z. Chi, and S. Riezler, "Estimators for stochastic "unification-based" grammars," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, Morristown, NJ, USA, 1999, pp. 535–541.

[7] E. Charniak and M. Johnson, "Coarse-to-fine n-best parsing and maxent discriminative reranking," *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pp. 173–180, 2005.

[8] P. Brown, R. Mercer, V. Della Pietra, and J. Lai, "Class-based n-gram models of natural language," *Computational linguistics*, vol. 18, no. 4, pp. 467–479, 1992.

[9] M. Osborne, "Estimation of stochastic attribute-value grammars using an informative sample," in *Proceedings of the 18th International Conference on Computational Linguistics*, 2000, pp. 586–592.

[10] M. Collins, "Discriminative reranking for natural language parsing," in *Proc. 17th International Conf. on Machine Learning*. Morgan Kaufmann, San Francisco, CA, 2000, pp. 175–182.

[11] A. L. Berger, V. J. Della Pietra, and S. A. Della Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[12] S. F. Chen and R. Rosenfeld, "A Gaussian prior for smoothing maximum entropy models," Carnegie Mellon University, Pittsburgh, PA, Tech. Rep., 1999.

[13] Y. Freund and L. Mason, "The alternating decision tree learning algorithm," in *Machine learning: proceedings of the sixteenth international conference (ICML'99)*. Morgan Kaufmann Pub, 1999, p. 124.

[14] JBoost, `http://jboost.sourceforge.net/index.html`.

[15] J. Šnajder, B. Bašić, and M. Tadić, "Automatic acquisition of inflectional lexica for morphological normalisation," *Information Processing & Management*, vol. 44, no. 5, pp. 1720–1731, 2008.

[16] P. Adolphs, "Acquiring a poor man's inflectional lexicon for German," in *Proceedings of LREC'08*, 2008.

[17] M. Forsberg, H. Hammarström, and A. Ranta, "Morphological lexicon extraction from raw text data," in *Proceedings of FinTAL'06*. Springer, 2006, pp. 488–499.

[18] A. Oliver, I. Castellón, and L. Màrquez, "Use of internet for augmenting coverage in a lexical acquisition system from raw corpora," in *Proceedings of the International Workshop on Information Extraction for Slavonic and Other Central and Eastern European Languages*, 2003.

[19] T. Baldwin, "Bootstrapping deep lexical resources: Resources for courses," in *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition*, Ann Arbor, USA, 2005, pp. 67–76.

[20] J. Nicholson, T. Baldwin, and P. Blunsom, "Die Morphologie (f): Targeted lexical acquisition for languages other than English," in *Proceedings of the 2006 Australasian Language Technology Workshop*, 2006, pp. 67–74.

[2] In our evaluation for Finnish, the accuracy of stem prediction was observed to be around 45% for a relatively small corpus with about 15 million tokens.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Automatic Extraction of Arabic Multi-Word Terms

Khalid Al Khatib
Department of Computer Science
Jordan University of Science and Technology
Irbid 22110, Jordan
Khalid_ikh@yahoo.com

Amer Badarneh
Department of Computer Information Systems
Jordan University of Science and Technology
Irbid 22110, Jordan
amerb@just.edu.jo

***Abstract*—Whereas a wide range of methods has been conducted to English multi-word terms (MWTs) extraction, relatively few studied have been applied to Arabic MWTs extraction. In this paper, we present an efficient approach for automatic extraction of Arabic MWTs. The approach relies on two main filtering steps: the linguistic filter, where simple part of speech (POS) tagger is used to extract candidate MWTs matching given syntactic patterns, and the statistical filter, where two statistical methods (log-likelihood ratio and C-value) are used to rank candidate MWTs. Many types of variations (e.g. inflectional variants) are taken into consideration to improve the quality of extracted MWTs. We obtained promising results in both coverage and precision of MWTs extraction in our experiments based on environment domain corpus.**

I. Introduction

AUTOMATIC term recognition (ATR) is still playing an important role in many Natural Language Processing (NLP) applications such as machine translation, book and digital library indexing, hypertext linking, and text categorization. The main objective of ATR is extracting domain specific terms from special language corpora [1].

One of the most important types of ATR is extraction of multi-word terms (MWTs); this comes from the advantages of using MWTs in machine translation, summarization, question answering systems, and many important computational linguistic applications. MWT can be defined simply as a group of words, which are consecutive and constitute a semantic unit [2].

There are three main approaches for extracting MWTs. The first one uses a linguistic filter that depends on syntactic patterns or MWT boundaries detection. The second approach uses a statistical filter to specify the probability of each sequence of words to constitute a MWT. The last one is a hybrid approach of the two previous approaches, this approach extracts candidate MWTs using a linguistic filter, and then it assigns each candidate MWT a score depending on some statistical methods [3].

Most of the statistical methods for MWT extraction concentrate on one of two features: the *unithood* which is "*the degree of strength or stability of syntagmatic combinations or collocation*" [4], and the *termhood* which is "*the degree to which a linguistic unit is related to domain-specific concepts*" [4]. Many methods have been proposed as a *unithood* measure such as mutual information [5], log-likelihood ratio (LLR) [6], and left/right entropy [7], while there is the C-value method [8] as an example of *termhood* measure.

In this paper, we adopt the hybrid approach to extract MWTs from an Arabic corpus. Needless to say that there is a rapid development of computational linguistic applications for Arabic language nowadays, Arabic is the official language of 22 countries, it is spoken by more than 200 million, and it has a very high esteem in the Muslim world [9]. The proposed approach includes two main steps: the linguistic filter and the statistical filter. In the first step, we propose syntactic patterns and use simple part of speech (POS) tagger to extract candidate MWTs. In the second step, two statistical methods are used to rank the candidate MWTs: the LLR method and the C-value method. We consider some related issues like morphological and syntactic ambiguities. For evaluation purpose, we use an environment domain Arabic corpus, the results indicated that our approach is effective, and can be used in many related NLP applications efficiently.

The contribution of our work includes two main points: in the linguistic side, we made an enhancement to the syntactic patterns to be simple and able to exclude a number of wrong candidate MWTs. Moreover, in the statistical side, we take into account both *termhood* and *unithood* measures, since we use a combination between the LLR method and the C-value method in the ranking process.

The paper is structured as follows. Some of the related work is described briefly in section two. In section three we present our proposed approach to extract MWTs. Section four explains how the approach treats the term variations. Section five shows the experiments and the results of applying the extraction approach. The last section contains the conclusion and the future work.

II. Related Work

A lot of work has been done to extract MWT in many languages. This work has been proposed by using linguistic filters, statistical methods, or both as a hybrid approach. However, the majority of the latest MWT extraction systems have adopted the hybrid approach, because it has given better results than using only linguistic filters or statistical methods [10].

978-83-60810-27-9/09/$25.00 © 2010 IEEE

As far as we know, there are a few MWTs extraction systems of Arabic language, one of them is the work which has been presented by Attia, M. A. [11], he has adopted the linguistic approach by doing manual and semi-automatically extraction of Arabic MWTs.

In another work, Boulaknadel, S.; Daille, B.; and Aboutajdine, D. [12] have adopted the hybrid approach to extract Arabic MWTs. The first step of their system is extraction of MWT-like units, which fit the follow syntactic patterns: {noun adjective, noun1 noun2, noun1 preposition noun2} using available part of speech tagger, taken into consideration graphical, inflectional, morphosyntactic, and syntactic variants. The second step is ranking the extract MWT-like units using association measures, these measures are: log-likelihood ratio, FLR, Mutual Information (MI^3), and t-score. The evaluation process includes applying the association measures to an Arabic corpus and calculating the precision of each measure using a collected reference list of Arabic terms [12].

Another system has been proposed by Bounhas, I. and Slimani, Y. [13]; they have proposed a hybrid approach to extract compound nouns. In the linguistic side, they used matcher between POS tagger and morphological analyzer to produce sequences of tokens, each token could be represented by a number of solutions, and then using a syntactic parser to extract candidates of compound nouns. In the statistical side, they applied the LLR method. In the evaluation step, they used almost the same corpus and reference list which have been used in [12]. Their results were promising especially with bigram MWTs [13].

III. Proposed Approach

The proposed MWTs extraction approach has the following features: (i) the system is simple as much as possible to avoid performance and complexity problems, (ii) accurate since there are previous systems [12] and [13] which have got good results, and (iii) able to cover the importance of MWT to increase the possibility for using it in other NLP systems such as summarization and machine translation systems.

The proposed approach for extracting Arabic MWTs is composed of two main steps: (i) the linguistic filter, where we extract candidate MWTs, and extract bigrams from candidate MWTs (ii) the statistical filter, where we rank bigrams by the LLR and C-value scores. In the following subsections, we cover the two steps in more details.

A. Linguistic Filter

There are many types of MWT, such as idioms, phrasal verbs, verbs with particles, compound nouns, and collocations [13]. Our choice was similar to [12] and [13] where we chose to deal with compound nouns, since we agree with the fact that nouns can represent document's subject efficiently [13]. To extract candidate MWTs, we left the syntactic patterns which have been used in many systems like [12], and propose new patterns based on definite and indefinite types of nouns. Table 1 and Figure 1 show our syntactic patterns.

Table 1.
Syntactic patterns of MWT

(1)	definite noun ⇒ one or more definite nouns
(2)	indefinite noun ⇒ one or more indefinite nouns
(3)	indefinite noun ⇒ one or more definite nouns
(4)	(1) or (2) or (3) ⇒ preposition ⇒ (1) or (2) or (3)

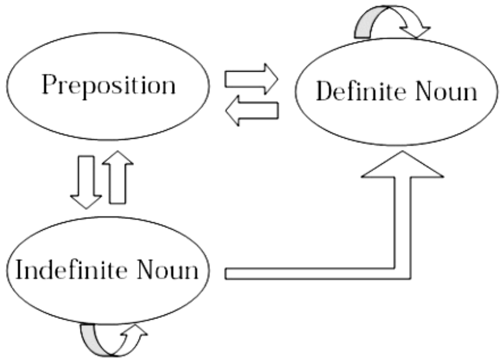


Fig.1: Graphical model of syntactic patterns

These syntactic patterns have some advantages over the patterns used in other extraction systems such as [12]. Firstly, it doesn't require an advanced part of speech tagger. It needs simple one with just three categories (noun, verb, particles). Obviously, this means better performance because most of the POS taggers still have problems in differentiation between nouns and adjectives [11], and less complexity since we don't need many words' classes such as adjectives and adverbs, which advanced POS taggers try to determine. Secondly, these patterns include the entire correct candidate MWTs which might be extracted by patterns in [12], and exclude a collection of wrong candidate MWTs which patterns in [12] may extract. Table 2 shows examples of MWTs extracted from proposed patterns and patterns proposed in [12].

The extraction of candidate MWTs starts with the preprocessing step, which includes four sub steps. The first one is the tokenization, where we separate text into main tokens (words). Words are always separated by white spaces or punctuation marks in Arabic language. The second sub step is the stemming, where the stem of each word is extracted using an available stemmer proposed by khoja, S. [14].

Table 2.
Examples of MWTs Extracted from Proposed Patterns and Patterns Proposed in [12]

Candidate MWT	New patterns	Patterns used in [12]	Is it correct MWT?
التنوع الحيوي biodiversity	yes	yes	yes
تقطير الماء water distillation	yes	yes	yes
تلوث حراري thermal pollution	yes	yes	yes
السماء غائمة the sky is cloudy	no	yes	no

This stemmer specifies the word's stem and type, the types which the stemmer can define are {stemmed word, stop word, strange word}, the stemmer has its own list of stop words which we modify through adding additional stop words. The third sub step is the frequency calculation. In this sub step, we calculate the frequency of each word as well as the frequency of each stem. This sub step is important in dealing with variations. The last sub step is the sentence segmentation. We use simple method for this purpose, where special punctuation marks are used to determine the boundaries of the sentences.

The second step after the preprocessing one is the word's classification (by means of Simple POS tagger). There are three main classes of the word in Arabic: noun, verb, and particle. What we care about in this step is distinguishing the nouns from other classes, since our patterns primarily depend on nouns. Although the available POS taggers can help us very well in this step, we decided to ignore them and adopted the approach which has been proposed by Ahmad T. and Salah A. [15].There are two reasons for that. First, this approach is simple and accurate. Therefore, it is able to keep one of the merits of our syntactic patterns, which is the simplicity. Second, this approach has a morphological analyzer phase. This phase is helpful on dealing with term variations.

The architecture of the adopted approach for words' classification contains three main phases. The first phase is the lexicon analyzer. In this phase a lexicon of stop lists in Arabic language is defined. This lexicon includes prepositions, adverbs, conjunctions, interrogative particles, exceptions, questions and interjections. All the words have to pass this phase, if the word is found in the lexicon, it is considered as tagged to one of the previous closed lists. The next phase is the morphological analyzer. Each word which has not been tagged in the previous phase will immigrate to this phase. In this phase, firstly, the affixes of each word are extracted, the affix is a set of prefixes, suffixes and infixes. After that, these affixes and the relation between them are used in a set of rules to tag the word into its class. It is important to say that this phase is the core of the system, since it distinguishes the major percentage of untagged words into nouns or verbs. The last phase is the syntax analyzer. This phase can help in tagging the words which the previous two phases failed to tag. It is consisting of two rules: sentence context and reverse parsing. The sentence context rule is based on the relation between the untagged words and their adjacent, where Arabic language has some types of relations between adjacent words. These relations can help in tagging the words into its corresponding class. The reverse parsing rule is based on Arabic context-free grammar. There are ten rules, which are used frequently in Arabic language.

The Third step to extract the candidate MWTs is extraction of sequences of nouns, as well as sequences of nouns that connected by a preposition. In this step, we consider each sentence as a separated unit, and using the words' classification approach to extract sequences of nouns. For the sequences of nouns that connected by prepositions, we had two types of prepositions: prepositions which constitute a separated word and prepositions which are stuck with another word. We deal with the two types because our syntactic patterns consider prepositions from the two types. Table 3 shows examples of prepositions' types. Table 4 shows examples of extracted sequences of nouns.

Table 3.
Examples of Prepositions' Types

separated proposition	من	e.g. التخلص من النفايات disposal of wastes
stuck preposition	ب	e.g. الري بالتنقيط drip irrigation

Table 4.
Examples of Extracted Sequences of Nouns

sequence of nouns	e.g. منظمة الأرصاد الجوية العالمية world meteorological organization
sequences of nouns that connected by a preposition	e.g. التحكم عن بعد remote control
	e.g. التعبير بالإشارة the expression by reference

The last step is testing each extracted sequence based on MWTs syntactic patterns, the sequences which fit the patterns will be considered as candidate MWTs. Figure 2 shows the main steps for extraction of candidate MWTs using the linguistic filter.

MWT might be classified based on the number of words. Bigram term is the term of two words. We decide to consider bigrams and discard the other terms which consist of more than two words. Simply, we noted from the terminology databases that the major percentage of compound nouns is bigrams.

In our work, we extract the bigrams from each candidate MWT. We noted that some bigrams are MWT while others are not. However, this is the last step before using the statis-

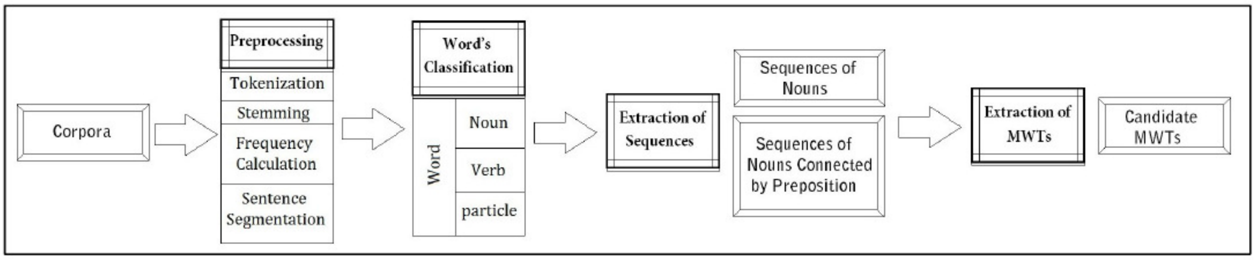


Fig. 2: The main steps for extraction of candidate MWTs using the linguistic filter

tical methods to rank the terms. Table 5 shows an example of bigrams' extraction.

TABLE 5.
EXAMPLE OF BIGRAMS' EXTRACTION

Candidate MWT	برنامج الولايات المتحدةلبحوث القطب الجنوبي united states Antarctic research program	
Bigrams	برنامج الولايات states program	NOT MWT
	الولايات المتحدة united states	MWT
	المتحدةلبحوث united for research	NOT MWT
	لبحوث القطب for Antarctic research	NOT MWT
	القطب الجنوبي Antarctic	MWT

B. Statistical Filter

Using statistical methods can help with morphological and syntactic ambiguities and therefore, increasing the quality and the quantity of correct extracted MWTs. In this step, we consider both *termhood* and *unithood* measures to get better results than using only one measure type [16].

To consider the *unithood*, we chose LLR method because it gives good results with Arabic MWTs extraction [12]. For the *termhood* we adopted C-value method because it has a wide acceptance as a valuable method to rank candidate MWTs [16]. LLR method can be used efficiently as significance of association measure between the two words in the bigram [17]. Regarding the C-value method, it requires simple modification to be able to rank the extracted bigrams. We list the entire candidate MWTs and the extracted bigrams as the first step, and then we apply the C-value equation only to the bigrams. Note that we would not be able to calculate the C-value score for the bigrams without some information about the candidate MWTs which contain those bigrams.

Practically, we make a list of bigrams ranked by the LLR. We make another list, which is ranked by the C-value method. Lastly, we combine the two lists to get a new list ofbigrams ranked by the two statistical methods. Figure 3 shows the Log-Likelihood ratio equations, and Figure 4 shows C-value method equation. Figure 5 shows the algorithm of proposed statistical filter.

Contingency table

	$V=v$	$V\neq v$
$U=u$	$O11$	$O12$
$U\neq u$	$O21$	$O22$

U: first word of the bigram. V: second word of bigram
$O11$: #compound nouns with U and V.
$O12$: #compound nouns with U but without V.
$O21$:#compound nouns with V but without U.
$O22$:#compound nouns without V and without U.

$$R1 = O11 + O12 \qquad C1 = O11 + O21$$

$$R2 = O21 + O22 \qquad C2 = O12 + O22$$

$$N = O11 + O12 + O21 + O22 = R1 + R2 = C1 + C2$$

$$LLR = -2\log\left\{\frac{L(O11, C1, r) * L(O12, C2, r)}{L(O11, C1, r1) * L(O12, C2, r2)}\right\}$$

$$L(k,n,r) = r^k \times 1 - r^{n-k}$$

$$r = \frac{R1}{N} \qquad r1 = \frac{O11}{C1} \qquad r2 = \frac{O12}{C2}$$

Fig.3: Log-Likelihood ratio equations

$$C-Value = \begin{cases} \log_2|a|.f(a) & \text{If a is not nested} \\ \log_2|a|(f(a) - \frac{1}{p(Ta)}\sum_{b\in Ta} f(b) & \text{Otherwise} \end{cases}$$

a :	The candidate MWT.
b :	Longer candidate MWTs.
$\|a\|$:	Length of the candidate MWT.
$f(a)$:	Frequency of occurrence of a in the corpus.
Ta :	Set of extracted candidate MWTs that contain a.
$P(Ta)$:	Number of candidate terms in Ta.
$f(b)$:	Frequency of occurrence of longer candidate MWT b in the corpus.

Fig.4: C-value method equation

```
Input : List of Candidate multi-word terms

output : List of Ranked Bigrams

Method :

Extract bigrams from List of Candidate multi-word terms
For each bigram b do{

    Calculate log-likelihood ratio value{
        Input: Bigram b
        Output: LLR value of bigram b
        Method: use equations in Fig.3
    }

    Calculate C-method value {
        Input: List of Candidate multi-word terms
               +
               Bigram b
        Output: C-method value of bigram b
        Method: use equation in Fig.4
    }

}

Sort (Bigrams, by LLR value, descending)
Sort (Bigrams, by C-method value, descending)
Make a list of bigrams sorted by LLR
where:
    The index of the bigram represents its rank

Make a list of bigrams sorted by C-value method
where:
    The index of the bigram represents its rank

Sort (Bigrams, by LLR rank+ C-value rank, descending)

return List of bigrams ranked by C-value method
                                   +
                         log- likelihood ratio
where:
    The index of the bigram represents its rank
```

Fig. 5: The statistical filter Algorithm

IV. Term Variations

When we try to extract MWT, term variation is one of the significant factors that should be studied, in other words, it is important to show how the proposed approach deals with different types of variations.

In our proposed approach, we started dealing with the term's variations in the statistical step. As we mentioned before, the input of the statistical step is the extracted bigrams which we try to rank using the statistical methods. Obviously, these methods use the frequency as a primary factor of weighting. What we did here is using the stem's frequency of nouns instead of word's frequency, it's clear that verbs are excluded from the stem's frequency calculating process.

To clarify the point, suppose we have the following bigrams which have graphical and inflectional variants:

(Environmental pollution)

تلوث البيئة	تلوث البيئه	بتلوث البيئة
لتلوث البيئة	تلوث البيئات	التلوث البيئي

The first word in all bigrams has the same stem, and we can say the same about the second word. This means that the statistical methods will consider these bigrams as identical, and it will give them the same score. After completing the list of ranked bigrams, an enhancement process will be available for this list; all the bigrams with graphical and inflectional variants will be removed except the best one, we consider the best choice is the bigram which has the smallest number of common affixes that might be existed in different inflectional forms. Moreover, some prefixes of words in bigrams are removed before choosing the best bigram. The best choice for the previous bigrams is [تلوث البيئة].

Morphosyntactic and syntactic variants are more complex than the previous ones, and need advanced linguistic processing to deal with, our proposed system can deal efficiently only with some types of morphosyntactic variants, as well as syntactic variants from modification type and postposition sub-type. Table 6 shows examples of variants' types that our approach deals with.

Table 6.
Examples of variants that our approach deals with

graphical variants	e.g. تيارات حرارية /تيارات حراريه (thermals)
inflectional variants	e.g. سماد نباتي (vegetable mould) أسمدة نباتية (vegetable moulds)
morphosyntactic variants modification/postposition	e.g. مدى الرؤية (visibility) مدى الرؤية الرأسي(vertical visibility)
syntactic variants	e.g. تلوث إشعاعي/ التلوث بالإشعاع (radioactive pollution)

V. Experiments and Results

A. The corpus

The lake of Arabic specialized domain corpora forced the researcher to build new corpora to evaluate their approaches. In fact, using different corpus from different terminology extraction approaches has a negative impact of the ability to compare between them.

To keep our system comparable (as much as possible) with previous work on Arabic MWTs, we used corpus with some similar properties to that which used in [12] and [13].

The corpus belongs to the environment domain and collected from number of websites. The website[1] which has been used in [12] and [13] is part of the corpus. Table 7 shows some information about the corpus.

TABLE 7.
STATISTICAL INFORMATION OF THE CORPUS

number of words (tokens)	522845
number of stemmed words	495618
number of nouns	281531
number of sequences of nouns	62761
number of candidate MWTs	43018

B. Evaluation and Results

Evaluation of ATR approaches is a complex task, basically, there are no specific standards for evaluate and compare different ATR approaches. However, the most of the approaches have used one of two evaluation methods (and sometimes both): reference list and validation [17].

For the evaluation purpose we decided to evaluate our approach using two methods. In the first one, we used the same way used in [12] and [13]. We consider the MWT is correct, if its translation is included in Eurodicautom[2] (terminological database). Unfortunately, Agrovoc[3] (terminological database includes Arabic terms) is not available currently. The second method is the manual validation of the terms. In fact, we found many correct MWTs which are not included in the used terminology database.

The results of our approach are given in Table 8. Obviously, the results show that using C-value method gives better results than using LLR method, while using the combination between the two methods gives us the best results. For the results of LLR method, we can explain the differences between our results and the results obtained in [12] to the difference of the used corpus.

Indeed, it is important to say that we count the terms with basic singular-plural and definitude variation as correct terms, since most of ATR studies allow for these kinds of variations [4]. Figure 6 shows the results of the proposed approach. Figure 7 shows sample of extracted Arabic MWTs ranked by the combination between C-value and LLR methods.

VI. CONCLUSION

In this paper, we have presented a hybrid approach to extract Arabic MWTs. We have concentrated on compound nouns as an important type of MWT, and chose to extract bigram terms, which constitute a high percentage of compound nouns. Extraction of MWT required substantial software development effort. The proposed approach started with the linguistic filter step, this step contains: preprocessing, word's classification, extraction of nouns' sequences, as well as nouns' sequences that connected by prepositions, testing each extracted sequence based on MWTs syntactic patterns, and finally, extraction of bigrams from candidate MWTs.

The next step is the statistical filter. This step includes rank the bigrams based on LLR and C-value methods, and this step follows by dealing with different types of term vari-

TABLE 8.
THE RESULTS OF THE STATISTICAL METHODS

# terms / Method	Top 25				Top 50				Top 100				Top 150			
	correct	not correct	with variations	precision	correct	not correct	with variations	precision	correct	not correct	with variations	precision	correct	not correct	with variations	precision
LLR	23	2	0	92%	43	7	0	86%	78	19	3	78%	117	30	3	78%
C-value	22	3	0	88%	45	5	0	90%	86	11	3	86%	128	18	4	85%
LLR+C-value	23	2	0	92%	47	3	0	94%	94	5	1	94%	133	16	1	89%

[1] http://www.greenline.com.kw

[2] http://iate.europa.eu

[3] www.fao.org/agrovoc/

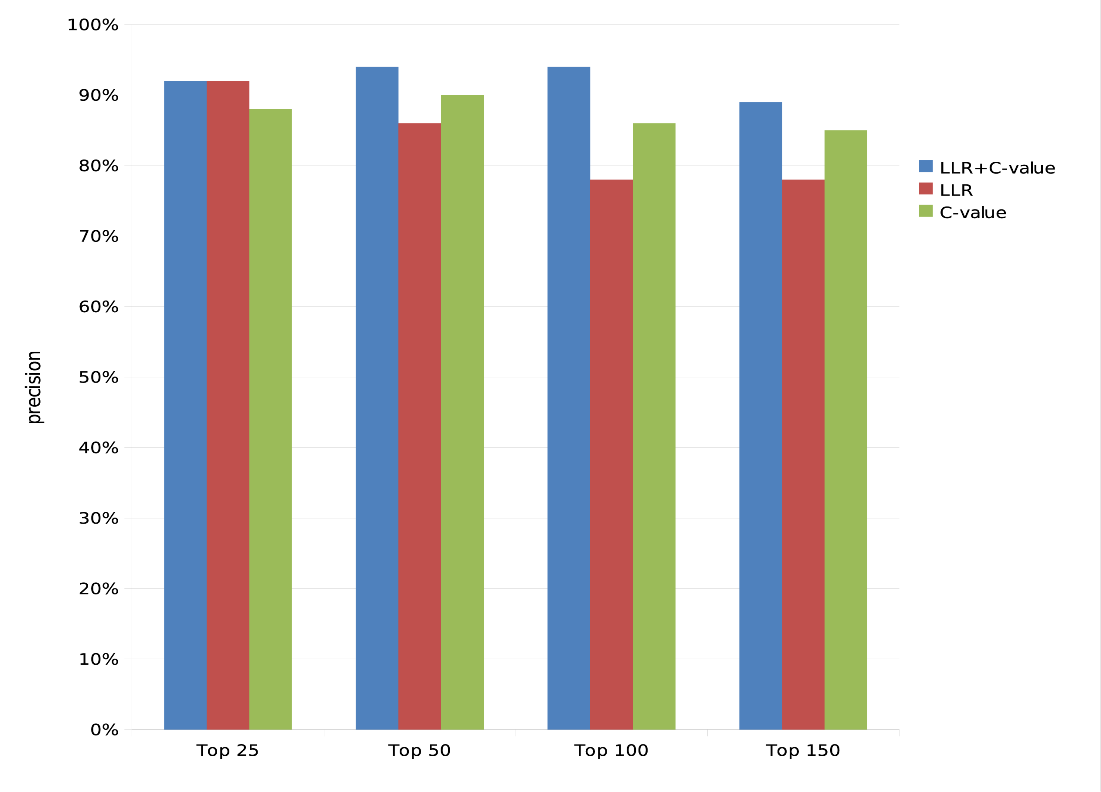
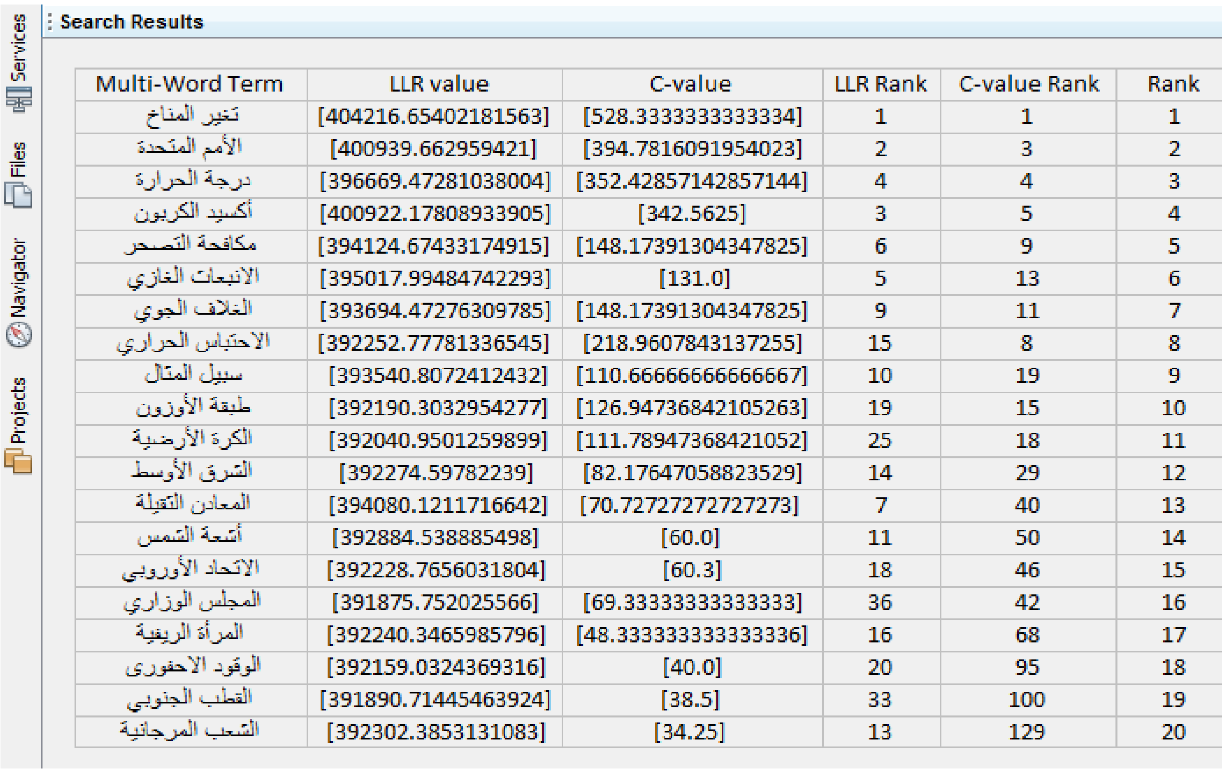


Fig.6: The results of the statistical methods



Multi-Word Term	LLR value	C-value	LLR Rank	C-value Rank	Rank
تغير المناخ	[404216.65402181563]	[528.3333333333334]	1	1	1
الأمم المتحدة	[400939.662959421]	[394.7816091954023]	2	3	2
درجة الحرارة	[396669.47281038004]	[352.42857142857144]	4	4	3
أكسيد الكربون	[400922.17808933905]	[342.5625]	3	5	4
مكافحة التصحر	[394124.67433174915]	[148.17391304347825]	6	9	5
الانبعاث الغازي	[395017.99484742293]	[131.0]	5	13	6
الغلاف الجوي	[393694.47276309785]	[148.17391304347825]	9	11	7
الاحتباس الحراري	[392252.77781336545]	[218.9607843137255]	15	8	8
سبيل المثال	[393540.8072412432]	[110.66666666666667]	10	19	9
طبقة الأوزون	[392190.3032954277]	[126.94736842105263]	19	15	10
الكرة الأرضية	[392040.9501259899]	[111.78947368421052]	25	18	11
الشرق الأوسط	[392274.59782239]	[82.17647058823529]	14	29	12
المعادن الثقيلة	[394080.1211716642]	[70.72727272727273]	7	40	13
أشعة الشمس	[392884.538885498]	[60.0]	11	50	14
الاتحاد الأوروبى	[392228.7656031804]	[60.3]	18	46	15
المجلس الوزاري	[391875.752025566]	[69.33333333333333]	36	42	16
المرأة الريفية	[392240.3465985796]	[48.333333333333336]	16	68	17
الوقود الاحفورى	[392159.0324369316]	[40.0]	20	95	18
القطب الجنوبي	[391890.71445463924]	[38.5]	33	100	19
الشعب المرجانية	[392302.3853131083]	[34.25]	13	129	20

Fig.7: Sample of extracted Arabic MWTs ranked by the combination between C-value and LLR methods

ation. The results show that our approach of using a combination between LLR and C-value methods in the ranking process gave better results than using only one of them. In general, we obtained promising results in both coverage and precision of MWT extraction in our experiments based on environment domain corpus.

In the future, we will work to enhance the linguistic filter to be able to extract more complex types of MWTs, use more combinations of statistical methods to rank the candidate MWTs, and extend our method to deal with n-grams MWTs.

References

[1] Korkontzelos, I.; Klapaftis, I. P.; and Manandhar, S.: *Reviewing and Evaluating Automatic Term RecognitionTechniques.* In Proceedings of the 6th international Conference on Advances in Natural Language processing, 2008.

[2] Zhang, W.; Yoshida, T.; and Tang, X: *A Study on Multi-word Extraction from Chinese Documents.* In Advanced Web and Network technologies, and Applications: Apweb, 2008.

[3] Koeva, S.: *Multi-word term extraction for Bulgarian.* In Proceedings of the Workshop on Balto-Slavonic Natural Language Processing, 2007.

[4] Kageura,K.;and Umino, B.: *Methods of Automatic Term Recognition A Review,* Termonology 3(2), 259-289.1996.

[5] Church, K. W.; Hanks, P.: *Word association norms, mutual information, and lexicography.* Computational Linguistics 16(1), 22–29, 1990.

[6] Dunning, T.: *Accurate Methods for the Statistics of Surprise and Coincidence.* Computational Linguistics, vol. 19(1), pp. 61-74, 1994.

[7] Patry, A.; Langlais, P.: *Corpus-based Terminology Extraction.* In the 7th International Conference on Terminology and Knowledge Engineering, pp. 313– 321 , 2005.

[8] Frantzi, K. T.; Ananiadou, S.; Mima, H.: *Automatic Recognition of Multi-Word Terms: the C-value/NC-value method.* International Journal on Digital Libraries Vol. 3, No. 2, pp.115–130, 2000.

[9] http://www.un.org/depts/OHRM/sds/lcp/Arabic/

[10] Tadić, M.; Šojat, K.: *Finding multiword term candidates in Croatian.* In the Proceedings of IESL2003 Workshop, pp. 102-107, 2003.

[11] Attia, M. A.: *Handling Arabic Morphological and Syntactic Ambiguity within the LFG Framework with a View to Machine Translation,* doctoral thesis, University of Manchester, Faculty of Humanities, 2008.

[12] Boulaknadel, S.; Daille, B.; and Aboutajdine, D.: *A multi-word term extraction program for Arabic language,* In the 6th international Conference on Language Resources and Evaluation LREC, pp. 1485-1488,2008.

[13] Bounhas, I.; Slimani, Y.: *A hybrid approach for Arabic multi-word term extraction,* NLP-KE 2009. International Conference on Language Processing and Knowledge Engineering, vol., no., pp.1-8, 24-27, 2009.

[14] http://zeus.cs.pacificu.edu/shereen/research.htm#stemming.

[15] Al-Taani, A. T.; Abu-Al-Rub, S.: *A rule-based approach for tagging non-vocalized Arabic words.* The International Arab Journal of Information Technology, Volume 6 (3): 320-328 , 2009.

[16] Thuy Vu; Ai Ti Aw; and Min Zhang: *Term extraction through unithood and termhood unification.* In Proceedings of the 3rd International Joint Conference on Natural Language Processing, 2008.

[17] Pazienza, M. T.; Pennacchiotti, M.; Zanzotto, F. M.: *Terminology extraction: an analysis of linguistic and statistical approaches.* In Knowledge Mining, Springer Verlag, 2005.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 419–428

ISBN 978-83-60810-27-9
ISSN 1896-7094

“Beautiful picture of an ugly place”. Exploring photo collections using opinion and sentiment analysis of user comments

Slava Kisilevich*, Christian Rohrdantz†, Daniel Keim‡
Department of Computer and Information Science
University of Konstanz, Germany
{slaks*,rohrdantz†, keim‡}@dbvis.inf.uni-konstanz.de

***Abstract*—User generated content in the form of customer reviews, feedbacks and comments plays an important role in all types of Internet services and activities like news, shopping, forums and blogs. Therefore, the analysis of user opinions is potentially beneficial for the understanding of user attitudes or the improvement of various Internet services. In this paper, we propose a practical unsupervised approach to improve user experience when exploring photo collections by using opinions and sentiments expressed in user comments on the uploaded photos. While most existing techniques concentrate on binary (negative or positive) opinion orientation, we use a real-valued scale for modeling opinion and sentiment strengths. We extract two types of sentiments: opinions that relate to the photo quality and general sentiments targeted towards objects depicted on the photo. Our approach combines linguistic features for part of speech tagging, traditional statistical methods for modeling word importance in the photo comment corpora (in a real-valued scale), and a predefined sentiment lexicon for detecting negative and positive opinion orientation. In addition, a semi-automatic photo feature detection method is applied and a set of syntactic patterns is introduced to resolve opinion references. We implemented a prototype system that incorporates the proposed approach and evaluates it on several regions in the World using real data extracted from Flickr.**

I. Introduction

With the fast development of user-centered Internet technologies, we witness a rapid growth of Web resources, which not only allow users to consume textual information, but also to generate their own. This leads to dramatic improvements of products and services. For example, nowadays it is difficult to imagine that we would book a hotel room without checking the hotels overall ranking or without reading comments previously written by other users. We are also less inclined to buy a product without reading comments or ratings about its quality. In fact, written opinions have become essential components in decision-making processes. Furthermore, opinionated texts are now common in almost all parts of our life. They are essential parts of blogs, news, financial market reports, product reviews, etc. However, textual information generated on the Web almost grow at an uncontrollable pace, and manual skimming through user opinions has become a time consuming process.

There has been extensive research within the past ten years on automatic opinion and sentiment analysis. Different algorithms and approaches have been proposed for the analysis of customer feedback data from web surveys [1], movie reviews [2], [3], [4], news articles [5], product reviews [6], [7], financial blogs and news [8], [9], stock message boards [10], opinions in the domain of fast food restaurants [11], and blogs [12].

A typical task in opinion mining is to determine whether a document (review, comment) is bearing a positive or negative connotation [13], [2], [6], [3], [10], [11], [14]. If either connotation is present, the task can be formulated as a classification problem with two class labels (positive and negative) [15]. Three different kinds of approaches have been used: Unsupervised [2], semi-supervised [14] and supervised [1], [3], [12], [10], [16], [9] ones. Supervised machine learning approaches perform good if sufficient labeled training data exist (for example, in the movie reviews domain users assign ranks to movies along with their opinionated text). However, in domains where labels are not easy to acquire or where opinion orientation is measured on a real-valued scale [17], unsupervised approaches are more favorable.

In this paper, we consider the problem of opinion and sentiment analysis of users' comments written for photos, uploaded to photo sharing web sites. Photo sharing web sites, in general, allow users to maintain their own albums of photos. Users can view photos of other members and write comments for a particular photo. In this paper we work with photo comments from Flickr[1].

Before proceeding further with the analysis, we need to understand what are the similarities and what are the differences between the domain of user photo comments and other domains. Having manually examined hundreds of user comments, we found some similarity to blogs [12], where opinions are stated in the beginning of the paragraph. Similar to blogs, the same user can write several comments about the same photo, but usually the first comment contains the opinions and sentiments, while subsequent comments mostly include neutral information like responses to comments of others or the photo owner. The following example shows two comments from the same user. In the first comment there is an expression of sentiment like “Powerful place and story”, while the second comment was made after the owner of the photo wrote his response.

[1] http://www.flickr.com/

978-83-60810-27-9/09/$25.00 © 2010 IEEE

(1) *this is great. I visited Dachau, but don't remember this part. but I hear they have added some things in the last 5 years. Powerful place and story, thanks for sharing*

(2) *I was there about 8 years ago and I don t recall this hall way. Was this one of the houses, or near the main complex where the museum and films were?*

As already mentioned, the owner of the photo can also participate in the discussion about his own photo. In this case, his opinions can introduce a certain bias, which suggests that comments of the photo owner should be excluded from the analysis. The following is a short example of two comments written by the owner of the photo to people as a response to their comments.

(1) *thanks for the comments. i also found the colors both beautiful and chilling...a very creepy place for sure*

(2) *Thanks! I was fortunate to actually capture the impression it made on me standing there in person*

Detailed inspection of user comments revealed that comments are noisy, relatively short, and with only few negations. They may be written in any languages, contain arbitrary syntactic structures and typos. Moreover, they may contain a mixture of opinions on the quality of the photo (usually positive) "Great shot", "Nice picture" and sentiments or moods expressed towards objects depicted on the photo ("Sad place"). As mentioned above, a widely used approach is to classify documents using a binary classification. This approach seems inappropriate in our case for two reasons: (1) Photo comments have two subjects of opinions (opinions on the photo and sentiment towards objects). Consequently, we will loose valuable information if the overall score will be a mixture of two opinion scores. (2) Since most of the opinions are positive, we will end up with most of the photo comments classified as positive. In order to draw a clear analysis, we propose two improvements over existing approaches. We extract two types of opinions: (1) opinions that relate to the photo quality, and (2) general sentiments targeted towards objects depicted on the photo. Supervised machine learning approaches are not feasible in our case since it is very hard to find agreements between human annotators on a real-valued scale, e.g. the difference in opinion strength between "Great shot" and "Amazing photo" cannot be clearly defined. For that reason, we propose an unsupervised approach for opinion scoring using concepts of word importance based on statistical properties derived from the field of information retrieval [18]. Further observations revealed that opinionated pieces of text are mostly accompanied by adjectives, which is in accordance with past findings [19], [20]. Based on these facts, we generated our own lexicon of adjectives extracted from the corpora of user comments, and analyzed its usage with respect to photo quality opinions and general sentiments, as well as their usage by commenters. We found that in the majority of cases adjectives are used directly with the subject of the opinion ("Great shot") and that the most frequently used adjectives are the same, even if different regions of the world are considered with photos of different topicality. The latter suggested that a finite lexicon of adjectives can be used for opinion analysis of photo comments in many regions around the world. In addition, we also discovered an interesting property of the frequency of adjectives, which is perfectly described by Zipf's law [21]. Our approach is based on a sentence-level opinion analysis, which makes it scalable in dynamic environments like photo comments, where a comment can be added at any time by any of the members.

We developed a desktop Google Earth-based system [22] that combines map navigation and photo exploration using *opinion* and *sentiment* scores as well as a number of derived textual features. The system is capable of showing the photos filtered by one of the features, locating them on the map or seeing them sorted sequentially in an additional window. We believe that our approach can be a very useful extension of photo sharing web sites that will enrich and improve the currently available service capabilities.

The main contributions of the paper can be summarized as follows:

- Our model is based on the corpora extracted from users' photo comments.
- We construct and work with a finite lexicon of opinion words in contrast to the majority of approaches in which seed lists are used to infer scores of unknown opinion words.
- We develop a model that consists of two types of scores: *opinion* regarding the photo and *sentiment* towards the subject of the photo. For this purpose, we suggest a semi-automatic extraction of photo features and a set of syntactic opinion reference patterns.
- We model the orientation strength based on word distributions without using any external dictionaries, while the semantic orientation (positive or negative) of a word is determined by the predefined lexicon of positive and negative opinion-bearing words.
- We provide a continuous scale for opinion and sentiment orientation.
- Our approach allows dynamic updates of scores when new comments are added to the system, which makes the whole method readily applicable in real-world tasks.
- We demonstrate our approach using a Google Earth-based framework.

II. Related Work

Existing approaches in the context of opinion analysis can be broadly divided into several categories. We will review the following categories as they are closely related to our work *opinion classification and orientation*, *lexicon generation* and *feature-based opinion analysis*. A more detailed overview can be found in [15].

A. Opinion classification and orientation

An unsupervised approach for review classification was applied in [2] using pointwise mutual information (PMI) between a phrase containing an adjective or adverb, and the positive word "excellent" and negative word "poor". The PMI probabilities were calculated based on the number of pages, retrieved by the AltaVista search engine, that contain one of the phrases or two phrases together. The review was classified as positive or negative using the average opinion orientation of all phrases.

[23] proposed a similarity measures of adjectives for semantic orientation using the WordNet [24] synonymy relation. The idea is to count the geodesic distance (similarity relation) of an arbitrary word to a word *good* and *bad* and to determine its orientation based on its similarity relation to one of these words.

A Naive Bayes Classifier was used in [3] for classifying movie reviews, while [10] use Naive Bayes as one of five classifiers with majority voting. A Support Vector Machine (SVM) classifier was used by [1] for classifying customer feedback data. [9] applied SVM on financial blogs.

[17] proposed a real-valued scale opinion orientation based on a classification of adverbs (doubt, strong and week intensifiers, negation and minimizers), different verb categories (positive, negative, conjecture and declarative verbs) and complex relationships of adverbs, adjectives and verbs in the text.

B. Lexicon generation

[13] proposed an approach for the identification of semantic orientations of words using a seed list of positive and negative orientations and the conjunctions (and, or, but) between adjectives with known orientation. [5] used a seed lexicon containing eight positive and eight negative words from the news domain to classify a news article as positive or negative. A vector of words from an article was constructed and similarity between it and the vector of negative and positive seed words was measured using cosine measure. A novel approach was proposed in [7] to construct a domain sentiment lexicon using a seed list of sentiment words and relations of these words to specific topics (in the product domain). The key observation is that sentiment words are directly associated with product features. This observation was used to identify new sentiment words using features and new features using new sentiment words. [12] used a Wikipedia dictionary to determine the polarity of adjectives. [25] generated a dictionary called SentiWordNet using WordNet with three sentiment scores (positive, negative and objective) for each WordNet synset.

C. Feature-based opinion analysis

In addition to the approaches that try to detect the sentiment of sentences or even documents as a whole, the task of feature-based analysis is to investigate to which feature (e.g. entity, topic, attribute) a sentiment or opinions refers. This is also very important in our case since we want to separate opinions that refer to photo features from other opinions. Having identified a set of features and a set of opinion words with respective orientation values (+1/-1) in a sentence, the task is to assign the opinion words to features. Different approaches have been suggested in the past. Some of them use distance-based heuristics ([26], [27]). The closer an opinion word is to a feature word, the higher is its influence on the feature. Other approaches exploit advanced natural language processing methods, like dependency parsers, to resolve linguistic references from opinion words to features.

[28] extract pairs (opinion word, feature) based on 10 extraction rules that work on dependency relations involving subjects, predicates and objects gathered from the Minipar dependency parser[2]. [29] use lexico-syntactic patterns in a bootstrapping approach for subjectivity classification. They define a set of 13 syntactic templates (e.g. subject passive-verb) and concrete example patterns for such templates (e.g. subject was satisfied). However, the purpose is only to resolve relations between opinion holders and verbs for subjectivity classification.

III. Photo Comment Corpus

In this section we outline the photo comment collection, the creation of the corpus and the preprocessing techniques.

A. Data Collection

We collected photo comments from Flickr, the largest web community for photo and web sharing, using its publicly available API[3]. Since the API does not allow downloading metadata for a particular region in the World, the downloading was performed as follows: An initial user id was used to download his photo metadata (owner id, photo id, photo url, date a photo was taken, geotagged information, comments). Then, we downloaded all the users' contacts. To speed up the process of retrieving heterogeneous users, we retrieved all groups to which the individual user belongs, and using group information, we were able to retrieve all the people who belong to these groups. Beginning in June, 2009 (as part of another project) we collected metadata for about 90 million geotagged photos from about 7.6 million users by the end of April, 2010.

B. Development of Corpora

1) Region selection: Five regions (Dachau, Auschwitz, Wisla, Krakow and Warsaw) were defined for analysis. The rationale behind selecting these regions is that we want:

- To find differences in comment types between regions
- To find differences in the usage of parts of speech (adjectives and nouns)
- To build a model that better reflects different kinds of comments

We assumed that Dachau and Auschwitz concentration camps should contain special kinds of comments (negative emotions) that would differ from comments in general touristic places. Wisla, we assumed, is a neutral region visited rarely by tourists while Krakow and Warsaw were selected as large Polish cities that include many touristic attractions.

Table I summarizes the statistics related to the selected regions.

[2] http://webdocs.cs.ualberta.ca/~lindek/minipar.htm

[3] http://flickrnet.codeplex.com/

TABLE I
STATISTICAL INFORMATION RELATED TO FIVE REGIONS SELECTED FOR ANALYSIS

Region	Area	# commented photos	# owners	# commenters
Krakow	$120km^2$	8127	1257	23045
Warsaw	$60km^2$	8690	1140	22695
Wisla	$43km^2$	117	39	603
Auschwitz	$12km^2$	505	138	1687
Dachau	$14km^2$	329	121	1062

2) Preprocessing: For every region, we selected photos that contain at least one comment. Photos that do not contain comments were removed from further analyses.

Photo comments are very noisy and unstructured. They may contain HTML tags which should be filtered from the original text. In addition, they may contain different irrelevant sections that have to be removed such as URL links or invitations to join a group. Below are two examples of comments (punctuation is preserved) that require the removal of URL links and invitations to join groups

(1) *Greetings!Using the "blog this" function above your picture, we have linked your picture to our WordPress blog <a href=http://osiddhartha.wordpress.com>SIDDHARTHA</a>*

(2) *Hi, I m a member of a group called <a href=groups/fiveflickrfavs>Five Flickr Favs</a>, and we'd love to have your photo added to the group*

Photo comments can be written in different languages or may contain mixtures of several languages. In order to analyze parts of speech usage, we had to apply a POS-tagger. Since there is no universal POS-tagger that can work on any language and we don't know exactly what languages are used in comments, we decided to remove all comments that are not written in English. For this, we used the TextCat language guesser[4]. The following languages were identified while scanning all the comments: *Polish, English, Swedish, Slovenian, Slovakian, Danish, Italian, Dutch, Spanish, French, German, Finnish, Albanian, Hungarian, Norwegian, Unknown.*

After removing all non-English comments, we were left with 4214 commented photos in Krakow, 4098 commented photos in Warsaw, 56 commented photos in Wisla, 311 commented photos in Auschwitz and 179 commented photos in Dachau.

In the next step, we applied the Stanford POS Tagger[5] on the English comments.

IV. METHOD

A. Definitions

Different terminology definitions are provided in the sentiment and opinion analysis literature. The terminology used in this paper mostly sticks to the definitions given in [15], but makes a clear distinction between opinions and sentiments. The important terms and their definition for this paper:

- **Photo Feature:** Nouns that describe the photo features - attributes, components or characteristics of the photo, e.g. "shot", "photo", "color", "composition", "light". Photo features in our case are usually directly related to the quality of the photo. It is common to distinguish between explicit and implicit features, i.e. features that are mentioned in a sentence and features that are not explicitly mentioned but implicitly referenced.
- **Orientation:** The semantic orientation of a word or a comment as a binary categorical variable with the parameter values "negative" and "positive" (the third possibility "neutral" is omitted in our scenario).
- **Orientation Strength:** The numerical strength of the orientation value ranging from 0 to ∞ in absolute numbers, whereas negative orientations are indicated by the algebraic sign "-".
- **Photo Opinion (PO):** Negative or positive user statements, that clearly refer to photo features of a certain photo, are summarized as the respective photo opinion. They express the users' opinions on the technical and artistic photo quality. For simplicity, we will only speak of opinions when we refer to photo opinions.
- **General Sentiment (GS):** Negatively or positively connoted user statements that cannot be attributed to a photo feature. As implied by the denomination, the general sentiment shall capture orientation statements that have a broader nature than opinions, i.e. sentiments and emotions that are evoked by the photo content. For simplicity, we will only speak of sentiments when we refer to general sentiments.
- **Ambiguity:** Not all users have concordant opinions or sentiments when commenting on a photo. However, at the end one single PO and GS value is computed for each photo that does not account for potential disagreements among users. Accordingly, for both PO and GS ambiguity values are provided that indicate perfect agreement (0.0) or complete disagreement (1.0) among users, as well as arbitrary real values in between.

B. Corpus-based lexicon generation

Opinion mining is heavily dependent on an opinion lexicon. The two common approaches for lexicon generation are dictionary-based and corpus-based ones. The former is based on bootstrapping a seed of opinion words from dictionaries like WordNet, SentiWordNet or Wikipedia, the latter is based on the corpus and, thus, inherently domain dependent.

We applied a corpus-based lexicon generation due to several reasons:

- We want to generate a new lexicon in the domain of photo comments since currently, at least to our knowledge, no such lexicon is publicly available
- Dictionaries like SentiWordNet may supply only a binary opinion orientation, while our task is to model opinion orientations on a real-value scale
- We want to investigate statistical properties of words used for commenting

[4] http://odur.let.rug.nl/~vannoord/TextCat/

[5] http://nlp.stanford.edu/software/tagger.shtml

It was shown in past research that there is a strong correlation between the presence of adjectives and opinions [19], [20]. Indeed, a careful analysis of photo comments showed that people often use short sentences like “Great photo”, “Nice picture”, “Sad place” to express their opinions or sentiments. The analysis also showed that the number of positive adjectives used in photo comments is higher than for negative ones and, overall, the number of positive comments is much higher than the number of negative comments. Any lexicon of positive and negative words will show that the words “great” and “nice” are positive. However, it is difficult to estimate which of these two words is “more positive than the other” using lexical features alone. For that reason, we decided to apply a measure, which is similar to the TF-IDF (Term Frequency-Inverse Document Frequency) measure used in information retrieval and text mining [18]. The idea is that standard opinion or sentiment words that are used frequently by majority of people receive lower scores than words that are used infrequently. In order to acquire word distributions, we extracted adjectives and nouns from the corpus, counted their occurrences in the five selected regions separately, and sorted them according to their occurrence from the highest to the lowest. Nouns were extracted in order to learn what words are commonly used as photo features. We used Yago-Naga stemmer[6] to convert all nouns into a singular form.

In order to minimize the bias of some active commenters, we counted word occurrence only once for each person. The reason why we selected five separate regions is that word occurrences may differ due to different topicalities. Moreover, the number of commented photos is different from region to region and the word distribution would inevitably be biased towards words used in regions with a lot of comments. Table II summarizes frequencies of adjectives in five areas.

An inspection of the results shows the following interesting patterns: The words *great, nice* and *beautiful* are the most frequent and equally ranked adjectives in all five regions, 33% of the adjectives are found within the 20 most frequent adjectives in every region, 58% of the adjectives are found in at least one region and 42% of frequent adjectives are found only in one region. This suggests that people use many common opinion words even if the context of photos is very different (Dachau concentration camp and Nature).

Another interesting finding is that the distribution of adjectives in all five regions can be described by Zipf’s law, which stays that if f is the frequency of a word in the corpus and r is the rank, then

$$f = \frac{k}{r} \qquad (1)$$

where k is a constant for the corpus. When we take the logarithm of both sides, we obtain a linear function with the slope of -1. The slope coefficients we obtained for Krakow is -1.138, Warsaw: -1.136 , Auschwitz: -0.988 and Dachau: -0.95 (Wisla was excluded because it does not have enough words for a reliable slope estimation). Apart the statistical properties of the word distribution, Zipf’s law can be also explained in terms of “least effort” principle: [30] *the tension between the goal of the speaker to minimize production efforts by using only few words very frequently and the goal of the listener to minimize perceptual confusion by having a large vocabulary of distinct words.*

TABLE II
20 MOST FREQUENT ADJECTIVES AND THEIR FREQUENCY IN FIVE SELECTED AREAS. WORDS THAT ARE COMMONLY USED IN FIVE REGIONS ARE COLORED IN YELLOW, IN FOUR REGIONS - GRAY, IN THREE - PINK, IN TWO - GREEN, IN ONE - WHITE

Krakow	Warsaw	Wisla	Auschwitz	Dachau
great,1469	great,1403	great,26	great,129	great,65
nice,864	nice,856	nice,14	nice,61	nice,29
beautiful,829	beautiful,756	beautiful,13	beautiful,57	beautiful,29
good,311	good,306	lovely,8	good,42	fantastic,17
wonderful,271	wonderful,257	awesome,7	powerful,31	powerful,14
lovely,238	amazing,215	amazing,6	amazing,30	excellent,14
amazing,202	cool,191	cute,6	impressive,27	awesome,11
interesting,200	lovely,184	good,5	sad,24	amazing,11
cool,196	fantastic,181	such,3	wonderful,24	sad,10
fantastic,168	excellent,174	excellent,3	excellent,22	impressive,10
excellent,153	interesting,173	wonderful,3	fantastic,18	very,8
awesome,137	awesome,166	right,2	awesome,17	interesting,8
very,129	very,133	pretty,2	interesting,16	such,7
perfect,116	perfect,104	cool,2	very,15	wonderful,7
gorgeous,74	gorgeous,79	new,2	strong,13	dark,7
such,71	cute,78	very,2	many,12	lovely,6
cute,68	little,61	fantastic,2	same,11	cool,6
much,62	such,55	terrific,1	white,11	scary,6
little,58	stunning,47	fierce,1	such,11	dramatic,6
black,55	impressive,45	perfect,1	cool,11	good,6

C. The Adjective Weighting Model

Having defined statistical and linguistic interpretations of the distributions of adjectives in the photo comments corpus, we are now ready to propose an adjective weighting model for opinion orientation.

We define the word opinion orientation w_{oo} using the principles of word relevance as defined in the TF-IDF measure and word distribution properties of Zipf’s law as follows:

$$w_{oo} = orientation(w) * log(\frac{f_{w,r=1}}{f_w} + 1) \qquad (2)$$

where $orientation(w)$ is a function which assigns 1 if the word w is positive and -1 if it is negative, $f_{w,r=1}$ is the frequency of the word having the rank 1 (Equation 1) and f_w is the frequency of the word w in the whole corpus.

The difference between TF-IDF and our approach is that the importance of the word in TF-IDF is measured for every word independently, while opinion orientation score is calculated relative to the most frequent word in the corpus. Thus, if the most frequent word is “great” with frequency of 1469 (see Table II) and the word ranked second is “nice” with the frequency of 864, “great” will get a score of 0.30 (log (1469/1469 + 1)), while the score of “nice” will be 0.43 (log

[6] http://www.mpi-inf.mpg.de/yago-naga/

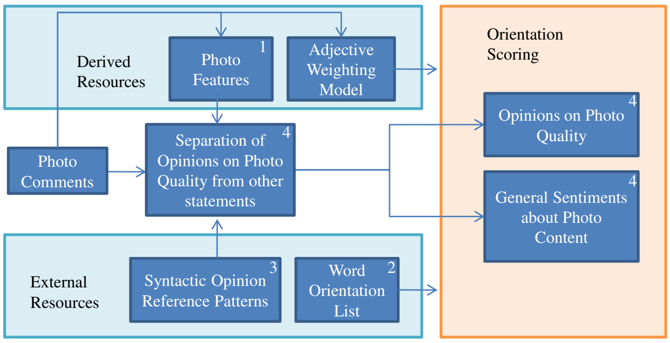

Derived Resources
Photo Features 1
Adjective Weighting Model
Orientation Scoring
Opinions on Photo Quality 4
General Sentiments about Photo Content 4
Photo Comments
Separation of Opinions on Photo Quality from other statements 4
External Resources
Syntactic Opinion Reference Patterns 3
Word Orientation List 2

Fig. 1. Overview about the interdependence of the different core text analysis processes. The numbers correspond to the paragraphs in Section IV-D, where details are provided.

(1469/864) + 1). One is added to *log* to avoid zero score of the most frequent word.

We should note, that the word frequency in the Equation 2 is absolute and can be applied to five regions separately. In order to make a global model that takes into account different word distributions, we need to find the relative order of all words from five regions. We proceeded it as follows:

- We calculated a ratio $\frac{f_{w,r=1}}{f_w}$ for every word
- An average of ratios for every word was calculated taking these ratios for the same word $w_{i,n}$ from every region n
- If the word $w_{i,n}$ was not found among the lexicon of the region n, its ratio was assumed to have the ratio of the last word in the lexicon of the region n

After building a weighted ratio for every word, we applied Equation 2 to obtain the global adjective weighting model.

D. Automatic Opinion and Sentiment Analysis

The automatic opinion and sentiment analysis consists of several interdependent steps as outlined in Figure 1. The analysis relies on both resources derived from the photo comment corpus itself and external resources. The details are provided in the following subsections.

1) The Photo Features: In order to determine which opinions relate to the photo, first a list of photo features had to be compiled. For this purpose a term extraction method was created that exploits certain characteristics of photo features: (1) such features usually correspond to nouns, (2) such features should not depend significantly on the photo location, and (3) such features should be frequent in photo comments. Consequently, (1) all nouns where extracted, that (2) appeared in photo comments of at least 4 out of 5 locations, and finally (3) the 100 most frequent among these terms were extracted as candidate photo features. The list was then manually revised and finally, 60 out of these nouns where considered in the analysis as photo features. The top 10 frequent nouns present in at least 4 locations were, in decreasing frequency order, “shot”, “photo”, “color”, “composition”, “light”, “picture”, “capture”, “love”, “image”, “work”. Here, “love” is one example that was manually deleted. In this case we could observe that high frequency of the noun “love” was due to a repeated error of the part-of-speech tagger, when occurrences of the verb “love” in very short sentences (e.g. “Love it!”) were misclassified as nouns.

Implicit features: If sentences were shorter than 6 words and did not contain a noun, it was assumed that comments implicitly meant the photo (e.g. “I love it.”, “Well done.”, “Very nice.”).

2) The Word Orientation List: As already mentioned, a manually enhanced version of the widely used Internet General Inquierer lexicon [31] was applied. It was used to determine the orientation of the word and incorporate it into Equation 2, i.e. +1 for positive, -1 for negative and 0 for neutral words (not contained in the orientation list). Before, words were reduced to their base form with Kuhlen’s algorithm [32], in order to increase the number of matches.

All words not contained by the adjective weighting model of Section IV-C, because they either did not appear in the photo comments or belong to a different part-of-speech category, were allocated the weight 1.

3) Syntactic Opinion Reference Patterns: In order to detect references of opinion words to photo features, a set of syntactic opinion reference patterns was defined, based on linear word order part-of-speech sequences[7]. A very simple example is the pattern “JJ NN” which stands for an adjective (JJ) directly followed by a noun (NN). In this case we can be sure, that the adjective refers to the noun. Hence, if the noun is a photo feature then the adjective and its orientation can be assigned to this feature. While in theory recursive patterns of arbitrary length (e.g. JJ* NN) are possible in natural language, in practice such patterns do not appear to a noteworthy extent. We could observe that the limited pattern set we defined, covers the vast majority of cases. The whole pattern set is provided in Figure 2. One main advantage is that the patterns encode the available linguistic knowledge about opinion references without requiring the time-consuming parsing of a full syntax structure tree or a typed dependencies graph. Our syntactic reference patterns cover most of the cases that other approaches detect with dependency parses. This is due to the fact that in English adjectives are usually very close to the nouns they refer to. Only very exceptional and infrequent cases like relational phrases ”‘the photo, that shows a tree, is really nice”’ cannot be resolved by our means. In case of verbs, our approach is not able to distinguish explicitly whether the feature is the subject or the object of the verb. In our tests, however, we could observe that this is not a problem. Verbs that express opinions (”‘to hate/to like”’) cannot have a photo feature as subject and in cases in which they are objects, they are covered by our analysis patterns. In addition, our method is less error-prone than dependency parsing, especially when applied to less formalized and clean writing, as in user-generated content.

[7] The used part-of-speech tags follow the Penn Treebank Tag-set definition: http://www.comp.leeds.ac.uk/ccalas/tagsets/upenn.html

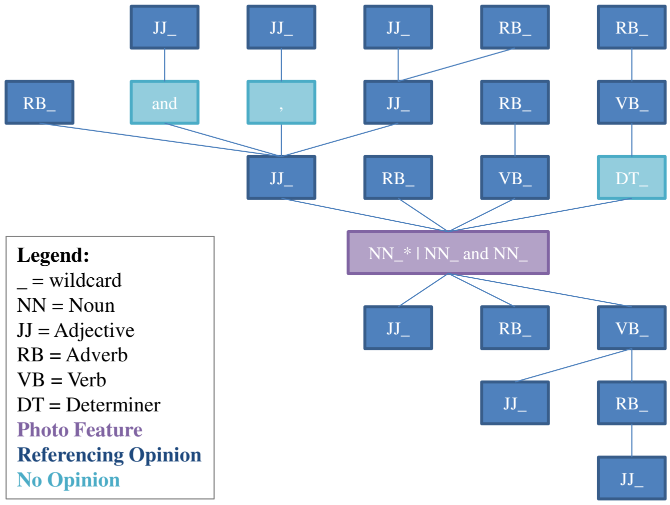


Fig. 2. Syntactic Opinion Reference Patterns. Word order patterns go from top (before photo features) to bottom (after photo features), the level indicates the exact position.

4) Identification and Separation of Photo Opinions and General Sentiments: One crucial point of the automatic text analysis is the detection and separation of (1) opinions about the photo quality (PO) and (2) general sentiments expressed about the photo content (GS).

The first part (1) is based on the extraction of photo features and the mapping of opinion statements to photo features. The described set of syntactic opinion reference patterns was applied for this mapping. For each photo feature in a sentence, all words were extracted that describe the feature according to one of the syntactic opinion reference patterns. The orientation scores of these words were then summed up to yield a photo opinion value. In this process, a simple heuristic is used to invert the orientation of negated words.

Accordingly, step (2) is based on all sentiment expressions that could not be attributed to photo features during step (1). This means that all words not referring to photo features were considered and their orientation scores summed up to yield a general sentiment value.

It should be noted that general sentiments only in very rare cases are falsely classified as photo opinions, whereas the contrary could be observed more frequently, due to different reasons (missing photo feature, implicitness).

Additionally, in both steps the ambiguity of comments is analyzed. This implies investigating whether different users express different opinions or sentiments on the same photo. The output of steps (1) and (2) are an opinion and sentiment value for each user comment. The ambiguity is then calculated separately for the opinions about a photo and the sentiments about a photo. Equation 3 shows how the opinion ambiguity value is calculated for a photo, based on the number of user comments with positive opinions ($\#pos$) and the number of user comments with negative opinions ($\#neg$). For the sentiments this works analogously.

$$amb = \begin{cases} 0 & \textit{if } (\#pos = 0) \vee (\#neg = 0), \\ \frac{Min(\#pos,\#neg)}{Max(\#pos,\#neg)} & \textit{else.} \end{cases} \qquad (3)$$

E. Statistical Proof of Concept

Because of the lack of an appropriate Gold Standard it is not easy to evaluate the sentiment and opinion analysis. Instead, we try to gain evidence for the suitability by performing statistical analysis. Table III shows mean and standard deviation values of opinions and sentiments for the different regions. As expected, the relative difference of mean photo opinion values between different locations is much smaller then that of mean general sentiments. This is in accord with our expectations, because the general sentiment is much more dependent on the location than the photo quality. There is a certain correlation between photo quality and general sentiments, which could be due to the fact that both cannot be separated unambiguously in all cases. However, the two concentration camp memorials Ausschwitz and Dachau, as expected, have very low general sentiment values and the two popular tourist places Warsaw and Krakow are allocated much higher values (even the same mean). Wisla, which we anticipated to be a rather neutral place, lies between the extrema with its general sentiment values. All in all, the statistics indicate that a reasonable separation of opinions and sentiments could be achieved.

Location	Op. Mean	Op. Stdv.	Sent. Mean	Sent. Stdv
Auschwitz	0.827	1.847	0.318	1.202
Dachau	0.776	2.16	0.268	1.026
Krakow	1.003	2.068	0.736	1.544
Warsaw	0.976	2.618	0.736	2.483
Wisla	0.945	1.209	0.516	1.314

TABLE III
AVERAGE AND STANDARD DEVIATION OF OPINION AND SENTIMENT SCORES

V. APPLICATION

In this section we demonstrate the desktop application that combines Google Earth[8], the custom engine built on top of Google Earth [22], and the navigation and filtering toolbox that implements the method for opinion and sentiment analysis of photo comments.

A. Usage Scenario

Our goal is to enrich the user experience by improving photo navigation in a selected region and adding more options for exploring the area. Google Earth has become a favorite platform among Internet users for map navigation and exploration. Google Earth contains different layers that include points of interest, photos, etc. At any time the user can navigate to a specific region in the World and explore the points of interest or photos that were taken there by tourists.

[8] http://earth.google.com/

The difficulty is that photos are displayed in Google Earth as small rectangular thumbnails. The actual image is displayed only when the user clicks on the thumbnail. To gain an actual view of the place, the user has to click on the thumbnails many (several) times and search through different photos. Similarly to Google Earth, the Flickr web site allows the navigation to a particular place using the provided search field. The web page will display large image thumbnails in a sequential order with an overall statistic of how many images were found. For example, for May 25, 2010, Flickr reports $256,827$ results when Warsaw is used as a keyword. Flickr allows sorting the results using three options *Relevant, Recent* and *Interesting*. In addition, Flickr allows locating photos on the map using its WorldMap[9] by providing the location and optionally the category (architecture, urban, forest). By issuing a search for Warsaw, Flickr found $73,174$ geotagged photos, displayed as image thumbnails on a horizontal strip and sortable according to two parameters: *Interesting* and *Recent*. The *relevant* option allows searching for images that contain the search keyword in their titles, while the *interesting* option is based on the non-disclosed algorithm that takes into account such features like *number of views, comments, etc*.

In our application, we implement two main features that are the core of the algorithmic part of the paper, namely *opinion* and *sentiment*, three derived features, namely *number of sentences in comments, opinion and sentiment ambiguities* and *the number of positive and negative opinion words in comments*. Additionally, we included an additional feature, which is part of the downloaded metadata *the number of times the photo was viewed*. The application has two main views and will be described in the following subsections.

B. Photo sorting

The control panel displayed in Figure 3(a) allows the user to receive information about the boundaries of the selected region (label 1). When the user changes the boundaries by manipulating the Google Earth map, the application connects to the server and updates statistical information related to the photos (label 2). In particular, the following statistics are sent by the server: total photos in the region, minimum and maximum opinion and sentiment scores, minimum and maximum number of sentences in comments, positive and negative opinion words and opinion ambiguity, minimum and maximum number of viewed photos.

The central part of the control panel contains a number of filtering options (label 3): filtering by *opinion*, *sentiment*, *sentences*, *ambiguity*, *number of positive and negative words*, etc. In addition, two *quantity filters* (label 4) allow limiting the number of displayed photos on the map view and in the control panel.

When one of the filtering options is invoked, the request is sent via REST protocol to the server along with all relevant information. The server generates two types of responses that are sent as one string to the client. The first response is formatted as Keyhole Markup Language (KML), an XML-based language for the visualization of geographic entities and the one which is used by Google Earth. In our case it contains a photo URL and all the relevant information about the photos (opinion and sentiment scores as well as comments). The KML file is extracted from the response and delegated to the underlying Google Earth engine for visualization. The second response is used by the control panel to show N-top photos (label 5) filtered by one of the provided options.

If the user clicks on one of the photos, the information about the selected photo is displayed on the left side including the coordinate of the photo, comments and scores (label 6). A double-click on the photo positions the map around that photo on the map view.

C. Map navigation

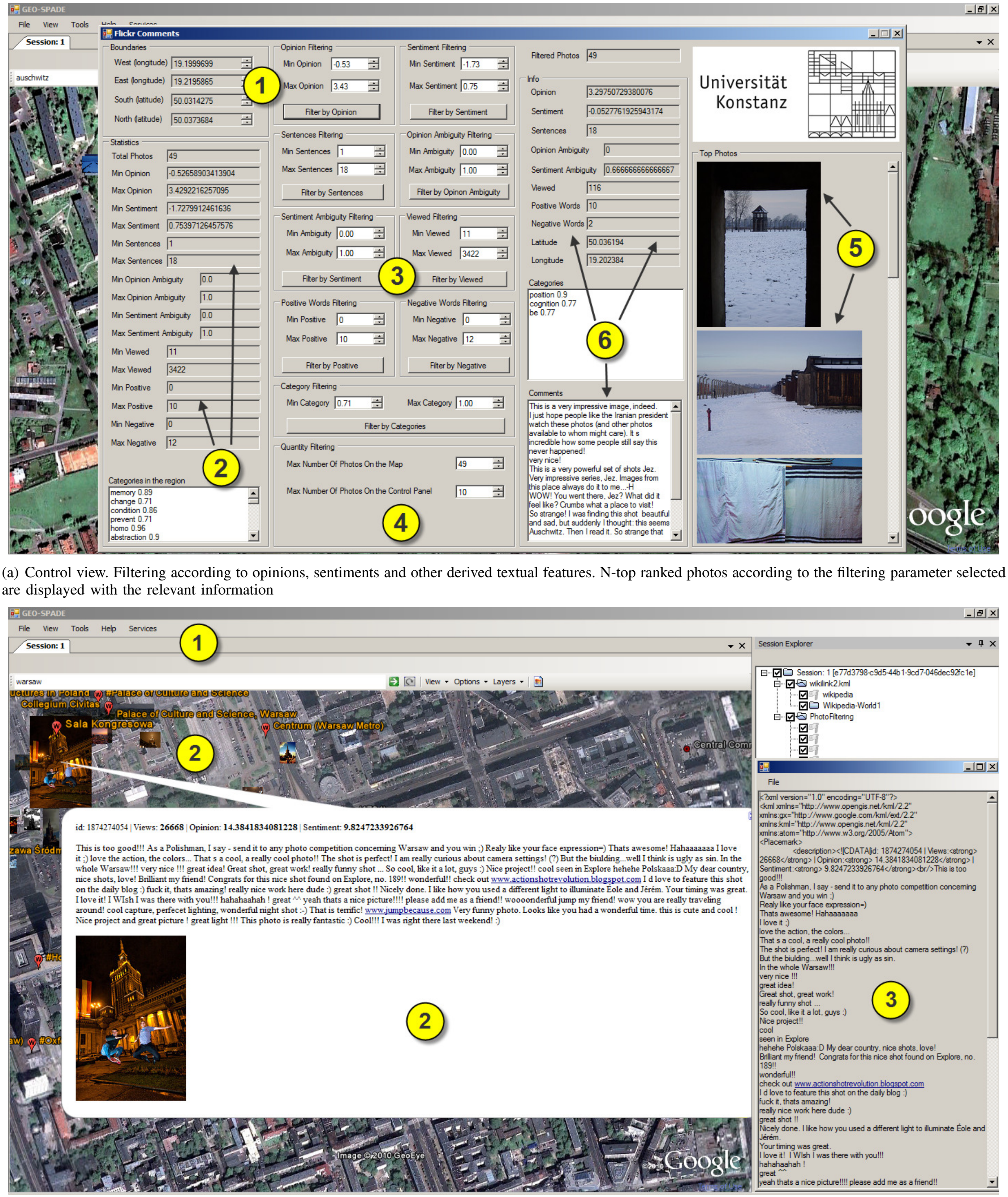
Map navigation (Figure 3(b), label 1) allows exploring the photos using the map view after they were filtered by one of the available scores. The exploration is similar to the functionality provided in the stand-alone Google Earth version or Flickr MapView. However, the difference is that the thumbnail of the image is directly visible on the map. This can save time as it does not require clicking on every thumbnail to see the underlying images. When the image thumbnail is clicked, the large image is displayed along with its comments and scores (see Figure 3(b), label 2). Since the data interchange format is XML-based, information about the whole set of filtered photos or an individual photo can be saved into the file and later visualized in any application that supports KML (label 3).

VI. Conclusions

This paper introduces a practical unsupervised approach for improving the user experience during the exploration of (geotagged) photos on photo sharing web sites by filtering and sorting photos using opinions and sentiments expressed in user comments written for uploaded photos.

Our approach is able to identify two types of opinions from the comments: opinions that are related to the photo quality and general sentiments or moods expressed towards the objects shown on the photo. Unlike most of the existing approaches in which binary (negative or positive) opinion orientation is used, we model opinion orientation using a real-valued scale. Using linguistic features, we build a finite lexicon of adjectives and calculate their opinion strength using a word importance paradigm borrowed from the information retrieval field. The opinion orientation (negative or positive sign) is calculated using a predefined lexicon of positive and negative opinion-bearing words. The identification and separation of photo opinions is based on a semi-automatic method for photo feature extraction and a set of predefined syntactic opinion reference patterns. The overall opinion and sentiment scores for a photo is the cumulative sum of all scores in the comments. This allows a dynamic update of scores if new comments are written for the photo.

[9] http://www.flickr.com/map/



Boundaries
West (longitude) 19.1999699
East (longitude) 19.2195865
South (latitude) 50.0314275
North (latitude) 50.0373684

Statistics
Total Photos 49
Min Opinion -0.52658903413904
Max Opinion 3.4292216257095
Min Sentiment -1.7279912461636
Max Sentiment 0.75397126457576
Min Sentences 1
Max Sentences 18
Min Opinion Ambiguity 0.0
Max Opinion Ambiguity 1.0
Min Sentiment Ambiguity 0.0
Max Sentiment Ambiguity 1.0
Min Viewed 11
Max Viewed 3422
Min Positive 0
Max Positive 10
Min Negative 0
Max Negative 12

Categories in the region
memory 0.89
change 0.71
condition 0.86
prevent 0.71
homo 0.96
abstraction 0.9

Opinion Filtering
Min Opinion -0.53
Max Opinion 3.43
Filter by Opinion

Sentiment Filtering
Min Sentiment -1.73
Max Sentiment 0.75
Filter by Sentiment

Sentences Filtering
Min Sentences 1
Max Sentences 18
Filter by Sentences

Opinion Ambiguity Filtering
Min Ambiguity 0.00
Max Ambiguity 1.00
Filter by Opinion Ambiguity

Sentiment Ambiguity Filtering
Min Ambiguity 0.00
Max Ambiguity 1.00
Filter by Sentiment

Viewed Filtering
Min Viewed 11
Max Viewed 3422
Filter by Viewed

Positive Words Filtering
Min Positive 0
Max Positive 10
Filter by Positive

Negative Words Filtering
Min Negative 0
Max Negative 12
Filter by Negative

Category Filtering
Min Category 0.71
Max Category 1.00
Filter by Categories

Quantity Filtering
Max Number Of Photos On the Map 49
Max Number Of Photos On the Control Panel 10

Filtered Photos 49

Info
Opinion 3.29750729380076
Sentiment -0.0527761925943174
Sentences 18
Opinion Ambiguity 0
Sentiment Ambiguity 0.666666666666667
Viewed 116
Positive Words 10
Negative Words 2
Latitude 50.036194
Longitude 19.202384

Categories
position 0.9
cognition 0.77
be 0.77

Comments
This is a very impressive image, indeed.
I just hope people like the Iranian president watch these photos (and other photos available to whom might care). It s incredible how some people still say this never happened!
very nice!
This is a very powerful set of shots Jez.
Very impressive series, Jez. Images from this place always do it to me...-H
WOW! You went there, Jez? What did it feel like? Crumbs what a place to visit!
So strange! I was finding this shot beautiful and sad, but suddenly I thought: this seems Auschwitz. Then I read it. So strange that

(a) Control view. Filtering according to opinions, sentiments and other derived textual features. N-top ranked photos according to the filtering parameter selected are displayed with the relevant information

id: 1874274054 | Views: **26668** | Opinion: **14.3841834081228** | Sentiment: **9.8247233926764**

This is too good!!! As a Polishman, I say - send it to any photo competition concerning Warsaw and you win ;) Realy like your face expression=) Thats awesome! Hahaaaaaaa I love it ;) love the action, the colors... That s a cool, a really cool photo!! The shot is perfect! I am really curious about camera settings! (?) But the biulding...well I think is ugly as sin. In the whole Warsaw!!! very nice !!! great idea! Great shot, great work! really funny shot ... So cool, like it a lot, guys :) Nice project!! cool seen in Explore hehehe Polskaaa:D My dear country, nice shots, love! Brilliant my friend! Congrats for this nice shot found on Explore, no. 189!! wonderful!! check out www.actionshotrevolution.blogspot.com I d love to feature this shot on the daily blog :) fuck it, thats amazing! really nice work here dude :) great shot !! Nicely done. I like how you used a different light to illuminate Éole and Jérém. Your timing was great. I love it! I WIsh I was there with you!!! hahahaahah ! great ^^ yeah thats a nice picture!!!! please add me as a friend!! woooonderful jump my friend! wow you are really traveling around! cool capture, perfecet lighting, wonderful night shot :-) That is terrific! www.jumpbecause.com Very funny photo. Looks like you had a wonderful time. this is cute and cool ! Nice project and great picture ! great light !!! This photo is really fantastic :) Cool!!! I was right there last weekend! :)

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2"
xmlns:gx="http://www.google.com/kml/ext/2.2"
xmlns:kml="http://www.opengis.net/kml/2.2"
xmlns:atom="http://www.w3.org/2005/Atom">
<Placemark>
        <description><![CDATA[id: 1874274054 | Views:<strong>
26668</strong> | Opinion:<strong> 14.3841834081228</strong> |
Sentiment:<strong> 9.8247233926764</strong><br/>This is too
good!!!
As a Polishman, I say - send it to any photo competition concerning
Warsaw and you win ;)
Realy like your face expression=)
Thats awesome! Hahaaaaaaa
I love it ;)
love the action, the colors...
That s a cool, a really cool photo!!
The shot is perfect! I am really curious about camera settings! (?)
But the biulding...well I think is ugly as sin.
In the whole Warsaw!!!
very nice !!!
great idea!
Great shot, great work!
really funny shot ...
So cool, like it a lot, guys :)
Nice project!!
cool
seen in Explore
hehehe Polskaaa:D My dear country, nice shots, love!
Brilliant my friend!  Congrats for this nice shot found on Explore, no.
189!!
wonderful!!
check out www.actionshotrevolution.blogspot.com
I d love to feature this shot on the daily blog :)
fuck it, thats amazing!
really nice work here dude :)
great shot !!
Nicely done. I like how you used a different light to illuminate Éole and
Jérém.
Your timing was great.
I love it!  I WIsh I was there with you!!!
hahahaahah !
great ^^
yeah thats a nice picture!!!! please add me as a friend!!
```

(b) Google Earth view (label 1). Clickable photo thumbnails are displayed on the map (label 2). The results of visualization can be seen and saved in KML format (label 3)

Fig. 3. Google Earth-based application for photo search using opinion and sentiments scores

We implemented a prototype desktop Google Earth-based system that implements the method described in the paper. It allows the exploration of geotagged photos using opinion and sentiment scores combined with the visualization of photos on the map.

In our future work, we shall concentrate on the improvement of the score assignment algorithm and work on multi-lingual solutions.

ACKNOWLEDGEMENTS

This work was partially funded by the German Research Society (DFG) under grant GK-1042 (Research Training Group "Explorative Analysis and Visualization of Large Information Spaces"). The authors would like to thank Sergey Pulnikov and Johannes Lichtenberger for their help in the implementation.

REFERENCES

[1] M. Gamon, "Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis," in *Proceedings of the International Conference on Computational Linguistics (COLING)*, 2004.

[2] P. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 417–424.

[3] F. Salvetti, S. Lewis, and C. Reichenbach, "Automatic opinion polarity classification of movie reviews," *Colorado research in linguistics*, vol. 17, no. 1, 2004.

[4] B. Ohana and B. Tierney, "Sentiment classification of reviews using SentiWordNet," in *9th. IT & T Conference*, 2009, p. 13.

[5] M. Sahlgren, J. Karlgren, and G. Eriksson, "SICS: Valence annotation based on seeds in word space," in *Proceedings of the 4th International Workshop on Semantic Evaluations*, 2007, pp. 296–299.

[6] K. Dave, S. Lawrence, and D. Pennock, "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews," in *Proceedings of the 12th international conference on World Wide Web*, 2003, p. 528.

[7] G. Qiu, B. Liu, J. Bu, and C. Chen, "Expanding domain sentiment lexicon through double propagation," in *Proceedings of the 21st international jont conference on Artifical intelligence*, 2009, pp. 1199–1204.

[8] A. Devitt and K. Ahmad, "Sentiment polarity identification in financial news: A cohesion-based approach," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, vol. 45, no. 1, 2007, p. 984.

[9] N. O'Hare, M. Davy, A. Bermingham, P. Ferguson, P. Sheridan, C. Gurrin, and A. Smeaton, "Topic-dependent sentiment analysis of financial blogs," in *Proceeding of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, 2009, pp. 9–16.

[10] S. Das and M. Chen, "Yahoo! for Amazon: Sentiment extraction from small talk on the web," *Management Science*, vol. 53, no. 9, pp. 1375–1388, 2007.

[11] A. Fahrni and M. Klenner, "Old wine or warm beer: Target-specific sentiment analysis of adjectives," in *AISB 2008 Convention Communication, Interaction and Social Intelligence*, vol. 1, 2008, p. 60.

[12] P. Chesley, B. Vincent, L. Xu, and R. Srihari, "Using verbs and adjectives to automatically classify blog sentiment," *Training*, vol. 580, no. 263, p. 233, 2006.

[13] V. Hatzivassiloglou and K. McKeown, "Predicting the semantic orientation of adjectives," in *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, 1997, p. 181.

[14] S. Argamon, K. Bloom, A. Esuli, and F. Sebastiani, "Automatically determining attitude type and force for sentiment analysis," *Human Language Technology. Challenges of the Information Society*, pp. 218–231, 2009.

[15] B. Liu, "Sentiment Analysis and Subjectivity," *Handbook of Natural Language Processing, Second Edition,(editors: N. Indurkhya and FJ Damerau)*, 2009.

[16] A. Drake, E. Ringger, and D. Ventura, "Sentiment Regression: Using Real-Valued Scores to Summarize Overall Document Sentiment," in *2008 IEEE International Conference on Semantic Computing*, 2008, pp. 152–157.

[17] V. Subrahmanian and D. Reforgiato, "AVA: Adjective-Verb-Adverb Combinations for Sentiment Analysis," *IEEE Intelligent Systems*, vol. 23, no. 4, pp. 43–50, 2008.

[18] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information processing & management*, vol. 24, no. 5, pp. 513–523, 1988.

[19] J. Wiebe, R. Brace, and T. O'Hara, "Development and use of a gold-standard data set for subjectivity classifications," in *Annual meeting-association for computational linguistics*, vol. 37, 1999, pp. 246–253.

[20] J. Wiebe, "Learning subjective adjectives from corpora," in *Proceedings of the National Conference on Artificial Intelligence*, 2000, pp. 735–741.

[21] R. Baayen, *Word Frequency Distributions*. Springer, 2002.

[22] S. Kisilevich, D. Keim, and L. Rokach, "A generic google earth-based framework for analyzing and exploring spatio-temporal data," in *12th International Conference on Enterprise Information Systems*, 2010.

[23] J. Kamps, M. Marx, R. Mokken, and M. De Rijke, "Using WordNet to measure semantic orientation of adjectives," in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004*, vol. 4, 2004, pp. 1115–1118.

[24] C. Fellbaum, *WordNet: An electronic lexical database*. MIT press Cambridge, MA, 1998.

[25] A. Esuli and F. Sebastiani, "SentiWordNet: A publicly available lexical resource for opinion mining," in *Proceedings of LREC*, vol. 6, 2006.

[26] X. Ding, B. Liu, and P. Yu, "A holistic lexicon-based approach to opinion mining," in *Proceedings of the international conference on Web search and web data mining*, 2008, pp. 231–240.

[27] D. Oelke, M. Hao, C. Rohrdantz, D. A. Keim, U. Dayal, L.-E. Haug, and H. Janetzko, "Visual opinion analysis of customer feedback data," in *VAST '09: Proceedings of the 2009 IEEE Symposium on Visual Analytics Science and Technology*, 2009, pp. 187–194.

[28] A.-M. Popescu and O. Etzioni, "Extracting product features and opinions from reviews," in *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Morristown, NJ, USA, 2005, pp. 339–346.

[29] E. Riloff and J. Wiebe, "Learning extraction patterns for subjective expressions," in *Proceedings of the 2003 conference on Empirical methods in natural language processing*, 2003, pp. 105–112.

[30] M. Baroni, "Distributions in text," *Corpus linguistics: An international handbook*, vol. 2, 2009.

[31] V. Buvac, "Internet general inquirer," http://www.webuse.umd.edu:9090/.

[32] R. Kuhlen, *Experimentelle Morphologie in der Informationswissenschaft*. Verlag Dokumentation, Munchen, 1977.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 429–433

ISBN 978-83-60810-27-9
ISSN 1896-7094

LEXiTRON-Pro Editor: an Integrated Tool for developing Thai Pronunciation Dictionary

Supon Klaithin, Patcharika Chootrakool, Krit Kosawat
Human Language Technology Laboratory (HLT)
National Electronics and Computer Technology Center (NECTEC)
112 Thailand Science Park, Phahon Yothin Rd., Pathumthani 12120, Thailand
Email: {supon.klaithin, patcharika.chootrakool, krit.kosawat}@nectec.or.th

Abstract—**Pronunciation dictionary is a crucial part for both Text-To-Speech and Automatic Speech Recognition systems. In this paper, we propose a tool to easily create and edit Thai pronunciation dictionary, called LEXiTRON-Pro Editor. This tool integrates Thai word segmentation, Thai Grapheme-to-Phoneme (G2P) conversion, and database system with statistics. It automatically proposes a word's pronunciation to users by 1 of the 3 options in the successive order: the pronunciation from LEXiTRON-Pro database, the pronunciation combined from syllables with highest probability, and the pronunciation from Thai G2P. However, users can switch to another option or even directly input their own pronunciation with an easy interface editor. Our LEXiTRON-Pro database contains initially 105,129 unique words and 24,736 unique syllables with pronunciations. Compared to the previous version, our new program can reduce the process of dictionary development from 5 to only 1 step and the number of tools used by linguists from 3 to only 1. Moreover, our experiment shows that the time consumption and the number of ungenerable words are significantly reduced while the pronunciation accuracy is considerably improved.**

I. Introduction

PRONUNCIATION dictionary is an essential component for both Thai Text-To-Speech (TTS) and Automatic Speech Recognition (ASR) systems. However, the procedure of developing a pronunciation dictionary is quite complicated and requires a lot of man-hours from linguists. To help them, many recent studies have focused on the Grapheme-to-Phoneme (G2P) conversion system to automatically generate phonemes of words and create easier the pronunciation dictionary. Several approaches have been proposed in the development of G2P to improve the phonetic transcriptions, such as Decision Tree, Statistical method, Pronunciation-by-analogy, and Rule-based approach. Some systems even combine various approaches together to increase efficiency. For example, Tarsaku *et al.* [1] had developed Probabilistic Generalized LR (PGLR) by combining Ruled-based and Decision Tree approaches which can achieve 72.87% of transcription accuracy. The Example-based Grapheme-to-Phoneme (EBG2P) conversion approach, developed by Paisarn Charoenpornsawat and Tanja Schultz [2], which generates pronunciations from syllables found in the training corpus, can reach 80.99% of transcription accuracy.

However, it is found that using only G2P to create the pronunciation dictionary has caused several problems. Firstly, Thai G2P cannot correctly deal with ambiguous strings that contain more than one possible segmentation. For example, "ตากลม" has two patterns of segmentation and accordingly two pronunciations, i.e. "ตาก|ลม|" (t-aa-k^-1| l-o-m^-0| : to be exposed to the wind) and "ตา|กลม|" (t-aa-z^-0|kl-o-m^-0| : round eyes). Unfortunately, Thai G2P always gives only one answer which is not always the right one. Secondly, words having more than one possible pronunciation could not be handled correctly. For example, "ประวัติศาสตร์" (history) can be pronounced "pr-a-z^-1| w-a-t^-1 t-i-z^-1|s-aa-t^-1|" or "pr-a-z^-1|w-a-t^-1|s-aa-t^-1|" but Thai G2P shows only one pronunciation. Lastly, Thai G2P could not correctly transcribe Thai Named Entities that are not typically found in the database, such as person names, acronyms, road names, points of interest, etc.

Given the above difficulties of Thai G2P, linguists or phoneticians are still indispensable in the development of the pronunciation dictionary. However, the previous version of Thai Pronunciation Dictionary, called LEXiTRON-Pro Version 1.0, required too many man-hours of linguists because there were several steps of manual tasks. Therefore, it is quite difficult and inconvenient to improve our LEXiTRON-Pro dictionary or create another one.

In this paper, we propose LEXiTRON-Pro Editor. It is an integrated tool to create a pronunciation dictionary more easily. This tool combines together Thai word segmentation, Thai G2P and Database with statistics, to enhance the accuracy of the phonetic transcription.

This paper is organized as follows. In the next section, we review basic concepts of the Thai language, the pronunciation dictionary and Thai word segmentation. In Section III, we summarize the previous version of LEXiTRON-Pro dictionary. Section IV explains how we reuse the old data. We propose our new application and explain the program interface in Sections V and VI. An experimental evaluation to compare our new program with the pre-

This work was supported by BEST Project under KET Platform of NECTEC.

vious system is done in Section VII and we conclude our paper in the last section.

II. Basic Concepts

A. Thai Language

The Thai language has 44 consonants and 24 vowels including 9 short vowels, 9 long vowels, and 6 diphthongs. Thai sound system can be derived in the format of /Ci-V-(Cf)-T/, where Ci denotes an initial consonant, V a vowel, Cf a final consonant which is optional, and T a tone [3].

Table I.
Thai consonant mapping to phonetic symbol

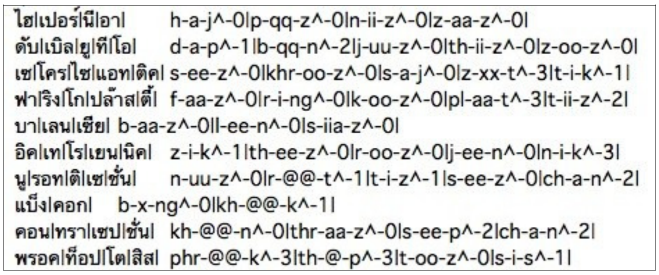
Consonant	Phoneme		Consonant	Phoneme	
	Initial (Ci)	Final (Cf)		Initial (Ci)	Final (Cf)
ก	k	k^	บ	b	p^
ข,ค,ฆ	kh	k^	ป	p	p^
ง	ng	ng^	ผ,พ,ภ	ph	p^
จ	c	t^	ร	r	n^
ฉ,ฌ,ช	ch	t^	ล,ฬ	l	n^
ซ,ศ,ษ,ส	s	t^	ว	w	w^
ญ,ย	j	j^	ห,ฮ	h	-
ฎ,ด	d	t^	ฝ,ฟ	f	p^
ฏ,ต	t	t^	ม	m	m^
ฐ,ฑ,ฒ,ถ,ท,ธ	th	t^	อ	z	-
ณ,น	n	n^			

Table II.
Thai vowel and mapped phoneme

Tongue Height	Tongue Advancement		
	Front (short, long)	Central (short, long)	Back (short, long)
Close	i, ii (อิ, อี)	v, vv (อึ, อือ)	u, uu (อุ, อู)
Mid	e, ee (เอะ, เอ)	q, qq (เออะ, เออ)	o, oo (โอะ, โอ)
Open	x, xx (แอะ, แอ)	a, aa (อะ, อา)	@, @@ (เอาะ, ออ)
Diphthongs	ia, iia (เอียะ, เอีย)	va, vva (เอือะ, เอือ)	ua, uua (อัวะ, อัว)

B. Pronunciation Dictionary

The pronunciation dictionary is a collection of words associated with their pronunciations in the form of phoneme sequences. Phonemes could be derived from standard sound representatives such as International Phonetic Alphabets (IPA) or Speech Assessment Methods Phonetic Alphabets (SAMPA). A letter-to-sound conversion (LTS) module takes the pronunciation dictionary as a primary source of knowledge to convert any textual word to its corresponding phoneme sequence. The LTS module plays an important role in building the phonetic transcription of speech corpus given speech orthographies. The output from the LTS tool is shown in Fig. 1. The first column is word list with syllable segmentation by "|" symbol and the second column is word's pronunciation. The pronunciation represents the word in the form of phoneme sequence with syllable segmentation and syllabic tone marked.



```
ไฮ|เปอร์|นี|อา|       h-a-j^-0|p-qq-z^-0|n-ii-z^-0|z-aa-z^-0|
ดับ|เบิล|ยู|ที|โอ|      d-a-p^-1|b-qq-n^-2|j-uu-z^-0|th-ii-z^-0|z-oo-z^-0|
เซ|โคร|ไซ|แอท|ติค| s-ee-z^-0|khr-oo-z^-0|s-a-j^-0|z-xx-t^-3|t-i-k^-1|
ฟา|ริง|โก|ปล้าส|ติ้|  f-aa-z^-0|r-i-ng^-0|k-oo-z^-0|pl-aa-t^-3|t-ii-z^-2|
บา|เลน|เซีย| b-aa-z^-0|l-ee-n^-0|s-iia-z^-0|
อิค|เท|โร|เยน|นิค|   z-i-k^-1|th-ee-z^-0|r-oo-z^-0|j-ee-n^-0|n-i-k^-3|
นู|รอท|ติ|เซ|ชั่น|      n-uu-z^-0|r-@@-t^-1|t-i-z^-1|s-ee-z^-0|ch-a-n^-2|
แบ็ง|คอก|    b-x-ng^-0|kh-@@-k^-1|
คอน|ทรา|เซป|ชั่น|  kh-@@-n^-0|thr-aa-z^-0|s-ee-p^-2|ch-a-n^-2|
พรอค|ท็อป|โต|สิส|  phr-@@-k^-3|th-@-p^-3|t-oo-z^-0|s-i-s^-1|
```

Fig 1. LTS output in LEXiTRON-Pro format

C. Thai Word Segmentation

Since Thai has no word boundary, word segmentation is the first thing to do. At present, there are two main approaches on Thai word segmentation. The first approach is machine learning based (MLB) approach which is a technique that learns from a tagged corpus in which word boundaries are explicitly marked with special annotations. This algorithm creates statistical models based on the features of characters surrounding the boundaries (e.g., n-gram of a candidate word boundary). The other approach is dictionary-based (DCB) approach which is based on string parsing technique – i.e., input characters are scanned and matched with word set from dictionary [4].

LongLEXTO, the Thai word segmentation tool used in our application, was developed by Human Language Technology Laboratory (HLT). This tool is dictionary-based approach using the longest matching (LM) technique. The longest matching is a selection algorithm for solving the ambiguity problem by scanning the input text from left to right and then selecting the longest match with a word in dictionary [5]. The main dictionary is extracted from LEXiTRON, English-Thai online dictionary, containing 35,936 words.

III. Previous Version of LEXiTRON-Pro dictionary

In the previous version, most work was done by hand, which can be summarized as follows:

- Extract a word list from variety of text articles
- Compare the word list with the original pronunciation dictionary (if exist) and extract only new words
- Generate pronunciations of new words with letter-to-sound conversion (LTS)
- Manually check and correct the results of LTS
- Add new word entries to the pronunciation dictionary

105,129 words were extracted by linguists. There were quite a variety of sources ranging from general words, road names, person names, and abbreviations. All words were then converted to phoneme sequences using LTS conversion. All entries were verified by hand before importing to the LEXiTRON-Pro dictionary. This dictionary will be used as a primary resource in our LEXiTRON-Pro Editor.

IV. Data Preparation

Before using the new program, we had to prepare some more data. We analyzed all syllables with consistent pronunciation in LEXiTRON-Pro dictionary and found 364,707 syllable entries. After that, we counted the frequency of the same syllables and found 24,736 unique entries. Then, all unique entries were imported to the LEXiTRON-Pro database.

The probability of each syllable's pronunciation is calculated by the frequency of each syllable's pronunciation compared with the total frequency of all possible pronunciations for that syllable. We use the following equation:

$$(\%) = \frac{f}{\sum_{i=1}^{n} (fi)} \times 100$$

When % is the probability of syllable's pronunciation
f is the frequency of syllable's pronunciation
n is the number of possible pronunciations for the same syllable

For example, the term “เศรษฐกิจ” (/set-tha-kit/: economy) contains syllables “เศรษฐ” and “กิจ”. The syllable “เศรษฐ” has 4 possible pronunciations: “s-ee-t^-1|”, “s-ee-t^-1 th-a-z^-0|”, “s-ee-t^-1 th-a-z^-1|”, and “s-ee-t^-1 th-a-z^-3|” with the frequencies of 16, 4, 54, and 2, respectively. Their calculated probabilities are then 21.05%, 5.26%, 71.05%, and 2.63%, respectively. All values are recorded in LEXiTRON-Pro database to help users to decide which pronunciation they want to use.

V. LEXiTRON-Pro Editor

A. Input Format

LEXiTRON-Pro Editor accepts the input file in plain text with TIS-620 or UTF-8 encoding. This file can contain word lists or paragraphs and may contain foreign words, punctuations, or numbers, all of which will be ignored.

B. Pronunciation Generation

Normally, the pronunciation dictionary is manually created, checked, and rechecked by phoneticians or skilled linguists. However, this method is inconvenient and time-consuming [6], [7]. Therefore, we have developed LEXiTRON-Pro Editor that generate pronunciation automatically, based on database system with statistics, incorporated with Thai word segmentation tool and Thai G2P. The procedure to generate pronunciation of LEXiTRON-Pro Editor is shown in Fig. 2.

In the first step, the input file, which may contain multiple paragraphs, is read by LEXiTRON-Pro Editor and then segmented into words using LongLEXTO. After that, these words will be further segmented into syllables by Thai G2P. The results from G2P algorithm are syllable-segmented words with symbol “|” and their pronunciations that will be used in the modification process. Next, each word from the previous process will be compared to LEXiTRON-Pro database. There are, at this step, three possible scenarios:

- If that word matches a word in the database, then retrieve its pronunciation from the database and display the result.
- If not, then use its syllables from Thai G2P instead to compare again to LEXiTRON-Pro database, syllable by syllable. if all the syllables are found in the database but there may be more than one possible pronunciation, our system will choose the most probable one and then combine all syllables' pronunciations together before displaying the result which is supposed to be the pronunciation of that word.
- If not all of the syllables are found, then display the word's pronunciation from Thai G2P instead.

For the first case, the results will not be checked again because they come from the validated database. For the second and third cases, the results must be checked and corrected by linguists.

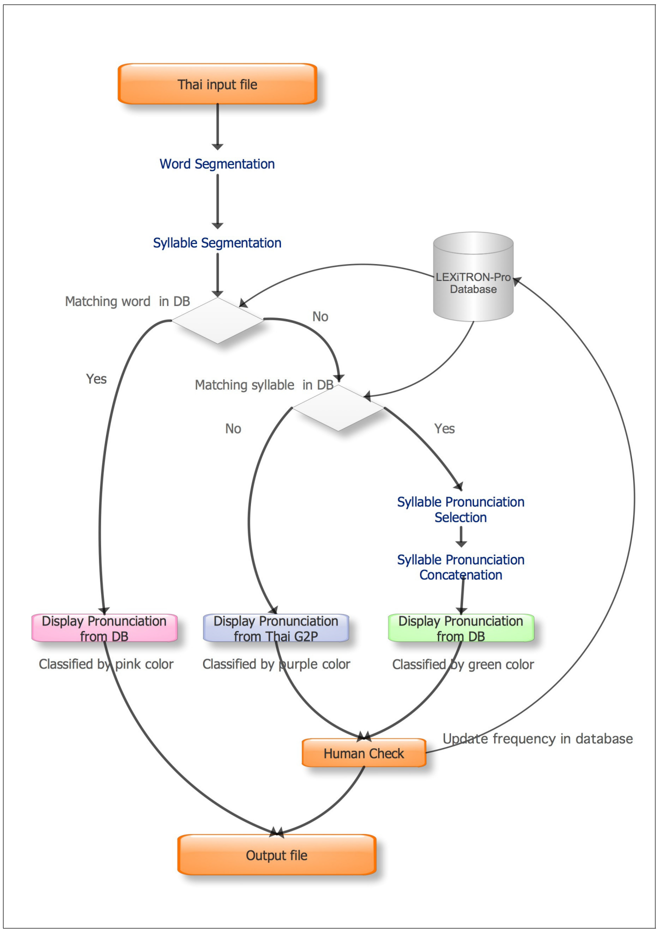

Thai input file
Word Segmentation
Syllable Segmentation
LEXiTRON-Pro Database
Matching word in DB
No
Yes
Matching syllable in DB
No
Yes
Syllable Pronunciation Selection
Syllable Pronunciation Concatenation
Display Pronunciation from DB
Classified by pink color
Display Pronunciation from Thai G2P
Classified by purple color
Display Pronunciation from DB
Classified by green color
Human Check
Update frequency in database
Output file

Fig 2. Work Flow in LEXiTRON-Pro Editor

C. *Pronunciation Verification*

Although the automatic generation of pronunciation is more convenient and can help linguists to reduce time, the results may contain some errors. Therefore, it is necessary to modify some results. We propose an easy interface to modify the pronunciation which will be explained in the section VI.B.

After verification, all data will be sent back to update the LEXiTRON-Pro database.

D. *Output Format*

The output file from LEXiTRON-Pro Editor has two options: word list format or paragraph format. The first format contains alphabetically ordered word list, divided into three columns: original words, words' syllables, and words' pronunciations. The other format contains alternate lines between word segmented text and words' pronunciation. The order of words' appearance is the same as the input file.

VI. Program Interface

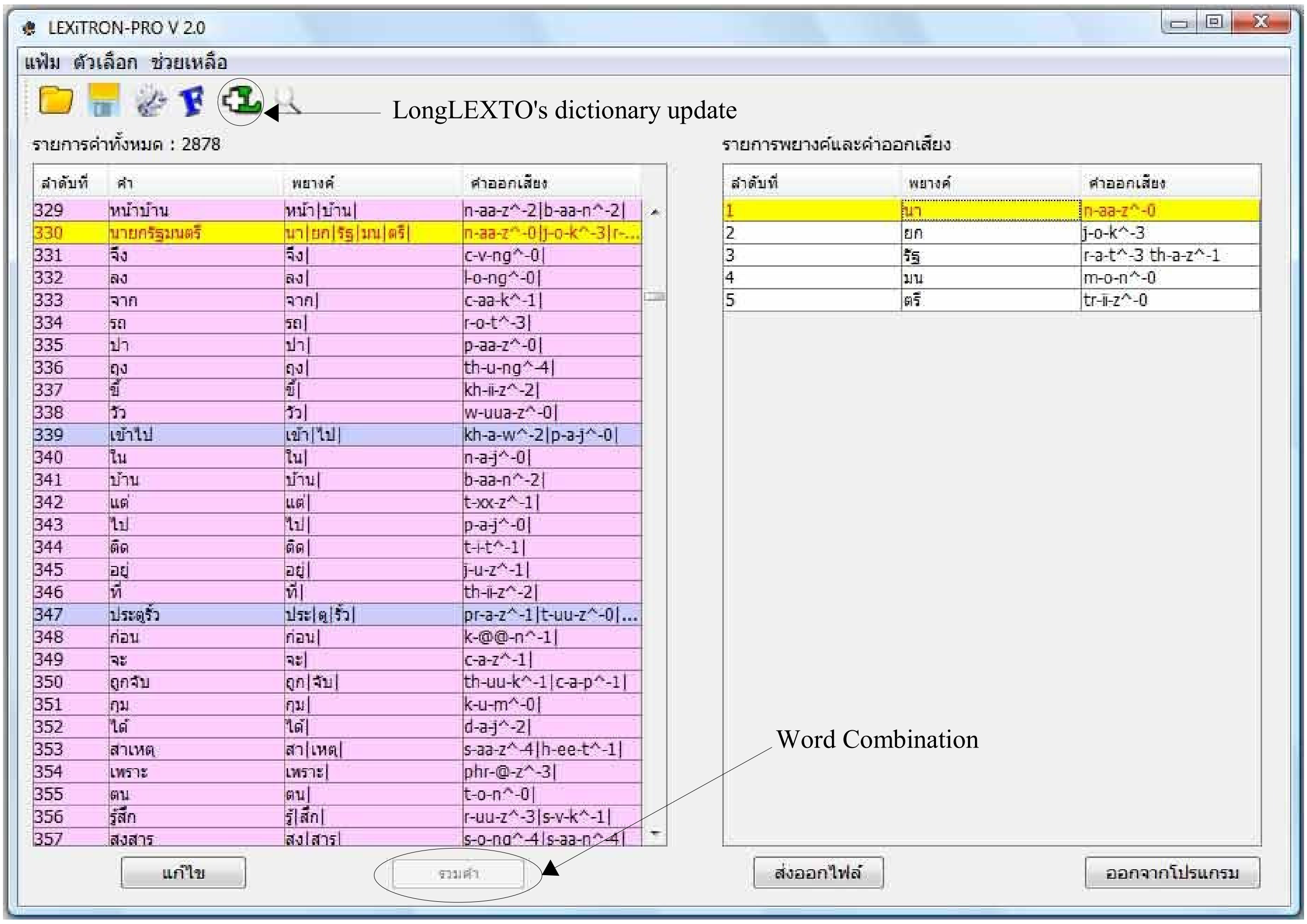
LEXiTRON-Pro Editor has two major user-interfaces:

A. *Main Interface*

LEXiTRON-Pro Editor automatically generates pronunciations from files and displays results in three columns as shown in Fig. 3. Words in the left table are sorted according to the order of appearance in the input file. Each line contains ordinal number, original word, word's syllables, and word's pronunciation. When a line is selected, every syllable and pronunciation of that word will be displayed in the right table, one syllable per line. It is possible to edit each pronunciation by double-clicking on it to open another window, which will be explained in the next subsection.

The main interface includes two other useful features:

- Word Combination: since LongLEXTO may commit some errors by segmenting one word into two or more, it is necessary to combine them to make a single word by selecting them and clicking on this button.
- LongLEXTO's dictionary update: after combining words in the previous step, we can update the LongLEXTO's dictionary with this button. In addition, we can add a new word by ourselves to the LongLEXTO dictionary. The modification will take effect immediately.



รายการคำทั้งหมด : 2878

ลำดับที่	คำ	พยางค์	คำออกเสียง
329	หน้าบ้าน	หน้า|บ้าน|	n-aa-z^-2|b-aa-n^-2|
330	นายกรัฐมนตรี	นา|ยก|รัฐ|มน|ตรี|	n-aa-z^-0|j-o-k^-3|r-...
331	จึง	จึง|	c-v-ng^-0|
332	ลง	ลง|	l-o-ng^-0|
333	จาก	จาก|	c-aa-k^-1|
334	รถ	รถ|	r-o-t^-3|
335	ป่า	ป่า|	p-aa-z^-0|
336	ถุง	ถุง|	th-u-ng^-4|
337	ขี้	ขี้|	kh-ii-z^-2|
338	รั้ว	รั้ว|	w-uua-z^-0|
339	เข้าไป	เข้า|ไป|	kh-a-w^-2|p-a-j^-0|
340	ใน	ใน|	n-a-j^-0|
341	บ้าน	บ้าน|	b-aa-n^-2|
342	แต่	แต่|	t-xx-z^-1|
343	ไป	ไป|	p-a-j^-0|
344	ติด	ติด|	t-i-t^-1|
345	อยู่	อยู่|	j-u-z^-1|
346	ที่	ที่|	th-ii-z^-2|
347	ประตูรั้ว	ประ|ตู|รั้ว|	pr-a-z^-1|t-uu-z^-0|...
348	ก่อน	ก่อน|	k-@@-n^-1|
349	จะ	จะ|	c-a-z^-1|
350	ถูกจับ	ถูก|จับ|	th-uu-k^-1|c-a-p^-1|
351	กุม	กุม|	k-u-m^-0|
352	ได้	ได้|	d-a-j^-2|
353	สาเหตุ	สา|เหตุ|	s-aa-z^-4|h-ee-t^-1|
354	เพราะ	เพราะ|	phr-@-z^-3|
355	ตน	ตน|	t-o-n^-0|
356	รู้สึก	รู้|สึก|	r-uu-z^-3|s-v-k^-1|
357	สงสาร	สง|สาร|	s-o-ng^-4|s-aa-n^-4|

รายการพยางค์และคำออกเสียง

ลำดับที่	พยางค์	คำออกเสียง
1	นา	n-aa-z^-0
2	ยก	j-o-k^-3
3	รัฐ	r-a-t^-3 th-a-z^-1
4	มน	m-o-n^-0
5	ตรี	tr-ii-z^-0

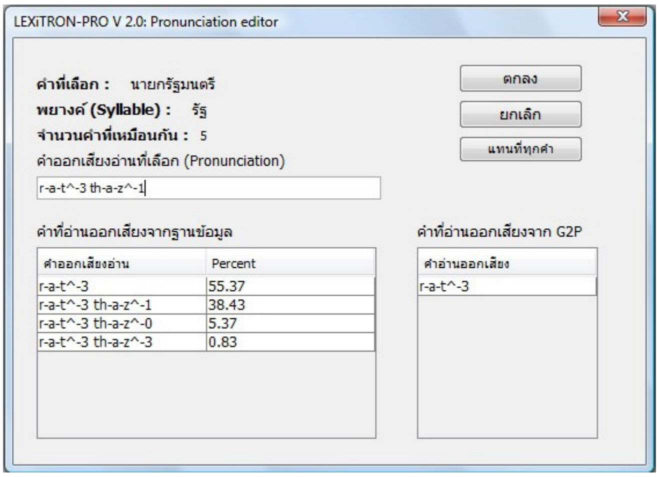
Fig 3. Main Interface

B. Pronunciation Editor

Since the automatic generation of pronunciation may be erroneous, especially when dealing with compound words, homographs, allophones and foreign words transliterated into Thai, we developed Pronunciation Editor for users to modify the syllable's pronunciation of the words with ease.

As shown in Fig. 4, users can edit any pronunciation by three methods:

- First, users can enter a new pronunciation by themselves in the top-left box.
- Second, users can choose the pronunciation proposed by Thai G2P in the bottom-right box.
- Last, users can choose one of the pronunciations from LEXiTRON-Pro database, as seen in the bottom-left box. If there are many possible pronunciations, they will be sorted by their calculated probability, as described in the section IV.



LEXiTRON-PRO V 2.0: Pronunciation editor

คำที่เลือก : นายกรัฐมนตรี

พยางค์ (Syllable) : รัฐ

จำนวนคำที่เหมือนกัน : 5

คำออกเสียงอ่านที่เลือก (Pronunciation)

r-a-t^-3 th-a-z^-1

คำที่อ่านออกเสียงจากฐานข้อมูล

คำออกเสียงอ่าน	Percent
r-a-t^-3	55.37
r-a-t^-3 th-a-z^-1	38.43
r-a-t^-3 th-a-z^-0	5.37
r-a-t^-3 th-a-z^-3	0.83

คำที่อ่านออกเสียงจาก G2P

คำอ่านออกเสียง
r-a-t^-3

Fig 4. Pronunciation Editor

VII. Experimental evaluation

To evaluate the performance of LEXiTRON-Pro Editor compared with the previous system, we performed an experiment on both systems to generate a pronunciation dictionary with a test set of 1,072 Thai named entities, including names of universities/colleges, mosques, temples, hospitals, police stations, train stations, government offices, monuments, etc. The performance was evaluated in three aspects: generation time, pronunciation accuracy, and number of ungenerable words. The results are presented in the Table III.

Table III.
Systems' evaluation

Aspect	Previous system	LEXiTRON-Pro Editor
Generation time	5 min.	45 sec.
Pronunciation accuracy	18.1%	73.6%
Ungenerable word	60 words	0 word

The results show that LEXiTRON-Pro Editor can reduce time consumption from 5 minutes to approximately 45 seconds in pronunciation generating process, while the accuracy is largely improved from 18.1% to 73.6%. Lastly, the number of ungenerable words, found 60 words in the previous system, becomes zero.

VIII. Conclusion

The goal of this application is to assist linguists to reduce time and errors from manual work by simplifying several steps of development process, changing some tasks to automatic methods and proposing an easy interface to users.

Compared to the previous version, our new program can reduce the process of dictionary development from 5 to only 1 step and can reduce the number of tools used by linguists from 3 to only 1 program. In addition, LEXiTRON-Pro Editor can automatically propose a word's pronunciation to users by 1 of the 3 options in the successive order: pronunciation from LEXiTRON-Pro database, pronunciation combined from syllables with highest probability, and pronunciation from Thai G2P. However, users can switch to another option or even input directly the pronunciation they want by themselves with our easy interface editor.

Our experiment shows that, with LEXiTRON-Pro Editor, the time consumption and the number of ungenerable word are significantly reduced while the pronunciation accuracy is considerably improved as well.

We plan to use this program to develop LEXiTRON-Pro Dictionary Version 2.0 by increasing its size to more than 130,000 words in the near future.

Since Thai pronunciation dictionary is a crucial component in other speech processing applications such as Thai TTS and ASR, the more completed dictionary means an opportunity to the more successful speech applications too.

References

[1] P. Tarsaku, V. Sornlertlamvanich, and R. Thongprasirt, "Thai Grapheme-to-Phoneme using probabilistic GLR parser," in *Proceeding of EUROSPEECH 2001*, Aalborg, Denmark, 2001, pp. 1057–1060.

[2] P. Charoenpornsawat and T. Schultz, "Example-based Grapheme-to-Phoneme conversion for Thai," in *Proceeding of INTERSPEECH 2006–ICSLP*, Pittsburgh, PA, USA, 2006, pp. 1268–1271.

[3] P. Chootrakool, C. Wuttiwiwatchai, and K. Kosawat, "A large pronunciation dictionary for Thai speech processing," presented at the ASIALEX 2009, Bangkok, Thailand, August 20–22, 2009, Paper P013.

[4] C. Haruechaiyasak, S. Kongyoung, and M. N. Dailey, "A comparative study on Thai word segmentation approaches," in *Proceeding of ECTI-CON 2008*, Krabi, Thailand, 2008, pp. 125–128.

[5] Y. Poovarawan and W. Imarrom, "Dictionary-based Thai syllable separation," in *Proceeding of the 9th Annual Meeting on Electrical Engineering of the Thai Universities*, Khonkaen, Thailand, 1986.

[6] L. Lamel and G. Adda, "On designing pronunciation lexicons for large vocabulary, continuous speech recognition," in *Proceeding of ICSLP 96*, Philadelphia, PA, USA, 1996, pp. 6–9.

[7] P. Pollák and V. Hanžl, "Tool for Czech pronunciation generation combining fixed rules with pronunciation lexicon and lexicon management tool," in *Proceeding of LREC 2002*, Las Palmas de Gran Canaria, Spain, 2002, pp. 1264–1269.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 435–438

ISBN 978-83-60810-27-9
ISSN 1896-7094

Automatic Detection of Prominent Words in Russian Speech

Daniil Kocharov
Saint-Petersburg State University
Universitetskaya emb., 11, 199034,
Saint-Petersburg, Russia
Email: kocharov@phonetics.pu.ru

***Abstract*—An experimental research with a goal to automatically detect prominent words in Russian speech is presented in this paper. The proposed automatic prominent word detection system could be further used as a module of an automatic speech recognition system or as a tool to highlight prominent words within a speech corpus for unit selection text-to-speech synthesis. The detection procedure is based on the use of prosodic features such as speech signal intensity, fundamental frequency and speech segment duration. A large corpus of Russian speech of over 200 000 running words was used to evaluate the proposed prosodic features and statistical method of speech data processing. The proposed system is speaker-independent and achieves an efficiency of 84.2 %.**

I. Introduction

THE SOLUTION of the prominent word detection task is to be used within the field of speech technologies while developing automatic speech recognition, unit-selection text-to-speech synthesis, spoken term detection, video and audio data indexing. For example, natural speech understanding systems need to know not only "what" has been said, but also "how" it has been pronounced. Intonation prominence is very important linguistic information for speech understanding, i.e. [1] showed that the use of word prominence degree helped to disambiguate the meaning of utterances.

The procedures of speech signal processing, prosodic features extraction and statistical speech data processing were developed during a series of investigations. The experiments used the data of CORPRES speech corpus created at the Department of Phonetics of Saint Petersburg State University [6]. The paper contains the experimental results and efficiency evaluation of the developed automatic prominent word detection system.

II. Prosodic Features of Word Prominence

A speaker prosodically emphasizes a word in an utterance to make it stand out of the surrounding words. The most common cure for doing that is a pitch accent that is acoustically expressed by the increase of local pitch maxima and minima. Intonational prominence reflects different aspects of pragmatics. It can express attitudes such as doubt, uncertainty and surprise, demonstrate anaphora references, the location of rheme and theme in the utterance, as well as show whether the utterance is a question or statement. Recently many research efforts have been dedicated to prominence detection due to its importance in such fields as natural speech understanding and emotion recognition in spontaneous speech. Almost all researchers assume that relative syllable and sound length, melody and loudness are highly connected with prominence [2]. The first research goal was to find the most efficient acoustic features for automatic prominent word detection.

A. Relative syllables and sounds length

The relative length of syllables and sounds is an obvious prominence feature. The speaker usually stretches a word if s/he wants to emphasize it. As there is no reliable algorithm of syllable detection for Russian and there is no syllable transcription in CORPRES, the speech corpus of Russian speech used in the present experiment, relative syllable length was not used. Two temporal features were used. The first feature is total word length in milliseconds. The second one is relative sound length within a current word that is expressed by the ratio of sound length within a current word and the length mean for the sound. The means were calculated for sound samples within the speech corpus. These features are possible to calculate as there is phonetic transcription with precise speech sound boundaries and orthographic transcription with precise word boundaries in CORPRES. Thus, there is no need to detect speech segment boundaries automatically, but it should be done by means of the automatic speech recognition in real-life applications.

B. Melodic features

The majority of researchers support the idea that melodic features are the most crucial for speech prominence, i.e. see [8]. In present research, the melodic contour of every word was examined separately. An original recently developed method of melody processing [6] was used. This method showed its efficiency in the system of automatic interpretation of tone unit prosody.

The melodic features were extracted from a preprocessed and smoothed melodic contour. The melodic contour is achieved as a result of automatic pitch detection system that has been developed earlier at the Department of Phonetics of Saint-Petersburg State University. The goal of preprocessing

This work was supported in part by the "Research Potential Development of Higher Education Institutions" Federal Program ("Automatic Interpretation of Prosody" Project)

is to eliminate microprosodic events and to get a smoothed melodic contour. This allows to get rid of calculation errors occurring within microprosodic events. Automatic melody preprocessing consists of the following four steps:

1) detection of voiced parts of the speech signal;
2) pitch detection within voiced parts;
3) microprosody and laryngalization rule-based processing;
4) melodic contour smoothing based on the algorithm of moving average.

The following melodic features were selected for prominence detection based on the analysis of other research and solutions as well as a series of experiments: maximum, minimum, mean and standard deviation of the fundamental frequency within a word. The rate of fundamental frequency change is also taken into account to model not only the fundamental frequency itself but the extent of its change as well. Thus, maximum, minimum, mean and standard deviation of the fundamental frequency change within a word were applied for this purpose. These features are applied based on the idea that the change of fundamental frequency is higher within a prominent word.

C. Intensity of speech signal and its spectrum

Speech loudness also correlates with prominence. Speech loudness corresponds with speech signal intensity or, more precisely, with its spectrum intensity. Meanwhile there are two main ways of modeling speech loudness. The first one is a calculation of spectrum intensity within certain, most significant frequency bands. The second is a calculation of speech signal intensity. The latter does not require FFT calculation that leads to much faster feature extraction. It is worth saying that the efficiency of signal based features is equal to the efficiency of spectrum based ones. The following features were used to express word loudness: maximum, minimum, mean and standard deviation of speech signal intensity within a current word.

The discrete extraction of signal intensity features is used. The speech signal within a word is divided into processing windows. Window length is 10 ms and window step is also 10 ms, that is windows do not overlap. The signal intensity within a processing window is a mean of signal amplitude values within the window. Thus, we calculate a single signal intensity value every 10 ms. This value array is used to obtain the features listed above.

D. D. General overview of proposed acoustic features

The acoustic features described above express three different aspects of speech prosody: melodic, dynamic and temporal description of prosody. They are independent at the first glance, that is changing one of them does not influence the others. But that is not really true. Changing one of them will influence the perception of the others. The perceptual characteristics of speech are more important than its real acoustic characteristics when one considers speech prominence. That is why all the features should be taken into account. For example, an increase in signal intensity leads to an increase in perceived pitch and an increase in fundamental frequency leads to an increase in perceived loudness [3]. It is quite obvious how to extract temporal and dynamic information. The task of key melodic feature extraction still requires a solution because it is not obvious what melodic features are the most important for automatic prosody interpretation. However, to make the feature extraction process consistent, it was decided to use the following list of features to model word prominence:

1) total word length, relative sounds length;
2) maximum, minimum, mean and standard deviation of the fundamental frequency within a word;
3) maximum, minimum, mean and standard deviation of the fundamental frequency change within a word;
4) maximum, minimum, mean and standard deviation of the speech signal intensity within a word.

The use of these acoustic features is based on the following reasons. First of all, it is well known that short words are rarely prominent, but the melodic contour within them usually changes greatly and rate of F0 change is a correlate of prominence. Thus, the use of word length as an acoustic feature helps to detect such words as non-prominent. The melodic and dynamic features are designed following the same principles: maximum, minimum, mean and standard deviation are calculated. This allows to estimate the range and variance of prosodic features. On the other hand, it allows to examine how words differ from each other, especially by statistical measures such as mean and standard deviation. These prosodic features are essential and almost all other features are based on them.

III. Statistical Processing of Prosodic Data

The choice of statistical data processing and acoustic modeling method that allows to achieve the best efficiency of automatic prominent word detection is no less crucial than the choice of acoustic features. The main statistical framework applied in speech technology at the moment is hidden Markov models (HMM) that would be perfect for the solution of this task within an automatic speech recognition system. HMM is the best choice when one needs to reveal a context dependency of objects or a dependency of certain objects appearing next to preceding objects. The solution of current task does not require this; it is possible to make an assumption about context independence of word prominence from the prominence of preceding words. Thus it was decided to use another method. It seems reasonable to use classification and regression trees (CART) to detect prominent words as it was done in [5]. CART is an effective classification method when classified objects are independent from each other. Besides that, CART allows to define a relative significance of features for a classification task. It is especially valuable from scientific point of view and allows us to develop, test and apply acoustic features that are more and more effective and reliable for a task of modeling and detecting prominent words. Usually the entropy is used as a splitting criterion in a CART framework. However, it has been decided to use probability of prominent words as a splitting criterion in the current system. There are two

TABLE I
THE EVALUTION RESULTS WITH DIFFERENT SETUPS

Experiment	Efficiency	Precision	Recall
SpInd	84,2	83.3	79.1
SpDep	77.1	81.2	73.4
Male Voices	89.7	90.4	80.1
Female Voices	87.3	88.2	78.8

reasons for that. The first one is the fact that when entropy is calculated all classes are supposed to be equally probable, but the number of prominent words is 4 times smaller than the number of non-prominent words. In case of using entropy this could lead to the situation when there are objects of different classes in all CART leafs: many non-prominent words and several prominent words. The other reason is that there are just two classes in this case: prominent and non-prominent words. Thus, uncertainty degree is unambiguously defined by a probability of one class. The experimental results showed that the probability of prominent words is a much more efficient splitting criterion than entropy.

IV. Experiments

A. Experimental Data Description

All the experiments were carried out with the CORPRES (Corpus of Russian Professionally Read Speech) corpus. It consists of recordings from 8 speakers, four men and four women. It contains 25 hours of fully annotated speech [7], three hours per each speaker. The corpus contains the following annotation levels:

1) pitch marks – boundaries of fundamental frequency periods;
2) phonetic events labeling – boundaries and labels of phonetic events;
3) phonetic transcription – boundaries and labels of speech sounds;
4) orthographic transcription – boundaries and labels of words;
5) prosodic transcription – boundaries and labels of tone units and pauses.

There are 211 383 running words in the fully annotated part of the speech corpus and 40 547 of them were labeled by experts as prominent.

B. Experimental Results

Cross-validation has been applied for efficiency measurement during experiments. This method is widely used in cases of lack of data and non-uniform data. Prominent words are non-uniformly distributed over the speech corpus and non-prominent words would be considered as a more probable class. The cross-validation allows to avoid that. It has been decided to use the same efficiency metrics as the ones used in search tasks for the task of prominent word detection can be considered a search task. These are error-rate, precision and recall. A series of experiments was held to estimate the efficiency of automatic detection of prominent words using the above mentioned prosodic features and statistical classifier.

A series of experiments was held:

1. Speaker Independent (SpInd): All data were uniformly devided into 10 parts, i.e. the recordings of every speaker were divided into 10 parts. Nine parts were used as training data and one part was used as test data, thus there were about 22.5 hours of training data and 2.5 hours of test data.

2. Speaker Dependent (SpDep): The purpose of the experiment was to evaluate the efficiency of the system when the recordings of one speaker are used as training data and the recordings of other speakers are used as test data. The data were divided in the following way: the data from 7 speakers (about 22 hours) were used for training and the data from 1 speaker (about 3 hours) were used for evaluation.

3. The last experiment was intended to evaluate the system with gender dependent data. First of all, data from male speakers were separated from data from female speakers. Each part of the data was divided into training data (9/10 of data, about 11.25 hours) and test data (1/10 of data, about 1.25 hours). Thus, four different experiments were held and the results are presented in Table I. The results in the table show several interesting tendencies. The results are much better for the gender dependent system then for gender independent. It might be caused by the significant differences in pitch between female and male voices. This proves the concept that pitch plays a major role in prominence detection.

Another conclusion is that SpInd yields better results then SpDep. This is probably due to the fact that training and test data within SpInd experiment included data from all speakers, while in SpInd experiment training data exluded the data of the test speaker. This shows that the data from seven speakers was not enough to train a speaker independent system and to predict the excluded speaker efficiently.

An overall efficiency of 84.2 % was achieved for speaker independent task. Table II shows the comparison of this result against the results achieved by other researchers in speaker independent systems.

The empty cells in the table mean that the authors did not present precision and recall results in their papers. As one can see, the efficiency of the current system is not the best one, but not the worst. However, it is worth highlighting that an amount of experimental data is by two orders of magnitude greater than the amount of data used to test other systems. Thus, the experimental results can be considered as positive and efficient enough to be a baseline for further research in the field of automatic detection of prominent words.

TABLE II
THE EFFICIENCY OF THE AUTOMATIC PROMINENT WORD DETECTION AS COMPARED TO SIMILAR RESULTS ACHIEVED BY OTHER RESEARCHERS

Research	Language	Amount of Data	Efficiency	Precision	Recall
Brenier et. al. [2]	English	2906 words	87,1		
Kroul [4]	Czech	2160 words	91.1		
Tamburini [9]	Italian	4780 syllables	82.5	75.6	77.9
Wang and Narayanan [10]	English	3247 words	76.2	82.1	73.4
Current System	Russian	211383 words	84.2	83.3	79.1

V. CONCLUSION

The paper presented results of the research dedicated to automatic detection of prominent words. Algorithms of prosodic features extraction were developed during this research. Three types of prosodic features were used: melodic, dynamic and temporal. CART with modified splitting criterion was used as a statistical classifier. The efficiency of the developed system was tested in a series of experiments. The efficiency of 84.2 % was achieved, that which is comparable to other research in this field. The undisputable advantage of this system is that it is the first such system of the kind that has been developed for the Russian language and it could undoubtedly be used within automated annotation of speech corpora modules and automatic speech recognition systems.

REFERENCES

[1] M. E. Beckman and J. J. Venditti "Tagging Prosody and Discourse Structure in Elicited Spontaneous Speech," *in Proceedings of Science and Technology Agency Priority Program Symposium on Spontaneous Speech,* Tokyo, Japan, 2000, pp. 87–98.

[2] J. M. Brenier, D. M. Cer, D. Jurafsky "The Detection of Emphatic Words Using Acoustic and Lexical Features," *in Proceedings of International Conference on Speech Communication and Technology 2005,* Lisbon, Portugal, 2005, pp. 3297–3300.

[3] H. Fletcher and W. A. Munson "Loudness, Its Definition, Measurement, and Calculation," *The Journal of the Acoustical Society of America,* vol. 5, 1933, pp. 82–108.

[4] M. Kroul "Automatic Detection of Emphasized Words for Performance Enhancement of a Czech ASR System," *in Proceedings of SPECOM 2009,* St. Petersburg, Russia, 2009, pp. 470–473.

[5] A. Rosenberg and J. Hirschberg "Detecting Pitch Accents at the Word, Syllable and Vowel Level," *in Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics Companion Volume: Short Papers,* Colorado, USA, 2009, pp. 81–84.

[6] P. Skrelin and D. Kocharov "Avtomaticheskaja obrabotka prosodicheskogo oformlenija viskazivanija: releventnie prosodicheskie priznaki dla avtomaticheskoj interpretatsii intonatsionnoj modeli," *in Trudi tretiego mezhdistciplinarnogo seminara Analiz russkoj rechi,* St. Petersburg, 2009, pp. 41–46.

[7] P. Skrelin, N. Volskaya, D. Kocharov, K. Evgrafova , O. Glotova, and V. Evdokimova "A Fully Annotated Corpus of Russian Speech," *in Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10),* Valletta, Malta, 2010, pp. 109–112.

[8] B. M. Streefkerk, L. C. W. Pols, L. F. M. Ten Bosch "Acoustical Features as Predictors for Prominence in Read Aloud Dutch Sentences Used in ANNâĂŹs," *in Proceedings of European Conference on Speech Communication and Technology 1999,* Budapest, Hungary, 1999, pp. 551–554.

[9] F. Tamburini "Automatic Prominence Identification and Prosodic Typology," *in Proceedings of Interspeech Conference on Speech Communication and Technology 2005,* Lisbon, Portugal, 2005, pp. 1813–1816.

[10] D. Wang and Sh. Narayanan "An Acoustic Measure for Word Prominence in Spontaneous Speech," *in IEEE Trans. Audio, Speech, and Language Processing,* vol. 15, no. 2, 2007, pp. 690–701.

Proceedings of the International Multiconference on Computer Science and Information Technology pp. 439–446

ISBN 978-83-60810-27-9
ISSN 1896-7094

Computing trees of named word usages from a crowdsourced lexical network

Mathieu Lafourcade, Alain Joubert
LIRMM – Université Montpellier 2 – CNRS
Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier
161, rue Ada – 34392 Montpellier Cédex 5 – France
Email: {lafourcade, joubert}@lirmm.fr

***Abstract*—Thanks to the participation of a large number of persons via web-based games, a large-sized evolutionary lexical network is available for French. With this resource, we approached the question of the determination of the word usages of a term, and then we introduced the notion of similarity between these various word usages. So, we were able to build for a term its word usage tree: the root groups together all possible usages of this term and a search in the tree corresponds to a refinement of these word usages. The labelling of the various nodes of the word usage tree of a term is made during a width-first search: the root is labelled by the term itself and each node of the tree is labelled by a term stemming from the clique or quasi-clique this node represents. We show on a precise example that it is possible that some nodes of the tree, often leaves, cannot be labelled without ambiguity. This paper ends with an evaluation about word usages detected in our lexical network.**

I. Introduction

In this paper, we describe an approach for acquiring lexical structures useful for word sense disambiguation (WSD). Having a word sense inventory is often mandatory prior to many text analysis application but those resources are quite rare especially for French. By text analysis, we mean here a task where word senses and relations amongst word senses are identified. Furthermore, being constructed either by experts (generally lexicographers) or from corpora, they barely reflect what people would think of. An adequate resource would then mostly refer to word usages than word senses (in the classical meaning of dictionaries) and would also try to evaluate the proper weight of an usage[1]. A very immediate usage of a term would have a higher weight than a rare one. We can then expect a quite large discrepancy between a proper inventory of usages and inventory of meanings (in dictionaries). Some usages would not be in dictionary either being elisions (like in French *sapin* for *sapin de Noël* – eng. *fir* for *Christmas tree*) or popular (like in French *caisse* as *car*). Some meaning would be absent of a *popular meaning inventory* when too technical, too rare or simply unknown by the vast majority of people. Can we try to define precisely what is the difference between a word sense and a word usage? Word usages are broader and would certainly include senses, but not the other way around. In French, for example, *sapin* (*fir*) beside being the tree, has a strong usage related to Christmas (*fir* as *Christmas tree*) and tends to become autonomous. A putative definition of a word usage could be a *specific meaning in a given context, popular enough to be spontaneously given by someone.*

The method and evaluation presented in this paper are based on a resource under construction but already freely available: the JeuxDeMots lexical network[2]. We first briefly remind the reader of the principles of two games which aim at building a base of relations between terms. The first of these two games (JeuxDeMots[3]) allows the construction of a lexical network, while the second game (PtiClic[2]) allows the user to strengthen associations acquired thanks to JeuxDeMots. With the network thus obtained, we tackle the problem of the word usage determination, by analysing the relations between every term and its immediate neighbours. The similarity between the various usages of the same term can be computed allowing us to build the classification tree of the usages of a term, the nodes of which being labelled.

[1] Definition for word usage is given several lines below, in this paragraph, and the weight of an usage is explained in section III.

[2] This resource is available at http://www.lirmm.fr/jeuxdemots/rezo.php for the lexical network and at http://www.lirmm.fr/jeuxdemots/diko.php for the obtained dictionary.

[3] JeuxDeMots and PtiClic are available at http://jeuxdemots.org and http://pticlic.org. An English version has recently been added, as well as a Thai, Japanese and Spanish versions (http://www.lirmm.fr/jeuxdemots/world-of-jeuxdemots.php)

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Such a word usage tree structure as of primary interest for WSD should be evaluated against users before considering using it in applications. Furthermore, one of the objectives is to be able to connect the relations, not on the very terms (with ambiguities for polysemous terms), but on their usages (thus by clearing up lexical ambiguities).

II. Lexical network construction

The basic principles of JeuxDeMots (JDM) software, the game design, as well as the incremental construction of the lexical network, have already been described in [1]. A game takes place between two players, in an asynchronous way, based on the concordance of their propositions. When a first player begins a game, an instruction concerning a type of competence (synonyms, opposite, domains …) is displayed, as well as a term[4] T randomly picked in a base of terms. This player has then a limited time to answer by giving propositions which, to his mind, correspond to the instruction applied to the term T. The same term, along the same instruction, is afterwards proposed to another player; the process is then identical. For the same target term T and a same instruction (synonyms, domains, free associations…), we record the answers common to both players. Validations are thus made by concordance of the propositions between pairs of players. This validating process is similar to the one used by [2] to index images and, as far as we know, this has never been done in the field of the lexical networks. In Natural Language Processing, some other Web-based systems exist, such as *Open Mind Word Expert* [3] that aims to create large sense tagged corpora with the help of Web users, or *SemKey* [4] that exploits WordNet and Wikipedia in order to disambiguate lexical forms to refer to a concept, thus identifying a semantic keyword.

The structure of the lexical network we are building relies on the nodes and relations between nodes, as it was initially introduced by [5]. More precisely, JDM game leads to the construction of a lexical network connecting terms by typed and weighted relations[5]. These relations are labelled by the instruction given to the players and they are weighted according to the number of pairs of players who proposed them. The morphosyntactic category was not initially indicated in JDM. A recent evolution, published in [6], has allowed us to introduce the notion of refinement of a term. This refinement can depend on the meaning (case of polysemous terms) or on the morphosyntactic category of the term.

In a similar way to JDM, a PtiClic game takes place in an asynchronous way between two players. A target term T, origin of relations, as well as a cluster of words resulting from terms connected with T in the lexical network produced by JDM are proposed to a first player. Several instructions corresponding to types of relations (synonym, hypernym, hyponym, predicate relations like possible/typical agent, patient or instrument, part-of and substance, ...) are also displayed. The player associates words of the cluster with instructions he thinks correspond by a drag and drop. The same term T, as well as the same cluster of words and the same instructions, are also proposed to a second player. According to a principle similar to that set up for JDM, only the propositions common to both players are taken into account, thus strengthening the relations of the lexical network. Contrary to JDM, the players of PtiClic cannot suggest new terms, but are forced to choose among those proposed. This choice of conception has to allow to reduce the noise due to misspelt terms or to the confusions of meanings.

The collaborative building of resources by non-experts may induce some errors. In fact, as one may expect, we detected some of them, such as classical orthographic mistakes (eg: *théatre* for *théâtre*) or traditional confusions (eg: French singer *Dalida* with the biblical character *Dalila*) … These well-known mistakes are relatively rare and they can be manually detected.

According to the JeuxDeMots Web site, at the time of the writing of this paper, the lexical network contains around one million relations linking 221000 terms. Around 800000 games have been played corresponding to more than 13000 hours of cumulated play.

III. Similarity between usages of a term

A. Word Usage Determination

If a term T is polysemous, the terms which are directly connected with it (semantically connected in the lexical network) form several different groups, each of these groups constituting a word usage of T. The notion of word usage (often referred to as "usage") is much more accurate and relevant than the notion of meaning which, as shown by [7], is relatively poor when we refer to traditional dictionaries or to resources as WordNet. Our hypothesis is that the usages of a term correspond in the network to the various cliques this term belongs to. A clique is a set of terms constituting a fully connected subgraph in the lexical network. Two terms T_i and T_j belong to the same clique if there is at least one relation between T_i and T_j and at least one relation between T_j and T_i. Our approach is similar to the one developed by [8] from dictionaries of synonyms.

Why using the JeuxDeMots lexical network for our experiment instead of WordNet [9], EuroWordNet or WOLF [10] ? These resources are handcrafted contrary to the JeuxDeMots lexical network which is crowdsourced through some games, and by itself it is interesting to assess if common word usage can be identified. Similar approaches like [11] have been conducted on contexonyms but on

[4] A term can be a compound word (for example: *Christmas tree*) and each of its words may be a term (*Christmas* and *tree*).

[5] A relation can be thus considered as a quadruplet: origin term, destination term, type and weight of the relation. Between two same terms, several relations of different types can exist.

resources trained on very large corpora (and again not extracted from people).

For the clique identification, we take into account all relation types available in the lexical network. Of course we might certainly consider that they do not contribute equally to the induced word usage, but the principle of JeuxDeMots induce that the most important relations have the highest weights and that the most important relation types (for a given term) are the most populated. So, there is, *a priori*, no need to stress on specific relation type, as this information is already implicitly present in the network. Estimating the relevance of a usage consists in obtaining a measure of its importance both in terms of frequency and of lexical coverage. Considering the principle of the weighting of the relations, the weight of a usage is correlated with the weights of the relations between the terms of the clique which characterizes this usage. We have the following notations :

- C is a given clique for the term T (this is a set of terms and instances of relation).
- C_{all} is the union of all C (that can be seen as the full pseudo clique for T).
- W(C) is the sum of the weights of the relations between the terms of C.
- Card(C) is the number of terms in C.

So, for a clique C related to the term T, we define formally the *relevance* as :

$$Rel(C) = W(C) * \log(Card(C_{all})/Card(C))$$

The Rel measure for a clique may be seen as an adaptation of the tf/idf measure where, for the sake of simplicity, we have not divided by $W(C_{all})$. If there is only one clique, the relevance is equal to 0 and of course in that case we consider that there is only one usage. Figure 1 presents the obtained usages for the term *sapin* (*fir*), and the relevance of each of these usages.

0: 'sapin' 'fiacre' REL = 52
fir, hansom
1: 'sapin' 'cercueil' REL = 55
fir, coffin
2: 'sapin' 'montagne' REL = 38
fir, montain
3: 'sapin' 'épicéa' 'ginkgo' 'conifère' 'cèdre' 'mélèze' 'résineux' REL = 66
fir, spruce, ginko, conifer, cedar, larch,conifer
4: 'sapin' 'vert' 'arbre' REL = 126
fir, green, tree
5: 'sapin' 'épicéa' 'épinette' 'conifère' REL = 59
fir, spruce, spruce, conifer
6: 'sapin' 'aiguille' REL = 43
fir, needle
7: 'sapin' 'conifère' 'arbre' REL = 139
fir, conifer, tree
8: 'sapin' 'guirlande' 'Noël' REL = 111
fir, garland, Christmas
9: 'sapin' 'boule' 'boules' REL = 51
fir, ball, balls
10: 'sapin' 'boule' 'Noël' REL = 108
fir, ball, Christmas
11: 'sapin' 'Noël' 'sapin de Noël' 'sapin de noël' REL = 84
fir, Christmas, Christmas Tree, Christmas tree
12: 'sapin' 'Noël' 'fête' REL = 152
fir, Christmas, celebration
13: 'sapin' 'arbre' 'bois' 'forêt' REL = 219
fir, tree, wood, forest
14: 'sapin' 'arbre' 'bois' REL = 148
fir, tree, wood
15: 'sapin' 'conifères' REL = 71
fir, conifers

Fig 1 : 16 usages for the term *sapin* (*fir*) as found in the lexical network at the writing time.

The most relevant clique is {**sapin, arbre, bois, forêt**} *(fir, tree, wood, forest)* with a score of 219, followed by {**sapin, Noël, fête**} *(fir, Christmas, celebration)* with a score of 152.

B. Clique Similarity

The similarity between two objects can be defined according to [12] as being a function of their common characteristics with regard to all their characteristics. In NLP, we find several definitions of the similarity, for example [13] or [14]. More recently, [15] evaluated several different measures of lexical semantic relatedness, while [16] presents a general survey on this question. In our case, it corresponds to the ratio between the weight of the relations connecting two cliques and the total weight of the relations on all the terms of these two cliques. We note W(E) the weight sum of the relations between the terms of the set E (as in 3.1). The similarity between two cliques C1 and C2 will be equal to the *Jaccard* indice :

$$Sim(C1,C2) = W(C1 \cap C2) / W(C1 \cup C2)$$

We should note here, that the Jaccard indice is in our case applied on the set of relations and that the 'cardinality' of this set is the sum of the weights. Usually the Jaccard indice is applied on the true cardinality of the sets, considering equally all elements of the set. Figure 2 shows the similarities between the cliques of the term *sapin* (*fir*).

IV. Classification Tree of Word Usages

A. Construction

Our aim is to obtain a representation of the various usages of a term T in the form of a tree, with the root grouping together all the meanings of T and the branches corresponding to its various usages. Generally, most of terms possess several not separate cliques. In that case, the further away we go from the root of the tree, the more we meet fine distinctions of usages. In fact, we build the tree of the usages of a term T according to a "bottom - up" method: from all of its cliques, that is, from its leaves and going back up to its root which groups together all the meanings of T. For that purpose, we apply an agglomerative hierarchical clustering algorithm: we merge the cliques, two by two, beginning with those whose coefficient of similarity is the highest: thus, we build quasi-cliques representing groups of usages, close during the first fusions, less and less close during the successive fusions. The merging algorithm ends when all coefficients of similarity are equal to zero.

The usage tree of a term is a structure expressing the refinements of its various meanings as deduced from the state of the lexical network. It thus constitutes a decision tree, a data structure which can be exploited for disambiguation. Furthermore, nodes of this tree are weighted allowing to identify usages that are the most common, which is both useful for guessing default cases and ordering usages from the most activated for people to the least activated.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	0.04	0	0.08	0.04	0	0	0	0	0	0.01	0.03	0
3	0	0	0	1	0.31	0.66	0.06	0.54	0.02	0	0.02	0.02	0.02	0.2	0.18	0.36
4	0	0	0.04	0.31	1	0.29	0	0.75	0.03	0	0.03	0.03	0.03	0.68	0.72	0.15
5	0	0	0	0.66	0.29	1	0.09	0.56	0.03	0	0.03	0.03	0.03	0.16	0.13	0.48
6	0	0	0.08	0.06	0	0.09	1	0.05	0	0	0	0	0	0	0	0.33
7	0	0	0.04	0.54	0.75	0.56	0.05	1	0.03	0	0.03	0.03	0.03	0.64	0.66	0.46
8	0	0	0	0.02	0.03	0.03	0	0.03	1	0.14	0.69	0.6	0.77	0.01	0	0
9	0	0	0	0	0	0	0	0	0.14	1	1	1	1	0	0	0
10	0	0	0	0.02	0.03	0.03	0	0.03	0.69	1	1	1	1	0.01	0	0
11	0	0	0	0.02	0.03	0.03	0	0.03	0.6	1	1	1	1	0.01	0	0
12	0	0	0	0.02	0.03	0.03	0	0.03	0.77	1	1	1	1	0.01	0	0
13	0	0	0.01	0.2	0.68	0.16	0	0.64	0.01	0	0.01	0.01	0.01	1	1	0.06
14	0	0	0.03	0.18	0.72	0.13	0	0.66	0	0	0	0	0	1	1	0.08
15	0	0	0	0.36	0.15	0.48	0.33	0.46	0	0	0	0	0	0.06	0.08	1

Fig 2 : Similarity matrix between the cliques of the term *sapin* (*fir*).
The clique numbers are those from figure 1.

B. Labelling

The labelling of the various nodes of the usage tree of a term is made during a width-first search, that is, according to a "top - down" method. The tree root is labelled by the term itself. Every node of the tree is labelled by a term stemming from the clique or the quasi-clique this node represents; the selected term is the one whose sum of the weights of its relation with the root term is the highest, after eliminating all the terms labelling the nodes of the tree situated in a depth lower than that of the concerned node. Thus, it is possible that a node cannot be labelled if all the terms which define it have already been used in the labelling of nodes previously done. In this case, this node, but also its brother and all its successors, are not labelled. Figure 3 shows the tree obtained for the term *sapin* (*fir*).

C. Tree pruning

The relevance of an inner node of the tree is computed as defined above considering the quasi clique obtained during the merging. We ignored the usages of the tree which relevance is below a given threshold (empirically set to 50 in our experiment). This threshold correspond to the configuration where both terms have been associated to each other to only two pairs of player (one pair for each direction) and maybe it could have been accidental. In figure 3, in the tree for the term *sapin*, we discard the *sapin (montagne)* node at level 1 which relevance is only equal to 38 (clique number 2 in Figure 1). The nodes for *sapin* as *fiacre*, *cercueil*, *arbre* and *Noël* have respectively a relevance of 52, 55 and more than 300 for the last two.

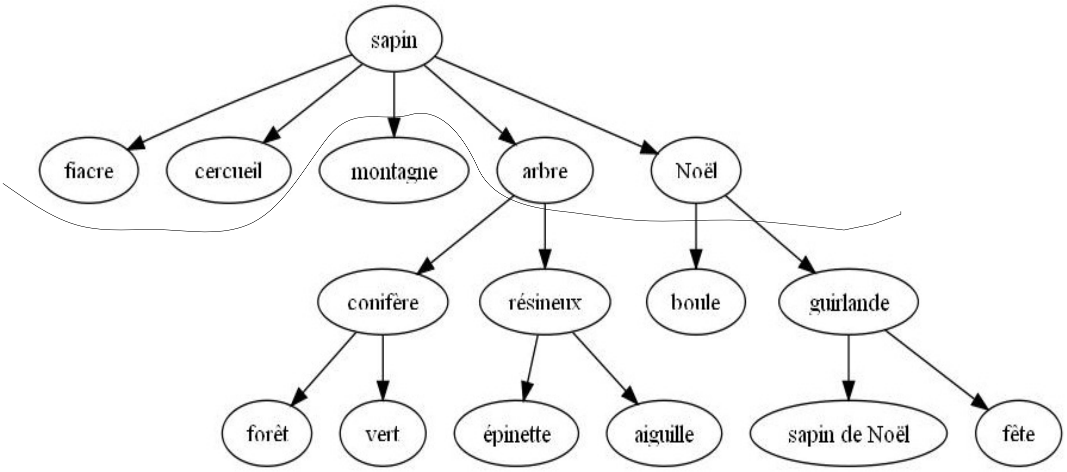


Fig 3 : Labelled word usage tree for the term *sapin* (*fir*). We pruned this tree by deleting nodes which cannot be labelled or which the relevance is too low (below 50 in our experiment)

V. Evaluation

A. Obtained Word Usages

Regularly, we examine the highly polysemous terms, and we ask an expert lexicographer to validate the meanings detected by JDM, after pruning the usage tree. The role of the expert does not extend beyond this validation: he doesn't add any meaning, even if he thinks some are missing, and he removes only those that obviously correspond to mistakes. As the time of writing this article, with the expert's help, we thus obtain 4412 validated word usages for 1263 terms, which correspond to a mean of around 3.5 usages per term. Out of these 1263 terms more than 80% are labeled as very common terms (at least one meaning should be known at the age of 12). The terms are mostly common nouns also some usages are tagged with other part-of-speech like verb or adjective. For example *dîner* in French can be at the same time a noun or a verb. The expert's role must go on to validate much more meanings detected in our lexical network.

B. User Evaluation

Evaluating the quality of such word usages is difficult, dramatically in the absence of adequate gold standard. As far as we aware of there is no such resources for French. If we had got any, a (semi-)automatic evaluation might have been feasible on a very much large scale. So, we decided to make a user based evaluation, trying to access qualitatively and quantitatively the word usages we obtained so far. We undertook the evaluation only at the first level of the usage tree computed for a given term.

We based our evaluation on naive users (i.e. not lexicographers) for two reasons. First, finding lexicographers for this task is not easy to say the least and certainly not more than few ones. Secondly, we wanted to have an evaluation confronting common people to our data. The idea is to identify word usages the same way (as a result at least) an average person would do. We remind here that the JeuxDeMots lexical network does not aim at being more than an average representation of associations between terms.

We asked 30 non-expert persons to undertake four slightly different tasks. Given a word and the set of associated named usages, they had to evaluate the number of missing usages and the number of supernumerary usages (either too specific or plain wrong). The task has to be done on two sets of different 50 words. On the first set, persons are not allowed to consult any dictionary (task Dict-), on the second they can check on dictionaries if they want to do so (task Dict+). The dictionary we proposed for reference is Wiktionary for French, but user where allowed to use any resources they want. Furthermore, the set of words is either taken from common words or non-common words (usually known or unknown at the age of 12).

An example of what is asked to the people is the following (translated for the purpose of this article):

> For the word *sapin*, we propose the following usages:
>
> - sapin(arbre) *(fir - tree)*
> - sapin(Noël) *(fir - Christmas)*
> - sapin(cercueil) *(fir - coffin)*
> - sapin (fiacre) *(fir - hansom)*
>
> Do you think some usages are missing, if yes how many ? Do you think some proposed usages are inappropriate, if yes how many ?

Word usages are ordered by decreasing relevance (as defined in section 3.1).

The four sets of words proposed to each evaluator are completely random (although verifying the constraints described before, that is to say either common or uncommon ones) and they are distinct (a given word may be present only in one set). Two different evaluators may have no distinct sets. The evaluators have all an age above 20 and had a similar proportion of 17 females and 13 males. The level of education was basically 2 years of university or more. The following tables present the collated result of this evaluation.

Common words	**Dict -**	**Dict +**
Missing usages	0.45	1.52
Added usages	0.66	0.37

Uncommon words	**Dict -**	**Dict +**
Missing usages	0.25	1.67
Added usages	0.76	0.28

Tables 1 & 2 : Average number of missing usages and added usages without and with the help of a dictionary for common (table 1) and uncommon (table 2) words used in our evaluation.

For users, added usages are those considered wrong (or at least far fetched). Missing usages are those which should have been present.

C. *Result Analysis*

How can we interpret those results? Without dictionary there is systematically less than one usage felt as missing or added. For the Dict- task, by debriefing the evaluators it appears that the added usage is quite often a proper usage that was unknown to the user (technical, old or rare). Conversely in the Dict+ task, the missing usages value rises as more usages, unknown to the user are found in the dictionary. It seems that globally we are missing much more usages than adding wrong ones. This is quite inline which the way the lexical network is constructed (by players indirect contribution). Missing usages are those quite specific, rare and basically unknown to users.

The task on uncommon words tends to strengthen this analysis. Indeed, without any dictionary people feel that they are more added usages than with common words and less missing usages. The result of the Dict+ is contravariant with the Dict-. Indeed, missing usage value rises and added usage value diminishes.

We can take several precise examples for illustrations. For the word *sapin* (a common word) we got :

sapin	**Dict -**	**Dict +**
Missing usages	0.5	0.9
Added usages	0.2	0.4

Table 3 : Average number of missing usages and added usages without and with the help of a dictionary for the term *sapin (fir)*.

The wiktionnary definitions are: **sapin** *masculin*

1. (Botanique) Arbre conifère résineux de la famille des abiétinées à aiguilles persistantes, au tronc droit, dont le fruit est un cône.
2. Bois de cet arbre utilisé en menuiserie.
3. (Par métonymie) Cercueil.
4. (Familier) (Vieilli) Fiacre.

Some people where doubting about the *sapin (fiacre)* usage although this is a correct one. The usage of *sapin* as wood (mater) is missing but roughly only one person out of two have been thinking of it without checking in a dictionary. The added usage compared to the dictionary is the *sapin (Noël)* although it is present as a locution.

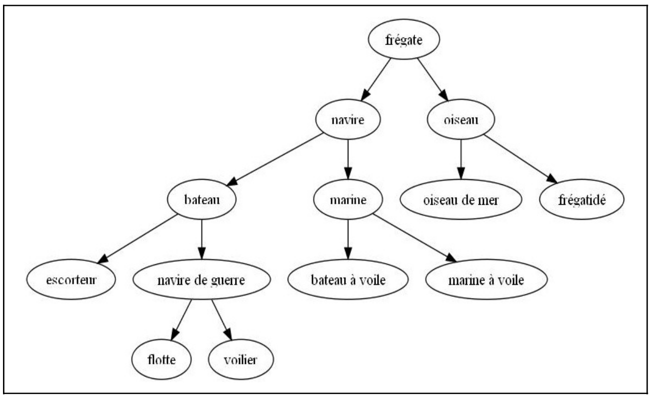


Fig 4 : Usage tree for the term *frégate (frigate)*.

For the word *frégate*, we found out the following usages:

- frégate (navire) *(frigate boat)*
- frégate (oiseau) *(frigate bird)*

frégate	**Dict -**	**Dict +**
Missing usages	0.15	0
Added usages	0.1	0

Table 4 : Average number of missing usages and added usages without and with the help of a dictionary for the term *frégate (frigate)*.

The wiktionnary definitions are: **frégate** *féminin*

1. (Histoire) (Marine) (Militaire) Bâtiment de guerre qui n'avait qu'une seule batterie couverte et qui portait de vingt à soixante bouches à feu.

2. (Zoologie) Oiseau de mer palmipède, d'une très grande envergure, et qui saisit à la surface de l'eau les poissons dont il se nourrit.

For *frégate,* some people made the distinction between the ancient boat and the modern boat. On a rare occasion the evaluator was doubtful on the bird meaning. Comparing with the dictionary we got an exact match.

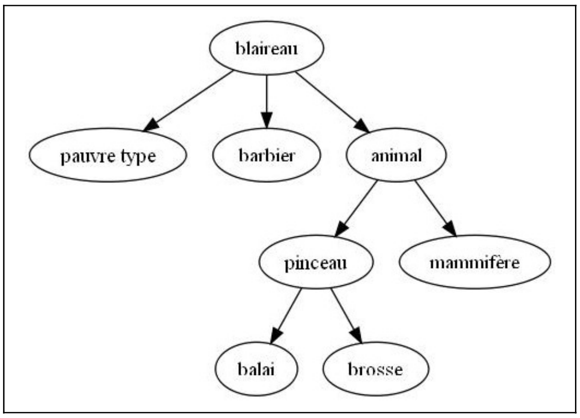


Fig 5 : Usage tree for the term *blaireau (badger, dork, shaving brush)*.

For the word *blaireau*, we found out and proposed the following usages:

- blaireau (animal) *(badger)*
- blaireau (pauvre type) *(dork)*
- blaireau (barbier) *(shaving brush)*

blaireau	**Dict -**	**Dict +**
Missing usages	0.3	1.1
Added usages	0	0

Table 5 : Average number of missing usages and added usages without and with the help of a dictionary for the term *blaireau (badger, dork, shaving brush)*.

The wiktionnary definitions are: **blaireau** *masculin*

1. Mammifère omnivore, bas sur pattes, au pelage noir, gris et blanchâtre, qui se creuse de profonds terriers.
2. (Arts) Brosse en poils de cet animal dont se servent les peintres et les doreurs.
3. Pinceau garni de ces poils dont on se sert, en se rasant, pour étaler et faire mousser le savon.
4. (Argot) (Vieilli) Nez.
5. (Argot) Individu grossier et antipathique ; imbécile, idiot.

For *blaireau* some people found that the *(painting) brush* (meaning 2 of wikipedia) is indeed missing. Most people missed the *nez* (nose – meaning 4) meaning, which is quite old.

All in all, although being preliminary, those results are very encouraging both on the soundness of the method for determining and naming word usage and the quality of the resource collected so far (although evaluating this resource was not the primary goal of this paper). They seem to correspond to what people know and not specifically to some resources made by lexicographers or experts.

What is the effect of the label on the evaluation? If for a given word usage a different label would have been chosen to which extend the results might be modified? In fact there is no much choice for a reasonable label, and generally choosing a substitute like an hyperonym (for example animal instead of bird, in case of frigate) does not alter the results. Of course, this is less and less true as we go deeper in the tree, but also there is less and less choices (if we stick to our labeling approach described in 4.2). Perhaps a deeper evaluation on this particular point should be conducted.

What can we do with the unlabeled usages? So far the answer is simple: nothing. But we should keep in mind that the network is in constant evolution and that some cliques existing now may be fusionned in the near future due to the players activity, or on the contrary being reinforced with new terms allowing them to be labeled.

VI. Conclusion

Viewing the results first obtained, these trees of named usages seem to correspond in their main structures to those a human non-expert would build. In particular, the main branches, directly stemming from the root, correspond in the majority of cases to the meanings of the root term as we could find them in a dictionary. These main branches are subdivided into sub-branches which are so many refinements in the usages.

In their detailed structures however, we notice elements that are different from what a human would have written as we showed on examples above. Are these differences (can we really speak about abnormalities?) due to our method of construction of the trees of the labelled usages with help of players who are not experts, or are they due to the fact that the lexical network is not "complete" enough yet? Moreover, do not the word usage trees sometimes distinguish too subtle refinements? Using the similarity between nodes, how would it be necessary to prune them?

A strong perspective of our work is to propose to the authors of the JeuxDeMots game to insert the identified word usages and proposed them to the players. Hence, the word usages are going to be associated to other terms of lexical database. It would be interesting to assess whether or not reapplying our algorithm leads to some convergence as expected. If it is not the case for a given term, either the usage identified or the label (or both) are not appropriate and should be revised.

VII. References

[1] M. Lafourcade and A. Joubert (2008) Détermination des sens d'usage dans un réseau lexical construit grâce à un jeu en ligne, *Conférence sur le Traitement Automatique des Langues Naturelles (TALN'08)*, Avignon, pp. 189-199

[2] L. von Ahn and L. Dabbish (2004) Labelling Images with a Computer Game, *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 319-326.

[3] R. Mihalcea and T. Chklovski (2003) Open Mind Word Expert: Creating Large Annotated Data Collections with Web Users' Help, *Proceedings of the EACL 2003 Workshop on Linguistically Annotated Corpora (LINC 2003)*, Budapest.

[4] A. Marchetti, M. Tesconi, F. Ronzano, M. Rosella and S. Minutoli (2007) SemKey: A Semantic Collaborative Tagging System, *Proceedings of WWW2007*, Banff, Canada

[5] A. Collins and M. R. Quillian (1969) Retrieval time from semantic memory, *Journal of verbal learning and verbal behaviour*, 8 (2), pp. 240-248.

[6] M. Lafourcade and A. Joubert (2010) Détermination et pondération des raffinements d'un terme à partir de son arbre des usages nommés, *Conférence sur le Traitement Automatique des Langues Naturelles (TALN'10)*, Montréal, (to be published)

[7] J. Véronis (2001) Sense tagging: does it make sense?, *Corpus linguistics' 2001 Conference*, Lancaster, U.K.

[8] S. Ploux and B. Victorri (1998) Construction d'espaces sémantiques à l'aide de dictionnaires informatisés de synonymes. *TAL*, 39(1), pp. 161-182.

[9] C. D. Fellbaum. (1998) WordNet: An Electronic *Lexical Database*. MIT Press, New York.

[10] D. Fišer and B. Sagot (2008). Combining multiple resources to build reliable wordnets. In *TSD 2008*, Brno, Czech Republic

[11] H. Ji, S. Ploux and E. Wehrli. (2003) Lexical knowledge representation with contexonyms. In *Proceedings of the 9th MT summit*, pp. 194-201

[12] A. Tversky (1977) *Features of similarity*, Psychological Review, 84, pp. 327-352

[13] C. D. Manning and H. Schütze (1999) *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge.

[14] C. Fairon and N. D. Ho (2004) Quantité d'information échangée : une nouvelle mesure de la similarité des mots, *Journées internationales d'Analyse statistiques des Données Textuelles (JADT'04)*, Louvain-la-Neuve (Belgique).

[15] A. Budanitsky and G. Hirst (2006) Evaluating WordNet-based Measures of lexical semantic Relatedness, *Computational Linguistics*, 32(1), pp. 13-47.

[16] P. D. Turney and P. Pantel (2010) From Frequency to Meaning: Vector Space Models of Semantics, *Journal of Artificial Intelligence Research*, 37, pp. 141-188.

ISBN 978-83-60810-27-9
ISSN 1896-7094

RefGen: a Tool for Reference Chains Identification

Laurence Longo
LilPa. Université de Strasbourg, 22, rue René Descartes,
67084 Strasbourg cedex, France
Email: Laurence.Longo@rbs.fr

Amalia Todirascu
LilPa. Université de Strasbourg, 22, rue René Descartes,
67084 Strasbourg cedex, France
Email: todiras@unistra.fr

Abstract—In this paper we present RefGen, a reference chain identification module for French. RefGen algorithm uses genre specific properties of reference chains and an accessibility measure to find the mentions of the referred entity. The module applies strong and weak constraints (lexical, morpho-syntactic, and semantic) to automatically identify coreference relations between referential expressions. We evaluate the results obtained by RefGen from a public reports corpus and we discuss the importance of the genre-dependent parameters to improve the reference chain identification.

I. Introduction

IN this paper, we present a new reference chain identification module RefGen, developed for French. Reference chains are linguistic markers indicating a topic shift or a topic continuation in the discourse [1]. RefGen is the main module of a topic detection system, integrated into a topic search engine. The search engine uses topic indexing to help users to retrieve relevant documents from the archives. The topic detection system takes into account the genres of the documents.

Reference chain identification is a key process for many NLP applications as topic detection or text summarization. To solve the reference, the systems identify the various referential expressions (e.g. pronouns, definite noun phrase, possessives) referring the same discourse entity. This is a criterion to form a reference chain in the document. A reference chain includes at least three referential expressions (e.g. *Barack Obama… il… lui 'Bara*ck Obama... he...his' / Le chat qui a mangé l'escalope...le chat...il' the cat that ate the escalope... the cat...it') which denote the same referent [2]. If this referent is common to several sentences of the same paragraph, it represents a potential topic candidate. Several paragraphs sharing the same referent indicates also a topic candidate.

Coreference resolution methods either apply heuristic rules manually defined (which select the most suitable antecedent for a given anaphora) or rules learned from annotated corpora. While supervised learning methods [3], [4] are effective in the processing of coreference relations, they require large, manually annotated training corpora. However, there is currently no large reference corpus annotated with reference chains in French [5] to apply machine learning techniques for reference chain detection. Coreference task of the SEMEVAL2010 conference provides annotated corpora for several languages, but no French data is yet available.

To identify referential expressions we propose a new knowledge poor method as adopted for pronoun [6] and coreference resolution [7], [8], [9]. We select the reference chain elements (mentions of the same entity) using criteria about accessibility and information content of various categories of referential expressions (Accessibility theory [10]), their syntactic function, but also some genre-dependent properties of the reference chains. The RefGen algorithm proceeds in two steps: it first selects the starting element of a reference chain and then it selects the next elements of the reference chain applying strong and weak constraints (lexical, morpho-syntactic, and semantic) [11] between antecedent-anaphora potential pairs.

The paper is organized as follows. In section 2 we describe the referential expressions, the coreference relations processed by our system, and the genre-specific parameters of the reference chains identified by a corpus analysis. In section 3, we present the RefGen module: the genre-dependent parameters used to identify chains, the annotation scheme adopted and the algorithm. We then discuss the first RefGen results obtained from a comparison with manually annotated corpora and we stress the importance of the genre-dependent parameters to improve the results of the algorithm (section 4). In section 5 we conclude and we present future developments.

II. The Reference Chains

A. Referential expressions

Following [2], we consider a reference chain as a relation between at least three mentions (three referential expressions). The reference chains include three types of constituents with a referential function: the proper nouns, the NPs (definite, indefinite, possessive or demonstrative) and the pronouns. The proper nouns have an important role in the discourse structure as they often open a reference chain, due to their capacity to point a unique, well-identified referent.

This work was supported by a French company, RBS (Ready Business System), Strasbourg (http://www.rbs.fr).

Indeed, a study of reference chains in the journalistic portraits [12] shows their importance in organizing the discourse. Apart from cases where there is a referential competition (the repetition of the proper noun eliminates ambiguity between two referents, e.g. “Paul and Pierre… Paul…”), the repetition of a proper noun signals a break in the reference chain. When a referring expression is used, it triggers a "particular recruitment process" of a referent. Thus, the demonstrative (e.g. “ce président”/‘this president’) points directly to the nearest referent while the anaphoric pronoun "il" recruits a referent that is already in mind and there is no concurrent referent [13]. The use of a particular mention (referential expression) is an indication for the reader to remember a specific referent. This referent becomes a local theme of the discourse. However, the use of a complete noun phrase instead of a pronoun is an indication of a reference change. These informations are useful to detect the end of the reference chain or the beginning of a new one.

We decide to first process single referential relations (excluding plural anaphora) between co-referent expressions within a paragraph. We treat direct coreference cases [14] for the coreferential NPs having the same head (eg “le changement climatique” 'climate change' / “ce changement” 'this change') and some indirect coreferences between a person name and a function (e.g. “Barack Obama … le president américain”). Other indirect coreference cases (hyponym/hyperonym) will be treated in the future extensions of the system.

B. Coreference relations

The elements of the same reference chain are related by coreference relations. This means that they are referring to the same entity. These coreference relations have various linguistic expressions: agreement in gender and number between antecedent and anaphor, similar syntactic functions or semantic relations (hyponym/hyperonym, or ontological relations). These properties might be simultaneously or only partially satisfied and they are usually exploited by automatic coreference resolution systems.

Several linguistic theories propose valid interpretations of these properties and rules to detect topic transitions and/or pronoun antecedents. [10] proposes a hierarchy of accessible entities in the discourse. The accessibility is defined in term of information content and rigidity. If the entity is accessible in mind, then it could be expressed by a low accessibility expression as pronoun or possessive. Its antecedent should be a previous nearest entity with high accessibility.

[15] treat the problem of the identification of the main discourse entity in term of focus, or center. The Centering theory uses an order of the possible centers of the discourse, following the syntactic function (the subject of the sentence is the most probable preferred center of the next sentence). This theory predicts topic changes, by defining four categories of focus change or maintenance, but treats only consecutive sentences.

Optimality theory reformulates the rules proposed by the Centering theory [15] in terms of constraints. [16] defines the specific notion of topic sentence as "the entity referred to in both the current and the previous sentence, such as the relevant referring expression in the previous sentence was minimally oblique".

To define the topic sentence as a sign of discourse coherence, [16] proposes a set of constraints:

- AGREE: Anaphoric expressions agree with their antecedents in gender and number
- DISJOINT: Co-arguments of the same predicates should be disjoint
- PRO-TOP: The topic is pronominalized
- FAM-DEF: Each definite NP is familiar, so refers to an entity already mentioned.
- COHERE: The topic of the current sentence is the topic of the previous one
- ALIGN: The topic is in the subject position.

These constraints, which might be reformulated in term of relations between an antecedent and an anaphor, are applied in a hierarchically manner. Moreover, the Optimality theory proposes criteria to select an antecedent from several candidates if it satisfies a maximum of these constraints. As [16] proposes, we also adopt an algorithm selecting antecedent-anaphor pairs by checking various categories of constraints.

C. Genre-dependent properties

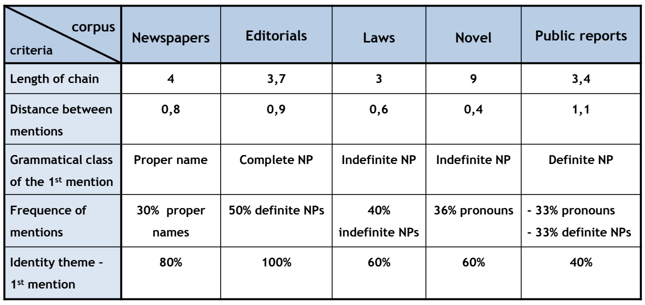
Other parameters used by our algorithm are genre dependent properties. Several studies in textual linguistics [17] aim to characterize genres, text types or registers, by a set of linguistic parameters (the frequency of lexical categories, the preference for some tenses, the frequency of the complex syntactic phrases). These linguistic parameters have a specific communicative purpose and they are used in a particular communicative situation. One category of these linguistic parameters is represented by the reference chain. Genres or types might influence the type of referential expressions used in the text and the choice of the various mentions of the same referent. Cohesion markers as reference chains are dependent on the genre as [12] identifies in newspapers portraits. We assume that reference chains have their specific properties, depending on the textual genre or on the type.

To identify the genre specific properties of the reference chains and to check this hypothesis on other genres, we study the reference chains in a French corpus (about 50,000 tokens) composed of five various genres [18]: newspapers from Le Monde (2004), editorials from Le Monde Diplomatique (1980-1988), a novel Les trois Mousquetaires (Dumas, 1884), some European legal standards from the Acquis Communautaire [19] and public reports from La Documentation Française (2001). We manually annotate the chains to determine which reference chain properties were relevant for a particular genre.

The reference chain study is based on [12]. For each genre, we examine the chains following five criteria:

- the average length of chains (the number of referential expressions referring the same discourse entity);
- the average distance between the elements of a chain (the number of sentences separating the elements);
- the frequency of the mentions depending on their grammatical class;
- the grammatical class of the starting element of a chain;
- the identity between the sentence theme and the first element of a chain.

The study reveals several differences across genres. For example, the average length of reference chains from text laws (Acquis Communautaire) is three mentions while the length is nine mentions for the novel. The difference between the average length of the two genres may be explained in that text laws involve many referents (referential competition between the referents, so many reference chains are opened) while the novel counts lots of descriptions about the main character (which maintains the current chain). Concerning the frequency of the referential categories, we notice that the newspapers contain mostly proper nouns (30.8%) while editorials contain 50% of definite noun phrases. Proper nouns are very frequent starting elements for newspapers reference chains, but indefinite noun phrases are preferred as first mention for text laws and for public reports. Indeed, the measures adopted by the European Commission have a generic scope extended to any state member of the community, hence the massive presence of indefinites (eg. “un Etat Membre”/ ‘Member State’, “une décision”/ ‘a decision’, “une mesure”/ ‘a measure’). In addition, the first element of the chain is identical to the sentence theme for 80% of the occurrences for the newspapers and only for 40% for the public reports. For this last criterion, we checked if it can be possible to gather the reference chains containing the same sentence topic (coreferent reference chains [2]) to identify the document topics.



corpus / criteria	Newspapers	Editorials	Laws	Novel	Public reports
Length of chain	4	3,7	3	9	3,4
Distance between mentions	0,8	0,9	0,6	0,4	1,1
Grammatical class of the 1st mention	Proper name	Complete NP	Indefinite NP	Indefinite NP	Definite NP
Frequence of mentions	30% proper names	50% definite NPs	40% indefinite NPs	36% pronouns	- 33% pronouns - 33% definite NPs
Identity theme - 1st mention	80%	100%	60%	60%	40%

Fig 1. The five genres and their properties

Thus, the corpus analysis of the reference chains highlights their genre-specific properties (Fig 1). We use these parameters to configure RefGen according to the genre.

III. The RefGen Module

In the section above, we present the study of reference chain properties on a corpus composed of several genres. Indeed, the study validates the hypothesis that reference chains have specific linguistic properties depending on the text genre and type (explanatory, narrative etc.). We exploit these properties for reference chain identification. Now, we present the RefGen module architecture and we present the linguistic annotations required (tagging, chunking and Named Entities Recognition). We explain the reference chains algorithm (CalcRef) before presenting the results of the evaluation.

A. The Architecture

RefGen is composed of several modules (Fig.2). The first processing module tags, lemmatizes and annotates the raw input text at chunk level. Then, we apply an annotation module, before proceeding to the reference chain identification step. Among the annotated phrases, we identify complex noun phrases and Named Entities, which represent potential candidates as the first mention of a reference chain. We identify the complex noun phrases among noun phrases modified by several prepositional phrases and/or modified by a relative clause. These phrases are very informative and they precisely identify the referred entity. In addition, in order to avoid wrong anaphora candidates, we annotate impersonal occurrences of the pronoun il. These occurrences are not taken into account by the reference computation module.

Then, after the annotation step, the reference computation module (CalcRef) associates a global accessibility score to each referential expression. Then, the module identifies the anaphora and their possible antecedents. To obtain only valid antecedent- anaphora pairs, the module checks several lexical, syntactic and semantic constraints.

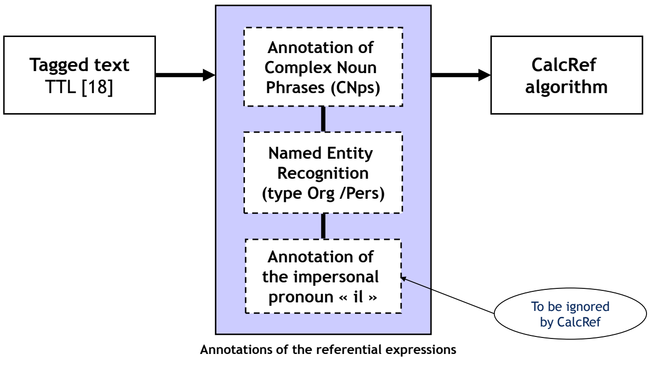


Fig 2. The architecture of the RefGen module

B. Annotations

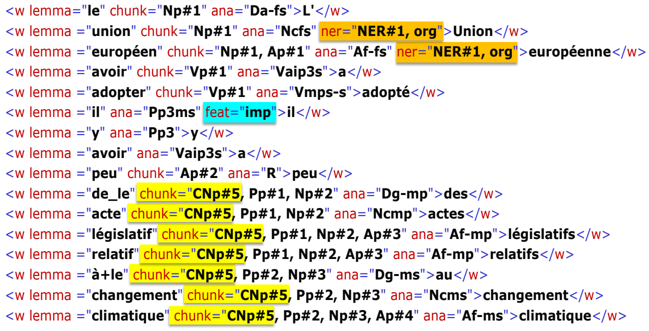
To identify the referential expressions, we tag, lemmatize and chunk the documents using TTL tagger [20].

This tagger identifies chunks (simple noun phrases (NP), prepositional phrases (PP)) and tags some morpho-syntactic properties (tense, mode, person, gender and number). TTL uses the MULTEXT tagset [21].

Then, we apply a set of rules to identify complex NPs (NP modified by at most two PP, NP modified by a relative clause), as “*l'élévation du niveau global de la mer*”/ 'the high of the global sea level' which are more informative than simple NPs. These complex NPs are self-explanatory and they often introduce a new discourse entity.

While we search topic entities, we apply some heuristic rules to identify person and organization names. Persons and organizations are the main actors of the text and they are often related to the main theme of the paragraph. As well, persons and organizations refer to a unique entity and they precisely identify this entity, so they represent good candidates to be a first mention of a reference chain.

In addition, we annotate the French impersonal pronoun il (e.g. *“il pleut”/'*it rains*'*) to eliminate non-anaphoric use of this pronoun. Fig.3 is an example of annotations including lemmas (`lemma`), chunks (simple `Np`, `Pp`; complex `CNp`), morpho-syntactic properties (`ana`), named entities (`ner`) and impersonal pronouns il (`feat="imp"`). We use these linguistic annotations to identify the reference chains and the anaphora pairs.



```
<w lemma="le" chunk="Np#1" ana="Da-fs">L'</w>
<w lemma ="union" chunk="Np#1" ana="Ncfs" ner="NER#1, org">Union</w>
<w lemma ="européen" chunk="Np#1, Ap#1" ana="Af-fs" ner="NER#1, org">européenne</w>
<w lemma ="avoir" chunk="Vp#1" ana="Vaip3s">a</w>
<w lemma ="adopter" chunk="Vp#1" ana="Vmps-s">adopté</w>
<w lemma ="il" ana="Pp3ms" feat="imp">il</w>
<w lemma ="y" ana="Pp3">y</w>
<w lemma ="avoir" ana="Vaip3s">a</w>
<w lemma ="peu" chunk="Ap#2" ana="R">peu</w>
<w lemma ="de_le" chunk="CNp#5, Pp#1, Np#2" ana="Dg-mp">des</w>
<w lemma ="acte" chunk="CNp#5, Pp#1, Np#2" ana="Ncmp">actes</w>
<w lemma ="législatif" chunk="CNp#5, Pp#1, Np#2, Ap#3" ana="Af-mp">législatifs</w>
<w lemma ="relatif" chunk="CNp#5, Pp#1, Np#2, Ap#3" ana="Af-mp">relatifs</w>
<w lemma ="à+le" chunk="CNp#5, Pp#2, Np#3" ana="Dg-ms">au</w>
<w lemma ="changement" chunk="CNp#5, Pp#2, Np#3" ana="Ncms">changement</w>
<w lemma ="climatique" chunk="CNp#5, Pp#2, Np#3, Ap#4" ana="Af-ms">climatique</w>
```

Fig 3. Annotations used to detect reference chains

C. The CalcRef Module

CalcRef is the main module of RefGen and it proceeds to reference chain identification by using genre-specific parameters and the linguistic annotations presented in the previous sections. Thus, we specify the genre of the indexed documents and we configure CalcRef according to the properties of the reference chains (average distance between the mentions, average chain length, and the preferred category of the first element of a chain). For example, for a corpus of public reports, the length of the chains is 4, the average distance is 2 sentences and the preferred type of the first element is a complete definite NP. To select the mentions of the reference chain, we define a set of weak and strong constraints, explained later in this section.

The algorithm follows the next steps:

```
 For each p paragraph:
 n=1st phrase of the paragraph;
 d= the average distance between candidates
(number of phrases)
 A.  Select the 1st mention candidates from
the complex noun phrases and the Named
Entities (their AG is greater than or equal
to 190).
 B.  Order the list of the first mentions
(according to global accessibility GA, the
function and the type), open several chains;
 C.  for each phrase, select the list of
anaphora candidates (GA less than 190). We
exclude the anaphoric use of the impersonal
pronouns.
   a. for each anaphoric candidate, select
 the candidate pairs checking the strong
 constraints (without reflexive pronouns)
 and apply weak constraints.
   b. order the pairs according to the
 number of checked constraints and select
 the pair checking the maximum number of
 constraints.
   c. recompose the reference chain from the
 identified pairs
 D. n=n+d and start again at A.
```

For each paragraph, CalcRef selects candidates for the first mention of the reference chains. Then, CalcRef identifies the next elements of the reference chains, by selecting a set of pairs of antecedent-anaphora candidates. Most of the candidates are filtered out by the application of several constraints. Then, we apply the transitivity of the coreference relation to compose the reference chains.

Ordering the referring expressions

For each paragraph, CalcRef selects candidates as first mention of the reference chains among expressions with a high degree of accessibility. [10] defines an accessibility hierarchy to classify the referential expressions according to their accessibility: less the referent is accessible, the referential expression should be longer, self-explanatory and rigid. Thus, indefinite NPs, proper nouns or complex NP, occupying the thematic position are used to mention a new entity (Table I), while short mentions as pronouns might be used to refer to entities already specified in the discourse. The global accessibility AG is computed by combining three elements: informativity (the amount of lexical information), rigidity (the possibility to pick up a specific referent) and attenuation (phonological size). With respect to the initial accessibility hierarchy, we also add indefinite noun phrases in the accessibility scale, even if this category of candidates generates several errors. We use weights on a scale of 10 to 110 for each of these elements (e.g. the global weight of the complete proper noun “Le président Barack Obama” is 220 while it is 150 for the pronoun “elle”).

TABLE I.
ACCESSIBILITY TABLE

Referential expressions	Informativity	Rigidity	Attenuation	Global Accessibility
Indefinite noun phrase	110	110	10	230
Complete proper noun	100	100	20	220
proper noun	90	90	30	210
Complex definite noun phrase	80	80	40	200
Simple definite noun phrase	70	70	50	190
Last name	60	60	60	180
First name	50	50	70	170
Demonstrative	40	40	80	160
Pronoun	30	30	90	150
Reflexive pronoun	20	20	100	140
Possessive	10	10	110	130

CalcRef computes the global weight of the candidates as a sum of the global accessibility weight and the syntactic role weight. We also define a scale for the syntactic role weights: 100 for the subject position, 50 for the direct object position, 30 for the indirect object and 20 for other syntactic functions. Then, genre dependent parameters (the preference for the first element type or the distance between the mentions) are used to increase the weight (+50) of some candidates (for example, if we treat law texts, indefinite NPs are preferred as starting elements of a chain). We order the first element candidates according to the global weight and we select the highest weight candidate as a first element of the current chain. We open a new chain for each element of the candidate list.

In addition, we use the accessibility scale to propose possible antecedent-anaphora pairs. The antecedent should have the global accessibility higher than the anaphor.

Searching valid antecedent-anaphora candidates

CalcRef selects the next elements of the reference chain from highly accessible expressions (pronouns, demonstratives etc.). We establish a set of possible antecedents from low accessible expressions. We combine elements from the two sets and we identify potential antecedent-anaphora pairs. The distance between the two elements of the pair should be less than the average distance defined by the genre parameters.

Then, we adapt the method proposed by [11]. This method checks several constraints between antecedent and anaphora to filter out impossible pairs. Indeed, [11] present a system implementing a constraint-based method for pronoun resolution inspired by the Optimality theory [16].

If the antecedent and the anaphor refer to the same discourse entity, they satisfy a set of constraints defined in section II (chapter B). These constraints are syntactic (similar syntactic function between the antecedent and the anaphor), morpho-syntactic (agreement in gender or in number) or semantic (hyponyms/hyperonyms). A pair antecedent-anaphor satisfying a maximum number of constraints is a valid candidate to be included in a reference chain. [11] adapts these constraints to several languages and proposes an implementation of the algorithm. In addition, the order of the constraints might be changed to obtain better results.

The Optimality theory limits the search space of the antecedent at the previous sentence. [11] propose the algorithm only for pronoun resolution. We extend the set of constraints to other anaphora categories (definite expressions, reflexive pronouns).

Following [11], we adapt the constraints for French. For each pair, we check some strong and weak constraints.

Weak constraints mean that they might be violated, even if there is a valid antecedent-anaphor pair:

- **MORPHO** – agreement in gender or number (between the personal pronoun and the candidate);
- **SYN** – the antecedent and the anaphora should have similar syntactic function;
- **SEM** – semantic relations between the antecedent and the anaphora (for example, person names might be valid antecedents of a NP expressing a function ({B. Obama – le président des Etats-Unis});
- **PROX** – the antecedent and the anaphor are near neighbours (for possessives and demonstratives).

Strong constraints must be satisfied:

- **IMB** – the imbrications mean that an element must not be nested in its antecedent, as [la soeur [de Marie]]), or co-arguments of a verb should not be coreferent;
- **TETELEX** – the identity between NP's head and the partial repetition of the same proper noun.

Moreover, for some specific anaphora, it is necessary to define the set of strong constraints to be satisfied. For example, for possessives or reflexives, the constraint **IMB** (checking the arguments of the verbs) is not useful.

If a pair fails to satisfy a strong constraint, then the pair is deleted from the candidate list. For each candidate pair satisfying all the strong constraints, we check the number of the weak constraints that are satisfied. If several pairs satisfy the same number of constraints, we keep the valid pairs into a large list.

For the semantic constraints, we apply the method proposed by [22]. We use a resource extracted from a 500,000 tokens corpus from computer science newspapers: we extract the occurrences of all the main verbs and their subjects. This resource is used to select a valid antecedent for the pronoun, when several possible antecedents satisfy the same number of constraints. For example, we search the antecedent of the pronoun "il*"* in:

*"*Le camion est passé en vitesse. Le chien a eu peur. Maintenant **il** mange son os". 'The truck was driven very fast. The dog was afraid. Now it eats *his bone.'*

The two candidates *"le camion"/'the truck'* and *"le chien"/'*the *dog'* satisfy the same number of constraints. To decide between the two candidates, we consult the resource. The verb *"*mange*"/'*to *eat'* has as subject *"*le chien*"/'*the *dog'* but no occurrences of a subject *"*camion*"/'truck'* are present in the resource. We deduce the preferred antecedent for "il" is *"*chien*"/'dog'*, because an animal is able to eat, but not an inanimate object.

Building reference chains

Then, we start from the first element of the chain and we search the pairs having this candidate as antecedent in the list. To build the reference chain, we apply the transitivity of the reference relation: if A is antecedent of B and B is antecedent of C, then they are part of the same chain. For example if we have three pairs "J. Chirac – il"; "il – il"; "le président - il" we can deduce that we have a reference chain with four mentions: {J. Chirac, il, le président, il}. We continue the process until the length of the current reference chain is greater than the average gender-specific length. We annotate the candidate pairs identified as part of the current reference chain.

We restart the whole process after selecting the next first candidate element from the ordered list of the current paragraph. The process is launched for every paragraph of the document.

IV. An Example

We present a full example processed by RefGen. We note the various entities mentioned in the discourse with small letters (i, ii, j, k, l, m, o, n, p, q, r, t, s, v, w, z). The example is extracted from a white paper of the European Commission about the climate change.

[**La lutte contre [le changement climatique]ii**]i doit se faire [à deux niveaux]j. Il s'agit d'abord et avant tout de réduire [les émissions de gaz à effet de serre]k ([au moyen [de mesures d'atténuation]l]m), puis de prendre [les mesures d'adaptation qui s'imposent]o pour faire [face aux conséquences inévitables de **[ce changement]p**]n. [L'Union européenne]q ([UE]r) a adopté il y a peu [des actes législatifs relatifs [au changement climatique]t]s, qui définissent [les mesures concrètes nécessaires à la réalisation de l'objectif fixé par [l'UE]v]w, à savoir réduire [les émissions de 20 % par rapport aux niveaux de 1990]z d'ici à 2020.

The algorithm first identifies the entities with a high global accessibility (Proper Nouns, indefinite descriptions or complex definite descriptions). In this genre (public reports), complex definite descriptions are very frequent. So RefGen identifies as potential first mentions the following candidates:

- **[La lutte contre le changement climatique]i,**
- les émissions de gaz à effet de serre]k,
- [au moyen [de mesures d'atténuation]l]m,
- les mesures d'adaptation qui s'imposent]o
- [face aux conséquences inévitables de [ce changement]p]n,
- [L'Union européenne]q,
- [des actes législatifs relatifs [au changement climatique]t]s,
- [les mesures concrètes nécessaires à la réalisation de l'objectif fixé par [l'UE]v]w
- [les émissions de 20 % par rapport aux niveaux de 1990]z.

The candidates are sorted by their global weight (the sum of the global accessibility, the syntactic function and the preference for the first mention category).The most probable first mention are i, q, s, w. We open 4 reference chains and we try to find the next pairs.

Then, for each first mention candidate, we establish a list of anaphor candidates (having the accessibility less or equal than 190): definite descriptions (ii, l, m, o, n, p, z), pronouns (il, il). Both occurrences of il are impersonal, so we check the validity of the constraints for definite descriptions. In this case **TETELEX** is the first constraint to be checked. For example, for the entity i, there is no other mention explicitly referring to "la lutte". But we found a reference chain starting from the entity ii.

We notice that several strong constraints **TETELEX** or **IMB** are violated for "[ce changement]p"(Table II). A constraint violated is marked as '*', a space means that the constraint is checked. We find many direct coreference cases, while pronouns are quite few and their use is impersonal.

TABLE II.
VALIDATION OF THE CONSTRAINTS FOR THE CANDIDATE P

	Id	MORPHO	IMB	SYN	SEM	PROX	TETELEX
p	i						
	ii			*			
	l	*					*
	m			*			*
	o						*
	k	*		*			*
	n	*	*				*

In contrast, we note an example extracted from the newspapers Le Monde diplomatique.

[M. Pons]i affirme en outre, dans [un entretien publié par le Figaro du 21 septembre]j, que " [l'immense majorité [des députés RPR]k]l souhaite [le calme et la sérénité]m et qu'[ils] se détermineront [le moment venu]n ". Minimisant [la fracture ouverte]p entre "balladuriens " et " chiraquiens ", **il** rappelle que [Jacques Chirac]r **lui** apparaît comme " [le candidat légitime]q " de **son** parti.

The algorithm first identifies the entities with a high global accessibility (Proper Nouns, indefinite descriptions, complex definite descriptions). So RefGen identifies as potential first mentions the following candidates:

- **[M. Pons]i,**
- [l'immense majorité [des députés RPR]k]l,
- [le calme et la sérénité]m,
- [le moment venu]n,
- [la fracture ouverte]p,
- [Jacques Chirac]r,
- [le candidat légitime]q.

The candidates are sorted by their global weight (global accessibility, and the syntactic function).The most probable first mention are i, l, r.

Then, we establish a list of anaphor candidates (having the global accessibility less than 190): definite descriptions (l, m, n, p, q), pronouns (ils, il, lui), possessives (son).

TABLE III.
VALIDATION OF THE CONSTRAINTS VALIDATION FOR THE PRONOUNS "IL" AND "LUI"

	id	MORPHO	IMB	SYN	SEM	PROX	TETELEX
il	i					*	-
	j			*		*	-
	l	*				*	-
	k	*		*		*	-
	n						
lui	i			*		*	-
	j			*		*	-
	l	*		*		*	-
	k	*		*		*	-
	n			*		*	-
	p	*		*		*	-
	r				*		

We notice that several strong constraints are violated between i and the definite descriptions. The pairs (i, il) and (i, lui) are the most probable (Table III**).**

V. EVALUATION

We present the first results of the evaluation of RefGen, we compare the reference chains extracted automatically against the manually annotated corpus. We present the results obtained for the CNp annotation module, for the NER module and for the chain identification module. The evaluation corpus is a small corpus (7,230 tokens) composed of public reports of the European Commission about the measures adopted by EU to limit the effects of the climate changes. We compute the recall, the precision and the f-measure of the intermediate modules, as well as the results for CalcRef. We check the results obtained for independent antecedent-anaphora pairs, as well as for reference chains (Table IV).

The NER annotation errors are due to the wrong identification of some acronyms or abbreviations (e.g. GES : gaz à effet de serre) which were annotated as organization names. Some NER annotation errors are due to tagging errors. The CNp identification module fails to identify several CNps (an NP modified by more than three PP), which were not described by the existing set of patterns. Indeed, the test corpus is characterized by very frequent, complex, informative NP.

The evaluation corpus contains 118 anaphoric pairs, but only 24 reference chains. Several antecedent-anaphora pairs are wrongly selected, due to tagging errors or due to the lack of external knowledge sources. For example, some of the antecedent-anaphora pairs were selected because they satisfy the same number of constraints (number, gender, syntactic function). An ontology might help to select the right antecedent.

We tested the system for three various configurations of the genre parameters. First, we use three genre-specific parameters:

(a) distance=2; length=4; preferred type=definite description.

We also tested the system after changing these parameters:

(b) we ignore these parameters and use only a default distance of 20 sentences

(c) we use the newspapers parameters (distance=1; length=3; preferred type = proper nouns).

If we ignore the parameters (case (b)), we obtain more antecedent-anaphora pairs than in the first case, due to the bigger distance between the mentions. Meanwhile, we obtain less reference chains because several smaller reference chains are grouped together. For all the cases, we obtain quite similar results for the pairs, but for the reference chain identification we obtain significant performance decreasing (b) f_measure: 0,51; (c) f-measure: 0,54 (Table IV).

TABLE IV.
FIRST EVALUATION RESULTS

	NER	**CNp**	**CalcRef (pairs)**	**CalcRef (reference chains)**
Recall	0,85	0,87	0,69	0,58
precision	0,91	0,91	0,78	0,70
f-measure (a)	0,88	0,89	0,73	0,63
f-measure (b)	0,88	0,89	0,71	0,51
f-measure (c)	0,88	0,89	0,70	0,54

VI. CONCLUSION

We presented RefGen, a reference chain identification method, developed for French. This new knowledge poor method uses a set of detailed linguistic annotations and the accessibility hierarchy of the referring expressions to select possible antecedent-anaphora pairs. Then, a set of lexical, syntactic and semantic constraints are used to filter some invalid pairs. RefGen also uses some genre-dependent properties of the reference chains (average length, preferred type of the first element, average distance separating several mentions of the same referent). These genre-dependent properties were identified from a corpus-based analysis. We describe a new algorithm designed to identify the reference chains and we present a first evaluation of the module. The evaluation is done with several genre-specific parameters. The evaluation results show an improvement of the results when we use the public reports parameters. The system is flexible; it is possible to add extra constraints to improve the quality of the output.

In the future, the module will be integrated into the topic detection system to be tested in real-life applications. The module will be extended to treat other cases of coreference: plural anaphora, hyponym/hyperonym equivalents, by adding knowledge sources as ontologies or synonym databases.

RefGen will be used to annotate large French corpora with coreference relations. It will then contribute to the development of a reference corpus for French, comparable with those provided by SEMEVAL for other languages.

In addition, future work concerns the adaptation of the system for other languages.

REFERENCES

[1] F. Cornish, Références anaphoriques, références déictiques, et contexte prédicatif et énonciatif. Sémiotiques, 8, pp. 31-57, 1995.

[2] C. Schnedecker, Nom propre et chaînes de référence. Recherches Linguistiques 21. Paris : Klincksieck, 1997.

[3] V. Ng and C. Cardie, "Improving machine learning approaches to coreference resolution", in Proceedings of the ACL (Association For Computational Linguistics), Morristown, pp. 104-111, 2002.

[4] V. Hoste, Optimization Issues in Machine Learning of Coreference Resolution. PhD thesis, 246 p, 2005.

[5] S. Salmon-Alt, Référence et Dialogue finalisé : de la linguistique à un modèle opérationnel. PhD thesis, Université H. Poincaré, Nancy, 2001.

[6] R. Mitkov, "Towards a more consistent and comprehensive evaluation of anaphora resolution algorithms and systems," Applied Artificial Intelligence: An International Journal, 15, pp. 253-276, 2001.

[7] S. Hartrumpf, "Coreference Resolution with Syntactico-Semantic Rules and Corpus Statistics," in Proceedings of CoNLL (Computational Natural Language Learning Workshop), 2001.

[8] A. Popescu-Belis, Modélisation multi-agent des échanges langagiers : application au problème de la référence et à son évaluation. PhD thesis, Université Paris-XI, 1999.

[9] K. Bontcheva, M. Dimitrov , D. Maynard , V. Tablan, and H. Cunningham, "Shallow methods for named entity coreference resolution," in Proceedings of TALN 2002, 2002.

[10] M. Ariel, Accessing Noun-Phrase Antecedents, London: Routledge, 1990.

[11] W. Gegg-Harrison and D. Byron, "PYCOT: An Optimality Theory-based Pronoun Resolution Toolkit," in Proceedings of LREC 2004, Lisbonne, 2004.

[12] C. Schnedecker, "Les chaînes de référence dans les portraits journalistiques : éléments de description," Travaux de Linguistique 51, pp. 85-133. Duculot, 2005.

[13] G. Kleiber, Anaphores et Pronoms. Louvain-la-Neuve : Duculot, 1994.

[14] H. Manuélian, Description Définies et Démonstratives : Analyses de Corpus pour la Génération de Textes. PhD thesis, Nancy 2, 2003.

[15] B. J. Grosz, S. Weinstein, and A. K. Joshi, "Centering: a framework for modeling the local coherence of discourse," Computational Linguistics 21(2), pp. 203-225, 1995.

[16] D. Beaver, "The optimization of discourse anaphora," Linguistics and Philosophy, 27(1): pp. 3–56, 2004.

[17] D. Biber, "Representativeness in corpus design," Linguistica Computazionale, IX-X, Current Issues in Computational Linguistics: in honor of Don Walker, 1994.

[18] L. Longo and A. Todirascu, "Une étude de corpus pour la détection automatique de thèmes," in Proceedings of the 6th journées de linguistique de corpus (JLC 09), Lorient, 2010.

[19] R. Steinberger, B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufiş, and D. Varga, "The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages," in Proceedings of the 5th LREC Conference, pp.2142-2147, 2006.

[20] R. Ion, TTL: A portable framework for tokenization, tagging and lemmatization of large corpora. Bucharest: Romanian Academy, 2007.

[21] N. Ide and J. Véronis, "MULTEXT (Multilingual Tools and Corpora), in Proceedings of the 14th International Conference on Computational Linguistics, Kyoto,1994.

[22] I. Dagan, A. Itai. "A statistical filter for resolving pronoun references". In Y. A. Feldman and A. Bruckstein, editors, Artificial Intelligence and Computer Vision, pages 125--135. Elsevier Science Publishers B.V, 1991.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 455–460

ISBN 978-83-60810-27-9
ISSN 1896-7094

Is Shallow Semantic Analysis Really That Shallow? A Study on Improving Text Classification Performance

Przemysław Maciołek, Grzegorz Dobrowolski
AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland
Email: {pmm,grzela}@agh.edu.pl

***Abstract*—The paper presents a graph-based, shallow semantic analysis-driven approach for modeling document contents. This allows to extract additional information about meaning of text and effects in improved document classification. Its performance is compared against the "legacy" bag-of-words and Schenker et al. approaches with $k-NN$ classification based on Polish and English news articles.**

I. Introduction

RESEARCH on computational linguistics has over 50 years of history. It is currently considered as an interdisciplinary field covering topics such as linguistics, psychology, cognitive science, artificial intelligence and others. While the research is done for more than a half of a century, there are still many basic open issues. One of such notable problems is relatively poor ability of machine text content understanding.

The state-of-the-art in modeling text meaning for document classification purposes is practically still basing on a statistical bag-of-words approach, developed in early 70's. In this method, single words are extracted and represented by a vector with their frequency.

Although many enhancements have been proposed to this approach, it has essential drawback: substantially limited amount of information that can be "captured" by such representation. Omission of information about words order is a striking example.

Numerous solutions leveraging graph-based representation of text were presented. One of such method was proposed by Schenker, Last, Bunke and Kandel [23], [22], [21]. The taken approach built document model as graph, where each sequentially found word was added as a next node. Also, links between nodes were tagged with information about section of the document where they were found (such as *title*, *body* or *link*). Thus instead of just calculating frequency of word occurrences, information about their order is also stored.

Other graph-based methods include graph node ranking method similiar to *PageRank* [15], analyzing structure of the document only (ignoring its contents) [8], extracting document features from previously built tree-like structures [10] and extracting features from Schenker *et al.* based graphs [14].

In this paper a new method is presented. Like in the Schenker *et al.* approach, document contents is represented using a graph model. However, the way the graph is built is different and takes into account part-of-speech information about each word. Also, semantic dictionary (such as *WordNet*) is used in the build process. The reasoning behind it is based on observations of language acquisition by children [24], [16] and statistical part-of-speech usage [12], [2].

The new method is compared to well-known Schenker *et al.* and bag-of-words approaches in document classification task. Test results are presented and commented in the final section of the paper.

II. Document Modeling Approaches

In this section, baseline text modeling methods are presented. As a prerequisite, they require processing raw document contents into a form that allows to extract relevant document features. The process might be summarized as an algorithm consisting of following steps:

1) Reading text from a source. Segmentation into sentences and words. Converting all letters to common case.
2) Removing stop-words. These are frequently found words, that do not provide substantial information about document contents (as they are commonly found in any kind of text). The stop-words list is arbitrary and might contain from zero to hundreds of such words. Typical elements are: *"the"*, *"in"*, *"he"*, *"one"*, *"of"*, *"is"*, etc.
3) Word stemming. In this process, the inflected or derived words are reduced to their stems (non-changeable parts). This allows properly treating words of the same base used in different forms (e.g. *"thankfully"* → *"thank*"*, *"thanks"* → *"thank*"*, etc.).

Vector Space

The most commonly used approach for document representation is to count each word occurrence, calculate its frequency and put it into a vector space. This allows to easily find distance between any two documents (vectors) using a selected metric, such as *Euclidean* or *Jaccard* distance [13].

It is worth noting that while the general idea of vector space text representation has not changed during the last 40 years, many methods improving selected vector features quality have emerged, such as *Latent Semantic Analysis*.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Schenker Text-to-Graph Approach

Schenker *et al.* [23] proposed a method (actually a variant of more general *text-to-graph* approach) for building a graph from hypertext. The process first marks three sections of the text: *title*, *links* and *text*. Next, the graph is being built using following rules:

1) If word (stem) A occurs for the first time, then new node A is created.
2) If word B occurs after word A in section 1, a connection $A \rightarrow B$ is created with label 1.

To reduce the graph size, only the n most relevant words are selected from the document for building the graph. Word occurrence counters are not incorporated (in the typical variant).

Definition 1: To calculate distance between any two graphs, following metrics are proposed:

$$dist_1(G_1, G_2) = 1 - \frac{|mcs(G_1, G_2)|}{max(|G_1|, |G_2|)} \quad (1)$$

$$dist_2(G_1, G_2) = 1 - \frac{|mcs(G_1, G_2)|}{|G_1| + |G_2| - |mcs(G_1, G_2)|} \quad (2)$$

$$dist_3(G_1, G_2) = 1 - \frac{|mcs(G_1, G_2)|}{|MCS(G_1, G_2)|} \quad (3)$$

where:
$mcs(G_1, G_2)$ - maximum common subgraph of graphs G_1 and G_2
$MCS(G_1, G_2)$ - minimum common supergraph of graphs G_1 and G_2
$|G|$ - size of graph G, defined as a sum of numbers of nodes (vertices) and edges: $|G| = |V| + |E|$

To calculate the distance, maximum common subgraph has to be found first. Finding such subgraph is, in general, a NP-complete problem. However, when taken into account that in the created graphs each node has a unique label, it is possible to construct an algorithm finding the solution with $O(|V|^2)$ computational complexity [6], [21].

The originally presented approach (*standard representation*) was further extended. One of the interesting variants was incorporating information about word counts into the graph model and discarding information about document section (such as *link*, *body* and *title*). Each node and edge was labeled with additional information about number of times each associated term appeared in the document (in case of nodes) and number of times two nodes were connected together (in case of edges). These numbers were stored as absolute values (*absolute frequency representation*) or as normalized values - divided by the maximum number of node occurrences in document (*relative frequency representation*).

Definition 2: Graph size used for models using frequency information is defined as:

$$|G|' = \sum_{i=1}^{|V|} v(i) + \sum_{j=1}^{|E|} e(i)$$

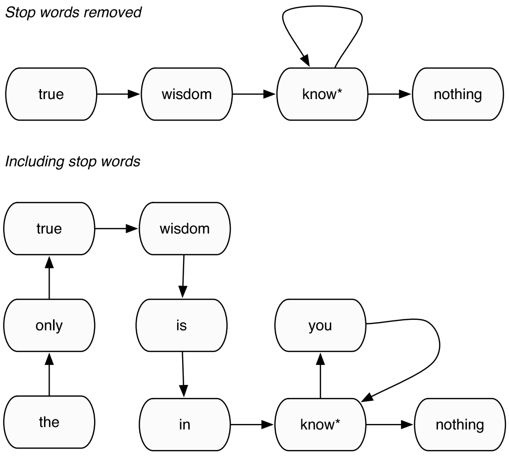


Fig. 1. Example of Schenker graph modeling for sentence *"The only true wisdom is in knowing you know nothing"*. Note: *know** is a stem of both *know* and *knowing*.

where:
$|V|$ - number of nodes in graph G,
$|E|$ - number of edges in graph G,
$v(i)$ - frequency associated to node i
$e(j)$ - frequency associated to edge j

According to the results obtained by Schenker, the *standard representation* provided generally better results for hypertext (HTML) documents in comparison to other *text-to-graph* algorithm variants. When information about links and title was not incorporated (such as when using *simple representation*) the performance slightly decreased. It might be expected that for documents with low number of hyperlinks (or without any of them) the frequency-type representation will provide the best results.

III. MOTIVATION

Almost any linguistics-related problem is still solved drastically better by humans than by machines. A human language is considered as one of the greatest achievements of evolution, practically unique for mankind. Even after so many years of research, we still possess relatively not too much knowledge about the way it really works. Basically, we do not even understand the mechanism in which the language is acquired (learned).

Our lack of knowledge in this area does not allow creating a general tool that would render a text understanding performance comparable to humans. However, while we do not (yet) see the "whole picture", a lot of observations about language development were collected during the years. This knowledge may help to improve computational linguistics mechanisms.

Verbs Absence During Language Acquisition

An important observation is that children learn primarily nouns, even if they can observe other parts of speech with similar frequency [24]. There is a huge disproportion: when child dictionary contains between 20-50 words, as much as

45% of them are nouns, and only 3% of them are verbs. While these numbers were observed for English language, in case of other languages the situation is very similar [3].

This finding might be linked as a consequence of the fact, that nouns are basically used for object labeling (such as "mom", "dad", "bed", "cat") and thanks to this they are relatively firm. The child usually knows elements (classes) which are named by nouns. On the other hand, the verbs might describe an action, state or occurrence and might have different meaning depending on the arguments and context used. Also, they might be often found in sentences describing abstract ideas. In effect, an advance of child development is required for possessing skills necessary for understanding and using verbs.

Polysemy and Parts of Speech Distribution

The WordNet 3.0 [7] database contains more than 155,000 unique strings, which are categorized either as nouns (117798 unique strings), verbs (11529), adjectives (21479) or adverbs (4481). Each string is assigned to one or more synsets (groups of words with similar meaning).

As it might be observed, there's notable difference in ratios of polysemous to monosemous words for different parts of speech (see table I). It is noticeably higher for verbs in comparison to other parts of speech.

It is reversed in case of nouns. Even if the absolute number of them is much higher than any other part of speech, the ratio is very low. Similar findings might be also found when analyzing large text corpora for other languages [12], [2].

TABLE I
WORDNET POLYSEMY STATISTICS [7]

POS	AVERAGE POLYSEMY (INCLUDING MONOSEMOUS WORDS)
NOUN	1.24
VERB	2.17
ADJECTIVE	1.40
ADVERB	1.25

Summary of Observations

An analogy can be observed: capabilities of machine text processing (classification, information retrieval, etc.) might be compared to skills of a young child, which does not yet posses general knowledge about the surrounding world.

Based on the presented observation, a following hypothesis might be suggested: the sub-optimal methodology for unstructured text processing is similar to the one observed for children language capabilities. That is, an emphasis should be put on information that are not troubled by ambiguity (such as nouns). Also, because there are many ways an action can be represented with different verbs, a method for generalizing their meaning (for example using synonyms from semantic dictionary) could improve machine text representation.

IV. SHALLOW SEMANTIC ANALYSIS FOR BUILDING A GRAPH MODEL

The method presented in this section is a special case of a more general solution, which is a member of family of methods based on the shallow semantic analysis approach for building a graph model. The algorithm presented in this paper is a variant, that was experimentally found to be sub-optimal for a document classification task. It is based on observations presented in the previous section and simplifications in predicate-argument sentence decomposition.

As a prerequisite, a number of steps must be performed on the input text. The complete process of transformation contains following phases:

1) Reading raw text from a source.
2) Text segmentation and removal of non-characters.
3) Tagging parts of speech; *Stanford POS Tagger*[1] [27], [26] is used for English and *TAKIPI*[2] [17] for Polish documents. The cited accuracy is approx. 97% for *Stanford POS Tagger* and 93.4% for *TAKIPI* (in the latter case, including gender and case resolution).
4) Finding synsets using semantic dictionary; *WordNet*[3] [7] and *plWordNet*[4] [18] are used for this purpose.
5) Word stemming. In case of English documents *Snowball Stemmer*[5] was used. For Polish, *Morfologik*[6] was chosen.

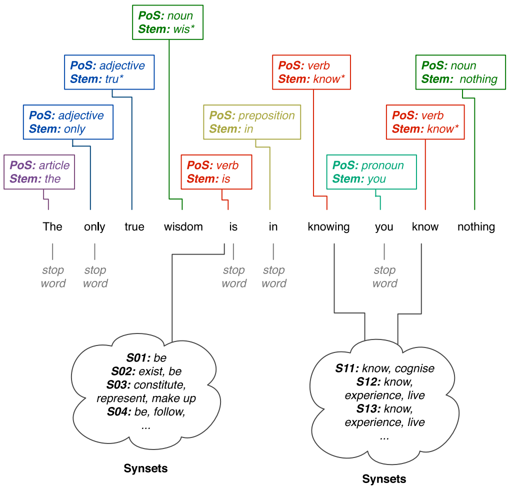


Fig. 2. Example of sentence decomposition

The processed document contents is sequentially analyzed, word after word. Graph representing the text is built using the following set of rules:

1) If a word A is a noun or adjective, and a node with label A does not yet exist, it is created. In case the word occurs rarely in the whole data set (e.g. less than

[1] http://nlp.stanford.edu/software/tagger.shtml
[2] http://nlp.ipipan.waw.pl/TaKIPI/
[3] http://wordnet.princeton.edu
[4] http://plwordnet.pwr.wroc.pl/
[5] http://snowball.tartarus.org/
[6] http://morfologik.blogspot.com

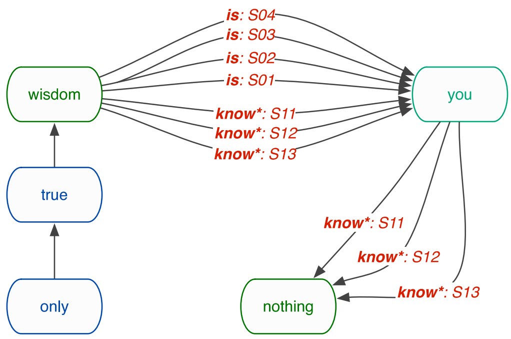


Fig. 3. Example of shallow semantic analysis based text-to-graph transformation result

predefined number of times), it is being replaced by its synsets.

2) If a noun or adjective B is found after another noun or adjective A, then a connection with empty label is created between nodes labeled with A and B.
3) If a verb C was found between nouns or adjectives A and B, then all synsets (sets of words with similar meaning and the same part-of-speech) of C are being added as connection between nodes A and B, labeled with synset identifier.
4) If adjective E is found between two nouns D and F, an additional direct connection between D and F is created.

New distance measures are proposed. They take into account not only the single maximum common subgraph, but all nodes and edges that were found to be common:

Definition 3: To calculate distance between any two shallow semantic analysis created graphs following measures will be used:

$$dist_4(G_1, G_2) = 1 - \frac{|G_1 \cap G_2|}{|G_1| + |G_2| - |G_1 \cap G_2|} \quad (4)$$

$$dist_5(G_1, G_2) = 1 - \frac{|V_1 \cap V_2|}{|V_1| + |V_2| - |V_1 \cap V_2|} \quad (5)$$

where:
$|V_a|$ - number of nodes in graph G_a.

The second of the new metrics ($dist_5$) takes into account number of nodes rather than the size of graph. It might be considered that such approach allows to find how many common subjects are raised in any two compared documents, especially giving the way the graph is being build, as the nodes are mostly constituted by nouns.

V. Tests

The analysed methods are tested using typical automated document classification scenario. It is based on assumption that having examples of documents from different classes it should be possible to automatically assign correct classes to previously unseen documents, as the distance between similar documents (that is from similar classes) should be generally minimal.

In presented scenario, $k - NN$ (k-Nearest Neighbors) classification algorithm is used. Each of the tested document collections is randomly split into two subsets: *training* and *test*. Next, for each document in *test* set, k nearest documents from *training* set are found. After that, a voting is performed among them. The most common occurring category is assigned to the *test* document.

To test usefulness of the new method for document classification problem, the following document collections (both English and Polish) have been chosen:

- *wiadomosci24.pl* - containing 1500 short articles from one of the leading internet news services in Poland. Each document is tagged with 1 to 5 out of 50 tags, such as: *Gdansk, Festival, Politics, Warsaw, Money, Culture, ...*.
- *Rzeczpospolita*[7] - a 800 documents subset of randomly chosen articles published in 2002 in one of major Polish newspapers. They are tagged with one of eight tags: *World, Culture, Law, Publicism-Commentary, Sport, Poland, Economy, Plus Minus*.
- *Reuters*[8] - a subset of 1800 articles randomly chosen from the *famous* Reuters-21578 "Lewis Split" data set. Each document is tagged with at least one out of 32 categories, such as: *wheat, trade, acq, ship, money-fx, etc.*
- *PDDP K-series*[9] [1] - a subset of 800 randomly chosen (out of 2340) *Yahoo!* articles originally extracted by Daniel Boley and used by various researchers for testing document classification performance (including Schenker). Each of them is tagged with 1 of 20 classes such as: *politics, tech, entertainment, business, etc.*

Benchmarks

Quality of results is typically measured using *precision* and *recall*. Instead of presenting both of these numbers, it is a common practice (used also in this paper) to present their harmonic mean also known as *F-measure*:

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (6)$$

As typically there are more than two classes in total being categorized, two approaches might be used for calculating the average results. The first one is calculating the precision and recall for each category and taking an average. It's called *macro-averaging*. The second one (*micro-averaging*) gives equal weight to every document (rather than category). In such case, precision and recall numbers are calculated for each document and averaged.

Results

For each document collection, considered here as a separate test case, documents are randomly split into training and test sets. Only documents from the ten most frequently occurring tags are selected.

[7] http://www.cs.put.poznan.pl/dweiss/rzeczpospolita
[8] http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html
[9] http://www-users.cs.umn.edu/~boley/ftp/PDDPdata/

The method parameters (such as maximum number of nodes in a graph, maximum number of features in bag-of-words approach, minimum word count to be selected as a feature, k-number, etc.) were optimized to achieve best results for each of the methods. In other words - multiple tests with varying parameters were run and the best case results are presented.

The tables and figures presented below show results for test/training set ratio of 0.5, so there is equal number of test and training documents. Tests were performed also for other ratios (in $0.1 - 0.9$ range) and relative results were similar (i.e. shallow semantic analysis based approach provided better results than other tested methods).

For *bag-of-words* method both *Jaccard* and *cosine* similarity (with *tf-idf*) were tested. Results for the better performing measure are presented. This is also the case for both graph methods - only the best performing metrics are presented in the results, even if all were tested.

The Schenker approach was implemented according to the description found in [23], [22], [21].

TABLE II
RESULTS - RZECZPOSPOLITA

		MICRO AVG. F1	MACRO AVG. F1
BAG-OF-WORDS	JACCARD	0.68	0.65
SCHENKER	$dist_3$	0.73	0.70
	$dist_4$	0.70	0.66
RANDOM		0.30	0.16
SHALLOW SEMANTIC ANALYSIS	$dist_4$	0.74	0.69
	$dist_5$	0.72	0.69

Polish Texts - Rzeczpospolita: Documents in this collection were tagged with the lowest number of tags in comparison to other sets. In effect, the categories assigned were more general. The only geographic tags were *world* and *Poland*.

TABLE III
RESULTS - WIADOMOSCI24.PL

		MICRO AVG. F1	MACRO AVG. F1
BAG-OF-WORDS	JACCARD	0.46	0.47
SCHENKER	$dist_3$	0.47	0.47
	$dist_4$	0.46	0.49
RANDOM		0.17	0.17
SHALLOW SEMANTIC ANALYSIS	$dist_4$	0.50	0.51
	$dist_5$	0.52	0.50

Polish Texts - wiadomosci24.pl: Lower (in comparison to other test cases) absolute results for *wiadomosci24.pl* articles might be related to the way the tags were assigned, as there were many disambiguations found. For example - document was tagged as *Sport* while in fact it was more related to other tags available for the set, such as: *Lodz, Football*.

English Texts - Reuters-21578: Reuters results provide highest absolute values. There are many reasons for this. One of them is the fact that the original collection was analyzed and "cleaned up" - so it might be expected that many incorrectly assigned tags were removed and some of the disambiguations fixed. Also, it was used as a training set for Stanford part-of-speech tagger, thus it should be most correctly tagged.

TABLE IV
RESULTS - REUTERS 21578

		MICRO AVG. F1	MACRO AVG. F1
BAG-OF-WORDS	JACCARD	0.87	0.73
SCHENKER	$dist_3$	0.87	0.71
	$dist_4$	0.87	0.72
RANDOM		0.46	0.13
SHALLOW SEMANTIC ANALYSIS	$dist_4$	0.89	0.78
	$dist_5$	0.88	0.75

TABLE V
RESULTS - PDDP K-SERIES

		MICRO AVG. F1	MACRO AVG. F1
BAG-OF-WORDS	JACCARD	0.81	0.78
SCHENKER	$dist_3$	0.83	0.79
	$dist_4$	0.84	0.80
RANDOM		0.22	0.15
SHALLOW SEMANTIC ANALYSIS	$dist_4$	0.85	0.81
	$dist_5$	0.84	0.81

English Texts - PDDP K-series: PDDP collection was a subject of tests by Schenker *et al.* Thus it was interesting to see how the shallow semantic analysis approach will compare to it. It is worth noting that size of the document collection used here is about twice as large as the collection size used by Schenker [22].

Analysis

For each of the test cases and benchmarks, except one (macro averaged F1 measure for *Rzeczpospolita* articles), shallow semantic analysis method presents results better than both the vector space and Schenker approaches. The improvement is most notable for *wiadomosci24.pl* and *Reuters* articles. The probable cause of this is the fact that *Rzeczpospolita* and *PDDP K-series* collections have too general tags assigned. For example - in case of *PDDP K-series* there is a single tag related to *business*, while for *Reuters* document collections there are: *acq*, *trade*, *money-fx* and others. It is similar for *Rzeczpospolita* vs. *wiadomosci24.pl* collections. The first one has only 8 possible tags, while the latter has 50 classes.

VI. CONCLUSIONS

A new method for building graph representation of text is presented. With a help of the part-of-speech tagger and semantic dictionary it is performing a shallow semantic analysis of input document producing a model reminiscent of semantic net. Preliminary experiments have been performed for both Polish and English documents to check its practical usability.

The obtained results show that the proposed approach has a slight edge over Schenker and bag-of-words approaches. This suggests that the new method is able to produce graphs better representing the latent meaning of document, even if it effectively uses only some of the terms from the original text representation.

It is important to note that the proposed algorithm, presented in section IV, is a current variant rather than a final version. Depending on the features of target documents and regimes, it might be accordingly modified and extended.

As for the metrics, it has been found that $dist_3$ produces the best results for Schenker method (with a close call for $dist_4$). For shallow semantic analysis approach, $dist_4$ gives the strongest performance. While the new proposed metrics work generally well for both graph methods, the $dist_1$, $dist_2$ and $dist_3$ produce poor results for the shallow semantic analysis.

The presented method is a subject of intensive research. The current focus is directed on two aspects. The first one is leveraging the *Anaphora Resolution* in the method, which should effect in even better classification results. The second is extracting weighted features from the graph (e.g. as presented by Markov *et al.* [14]) and using them with more sophisticated classifier (such as SVM).

REFERENCES

[1] D. Boley. Principal direction divisive partitioning. *Data Mining and Knowledge Discovery*, 2:325–344, 1997.

[2] L. Borin and K. Prütz. Through a glass darkly: Part-of-speech distribution in original and translated text. *Language and Computers*, 15, 2000.

[3] M.C. Caselli, E. Bates, P. Casadio, J. Fendon, L. Fenson, L. Sanderl, and J. Weir. A cross-linguistic study of early lexical development. *Cognitive Development*, 1995.

[4] Tommy W. S. Chow, Haijun Zhang, and M. K. M. Rahman. A new document representation using term frequency and vectorized graph connectionists with application to document retrieval. *Expert Syst. Appl.*, 36(10):12023–12035, 2009.

[5] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[6] Peter J. Dickinson, Horst Bunke, Arek Dadej, and Miro Kraetzl. On graphs with unique node labels. In Hancock and Vento [9], pages 13–23.

[7] Ch. Fellbaum, editor. *WordNet - An Electronic Lexical Database*. The MIT Press, 1998.

[8] Peter Geibel, Ulf Krumnack, Olga Pustylnikov, Alexander Mehler, Helmar Gust, and Kai-Uwe KÃijhnberger. Structure-sensitive learning of text types. In Mehmet A. Orgun and John Thornton, editors, *Australian Conference on Artificial Intelligence*, Lecture Notes in Computer Science, pages 642–646. Springer, 2007.

[9] Edwin R. Hancock and Mario Vento, editors. *Graph Based Representations in Pattern Recognition, 4th IAPR International Workshop, GbRPR 2003, York, UK, June 30 - July 2, 2003, Proceedings*, volume 2726 of *Lecture Notes in Computer Science*. Springer, 2003.

[10] Chuntao Jiang, Frans Coenen, Robert Sanderson, and Michele Zito. Text classification using graph mining-based feature extraction. In *SGAI Conf.*, pages 21–34, 2009.

[11] Jurafsky, Daniel, Martin, and H. James. *Speech and Language Processing (2nd Edition) (Prentice Hall Series in Artificial Intelligence)*. Prentice Hall, 2 edition, 2008.

[12] G. Leech, P. Rayson, and A. Wilson. *Word Frequencies in Written and Spoken English: based on the British National Corpus*. 2001.

[13] C. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition, 2008.

[14] A. Markov, M. Last, and A. Kandel. The hybrid representation model for web document classification. *Int. J. Intell. Syst.*, 23(6):654–679, 2008.

[15] Rada Mihalcea and Paul Tarau. TextRank: Bringing order into texts. In *Proceedings of EMNLP-04 and the 2004 Conference on Empirical Methods in Natural Language Processing*, 2004.

[16] E. Newport, H. Gleitman, and L. Gleitman. Mother i'd rather do it myself: Some effects and non-effects of maternal speech style. *Talking to Children: Language Input and Acquisition*, 1977.

[17] M. Piasecki. Polish tagger TaKIPI: Rule based construction and optimisation. *Task Quarterly*, 11(1–2):151–167, 2007.

[18] Maciej Piasecki, StanisÅĆaw Szpakowicz, and Bartosz Broda. *A Wordnet from the ground up*. Oficyna wydawnicza Politechniki WrocÅĆawskiej, WrocÅĆaw, Polska, 2009.

[19] S. Pinker, L. R. Gleitman, and M. Liberman (Eds.). *An Invitation to Cognitive Science, Vol. 1 Language*. The MIT Press, 2 edition, 1995.

[20] M. F. Porter. An algorithm for suffix stripping. *Program*, 1980.

[21] A. Schenker, H. Bunke, M. Last, and A. Kandel. *Graph-Theoretic Techniques for Web Content Mining (Machine Perception and Artificial Intelligence) (Series in Machine Perception and Artificial Intelligence)*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2005.

[22] A. Schenker, M. Last, H. Bunke, and A. Kandel. Classification of web documents using a graph model. In *Proceedings of the Seventh International Conference on Document Ana lysis and Recognition*, 2003.

[23] A. Schenker, M. Last, H. Bunke, and A. Kandel. A comparison of two novel algorithms for clustering web documents. In *Second International Workshop on Web Document Analysis*, Edinburgh, UK, 2003.

[24] J. Snedeker and L. Gleitman. *Weaving a Lexicon*, chapter Why it is hard to label our concepts, pages 255–293. Bradford Book, 2004.

[25] A. Strehl, J. Ghosh, and R. Mooney. Impact of similarity measures on web-page clustering. In *AAAI-2000: Workshop of Artifical Intelligence for Web Search*, 2000.

[26] K. Toutanova, D. Klein, C. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of HLT-NAACL 2003*, 2003.

[27] K. Toutanova and C. D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 2000.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 461–467

ISBN 978-83-60810-27-9
ISSN 1896-7094

PerGram: A TRALE Implementation of an HPSG Fragment of Persian

Stefan Müller
German Grammar Group
Freie Universität Berlin
14195 Berlin
Stefan.Mueller@fu-berlin.de

Masood Ghayoomi
German Grammar Group
Freie Universität Berlin
14195 Berlin
Masood.Ghayoomi@fu-berlin.de

***Abstract*—In this paper, we discuss an HPSG grammar of Persian (PerGram) that is implemented in the TRALE system. We describe some of the phenomena which are currently covered. While working on the grammar, we developed a test suite with positive and negative examples from the linguistic literature. To be able to test the coverage of the grammar with respect to naturally occurring sentences, we use a subcorpus of a big corpus of Persian.**

I. Introduction

IN THE past two decades, Head-driven Phrase Structure Grammar (HPSG, [1], [2]) was used successfully to formalize the phonology, morphology, syntax, semantics, and information structure of various languages. Apart from the role HPSG plays in theoretical linguistics, there is also an active community developing large scale implemented grammar fragments that can be used for parsing and generation. While there are large scale grammar fragments available for major languages like English [3] and German [4], [5], [6], [7], other languages such as Persian are understudied.

In this paper, we describe a fraction of the phenomena that are covered in an implemented fragment of Persian (PerGram). The focus is on the implementation of the grammar.

The structure of the paper is as follows: we will say a few words about Persian in general and also briefly describe some of its specific syntactic properties in Section II. Since the grammar is implemented in the TRALE system, we will introduce the general features of TRALE in Section III. In Section IV, some of the phenomena implemented in PerGram are discussed. Section V describes the test suites we are using. In Section VI, we summarize the paper.

II. Persian

Persian is a member of the Indo-European language family and it has many features in common with the other languages of this family in terms of phonology, morphology, syntax, and lexicon. Persian uses a modified version of the Arabic script and is written right to left. However, the two languages differ from one another in many respects. Persian belongs to the subject-drop languages with an SOV constituent order in unmarked structures. The constituent order is relatively free. Verbs are inflected for tense and aspect, and they agree with the subject in person and number. The particle /rā/ is used as an object marker. The language does not make use of gender [8]. Persian has simple and compound prepositions [9], [10]. There exists a closed list of simplex verbs and an open list of compound verbs.

III. TRALE

The TRALE system [11], [12], [13] is an extension of the Attribute Logic Engine (ALE) [14], [15]. TRALE has a parser and a generator and can be used to implement theoretical HPSG proposals rather directly. The only specification that is needed in addition to the constraints that are known from theoretical papers is a phrase structure backbone that has to be defined to guide the parsing process. The system is run with a graphical user interface called Grale that can be used to visualize lexical items, lexical rules, types, and macros as an Attribute Value Matrix (AVM) including relational constraints that might be attached to the respective objects. The GUI does unfilling, that is, features and types that are not more specific than the types in the type definition (the signature, see below) are not displayed. This is a very important feature for debugging large scale grammars. The most recent extension of TRALE features a Java-based graphical debugger, which makes it possible to debug the unification operations stepwise and to debug relational constraints during parsing [16].

To declare a grammar in TRALE, a *signature* file is required which defines the types and the features that are appropriate for them. In addition, a TRALE grammar has one or several files containing the definitions of lexical items, lexical rules, phrase structure rules, relational constraints, principles, etc.

To make the development of large scale grammars feasible, macros can be used in the definition of linguistic objects. Like types, macros can be organized in hierarchies. While the type hierarchy is stated explicitly in the signature file, the macro hierarchy is defined rather implicitly by calling macros in the definition of other macros. In contrast to types, macros can be parametrized. (1) shows an example of a parametrized macro in the lexical entry for /man/ ('I'):

(1) man ~~> @pers_pronoun(first, sg, human).

The macro for personal pronouns takes three parameters: one for person, one for number, and one for the semantic type of the pronoun. Macros are very useful for the development of the lexicon since they hide the complexity of the grammar and therefore make it possible for inexperienced users to write lexical entries.

While HPSG in general deals with linguistic phenomena belonging to all main dimensions of grammar (phonology, morphology, syntax, semantics, pragmatics), the implementation currently considers morphology, syntax, and semantics. The syntactic analysis uses the feature geometry and makes the basic assumptions that were worked out in [7] for German. See [17] for the details on Persian. The semantic analysis is based on Minimal Recursion Semantics (MRS) introduced by Copestake et al. [18]. MRSes can be displayed using 'utool' [19]. 'utool' also provides a scope resolver for MRS.

The TRALE system supports Unicode and it is therefore possible to parse Persian text that is written in the Arabic script. However, since in written Persian short vowels are omitted, a lot of information such as Ezafe, which we will describe in the following section, is left implicit. In order to be able to formalize and test constraints regarding the distribution of linguistically important short vowels, we transcribe Persian words with Latin characters. We already have a version of the lexicon in Arabic script and plan to extend the grammar in a way that makes it possible to use it with or without the transcription.

The implemented fragment of Persian shares a common core with fragments of German, Mandarin Chinese [20], Danish [21], and Maltese [22]. For more information on this core and for downloading the grammars see http://hpsg.fu-berlin.de/Projects/core.html.

IV. The Covered Linguistic Phenomena

A. Principles and Schemata

The grammar uses several immediate dominance schemata and principles that are similar to the ones that were originally suggested by P&S94 [2]. The Persian grammar uses a Head-Adjunct-Schema, a Head-Complement-Schema, a Head-Specifier-Schema, and a Head-Filler-Schema. In addition to these schemata, a Head-Cluster-Schema is used for the formation of complex predicates (see [23], [24], [17] for analyses of predicate complexes in German and Persian). In addition to these more general ID schemata, the grammar uses language specific schemata for the combination of nominal elements with their possessor and for noun compounding.

Principles that hold for all of the mentioned ID schemata are factored out of the schemata and are represented as constraints on an appropriate type, for instance *phrase* or *headed-phrase*. Examples of such principles are the 'head feature principle', the 'semantics principle', the 'specifier principle', and the 'nonlocal feature principle'.

B. Morphological Rules

The grammar contains morphological rules both for derivation and inflection. The morphological rules are modeled as lexical rules. For instance, for inflectional morphology, we use lexical rules that map roots or stems to fully inflected words.

The following lexical rule (LR) is responsible for noun inflection. It maps a nominal stem onto a word with exactly the same syntactic and semantic properties.

$$\begin{bmatrix} \textsc{phon}\ \boxed{1} \\ \textsc{ss}\ \boxed{2}\begin{bmatrix}\textsc{loc} \mid \textsc{cat} \mid \textsc{head}\ \textit{noun}\end{bmatrix} \\ \textsc{affix}\ \begin{bmatrix} \textsc{phon} & \boxed{3} \\ \textsc{num} & \boxed{4} \\ \textsc{sort} & \boxed{5} \\ \textit{noun-i-affix} & \end{bmatrix} \\ \textit{stem} \end{bmatrix} \mapsto \begin{bmatrix} \textsc{phon}\ \boxed{1} \oplus \boxed{3} \\ \textsc{ss}\ \boxed{2}\begin{bmatrix}\textsc{loc} \mid \textsc{cont} \mid \textsc{ind}\ \begin{bmatrix}\textsc{num}\ \boxed{4} \\ \textsc{sort}\ \boxed{5}\end{bmatrix}\end{bmatrix} \\ \textit{word} \end{bmatrix}$$

The input of the rule has a special feature the value of which contains the information about the affix. For nominal inflection, the affix has to have the type *noun-i-affix*. There is a constraint on this type that disjunctively specifies the inflectional paradigm.

$$\textit{noun-i-affix} \Rightarrow \begin{bmatrix} \textsc{phon} & \langle\rangle \\ \textsc{num} & \textit{sg} \end{bmatrix} \vee \begin{bmatrix} \textsc{phon} & \langle \textit{ān} \rangle \\ \textsc{num} & \textit{pl} \\ \textsc{sort} & \textit{human} \end{bmatrix} \vee \begin{bmatrix} \textsc{phon} & \langle \textit{hā} \rangle \\ \textsc{num} & \textit{pl} \end{bmatrix}$$

The respective PHON values provide information about the phonological contribution of the stem and the affix. These values are lists of phonemes and they are concatenated in the output of the lexical rule. $\oplus$ stands for the *append* relation. In addition to the concatenation of the PHON values, the values of NUM and SORT of the output of the rule are instantiated with the features provided by the affix. SORT is a feature that is used to enforce selectional restrictions. The values are based on a semantic ontology which will be described in the subsection IV-E.

In the lexical rule given above, the SYNSEM value of the input is identified with the SYNSEM value of the output. This is different in LRs for derivational morphology. For instance, in the LR that derives adjectives from verbs by appending the suffix *-i* ('-able'), the part-of-speech and the valence specification changes. In addition to these syntactic changes, the semantic contribution of the verb is embedded under a modal operator.

Apart from this derivational LR, we have lexical rules for participle to adjective conversion and for agentive nominalizations. All these morphological rules interact properly with the formation of complex predicates. See [17] for details.

C. Ezafe

The so-called Ezafe is a short vowel /e/ which functions to link the elements of a noun phrase (see for instance [25]). Ezafe appears on: a noun before another noun (attributive); a noun before an adjective; a noun before a possessor (noun or pronoun); an adjective before another adjective; a pronoun before an adjective; the first names before the last names; a combination of the above [26]. Ezafe is realized as /e/ after consonants and /i/ and as /ye/ after vowels other than /i/. Ezafe does not appear on a bare noun or adjective and its appearance indicates that the end of the syntactic phrase is not reached.

We defined an LR which adds the Ezafe at the end of a word. To distinguish Ezafe-marked words from unmarked ones, we employ a binary valued feature EZAFE. The lexical rule applies to words that have the value '−' and licenses words with the value '+'.

EZAFE is an edge feature, that is, a complex phrase is Ezafe-marked if an Ezafe is present at its right periphery. (This is similar to the possessive *'s* in English.) The Ezafe marking of phrases is taken care of by the following constraint on phrases:

$$\textit{phrase} \Rightarrow \begin{bmatrix} \textsc{ss} \mid \textsc{ezafe}\ \boxed{1} \\ \textsc{dtrs}\ \boxed{2} \end{bmatrix} \wedge \textit{last}(\boxed{2}, \left[\textsc{ss} \mid \textsc{ezafe}\ \boxed{1}\right])$$

The relational constraint *last* succeeds if the second argument ($[\textsc{ss} \mid \textsc{ezafe}\ \boxed{1}]$ in the example above) is the last element of the list that is provided as the first argument. The DTRS list is a list of daughters that is ordered according to the surface order of constituents, so *last* returns the rightmost daughter. The EZAFE value of this daughter is shared with the EZAFE value of the mother. Since there is no reference to the number of daughters in the constraint above, it applies to unary and binary branching phrases alike. Currently we only have unary and binary branching rules in the grammar, but of course the constraint would apply to structures with three or more daughters as well.

The distribution of the Ezafe is constrained by implicational statements like the following:

$$\begin{bmatrix} \textsc{head-dtr} \mid \textsc{ss} \mid \textsc{loc} \mid \textsc{cat} \mid \textsc{head}\ \ \textit{noun} \\ \textit{head-argument-phrase} \end{bmatrix} \Rightarrow \left[\textsc{head-dtr} \mid \textsc{ss} \mid \textsc{ezafe}\ -\right]$$

This constraint applies to combinations of nouns with their arguments. Since prepositional arguments have to be realized outside of the Ezafe domain, the head daughter is required to have the EZAFE value '−'. The schema that is used for the combination of a noun with a possessor requires the nominal constituent to have the EZAFE value '+'.

D. Negation

A verb or an auxiliary can be negated by attaching the prefix /na-/. This is implemented by a lexical rule that adds the phonological material and embeds the content of the verb under the negation relation. The syntactic properties of the verb are not affected by the negation and are carried over from the input of the rule to the output. Persian differs from languages like German in that it is impossible to negate a non-finite verb that is embedded under a modal. This is captured by a constraint that requires that the input to the lexical rule is a finite verb.

The LR applies to auxiliaries, simplex verbs, and the verbal element of complex predicates. In the latter case, the negation scopes over the whole complex predicate even though the negation attaches to the verb before the other part of complex predicate is combined with the negated verb. For details see [17].

E. Nouns

Several kinds of nouns are modeled. We implemented common nouns with and without arguments. The arguments are always optional and we have subclasses for nouns that take CPs and for nouns that take PPs as complement. In addition to common nouns, the grammar contains lexical items for proper nouns. Apart from common nouns and proper nouns, we have lexical entries for nouns that play a special role in complex predicate formation (process nouns and verbal nouns). See [17] for details on these nouns.

All non-expletive linguistic objects are classified with respect to an ontology. The ontology contains types like *human*, *agentive*, *substance*, and *geo-location*. This ontology is an extended version of the ontology that was developed in Verb*mobil* [27]. It can be used to specify sortal restrictions of governing verbs with respect to their arguments. Apart from this, it can be used to enforce certain syntactic constraints. For instance, one allomorph of the plural affix /-ān/ is only used with nouns that refer to humans.

F. Verbs

In Persian, there are two classes of verbs: a closed list of simplex verbs; and an open list of compound verbs. The latter group is composed of a preverbal and a verbal element. The verbal elements which belong to a subclass of the simplex verbs are called 'light verbs' and the whole predicate formation process is called 'light verb construction'. The implementation of this phenomenon in our grammar is based on [17] and is not discussed here.

Currently, the grammar has lexical entries for the following kinds of verbs: mono-valent verbs with one NP argument, bi-valent verbs with two NP arguments, bi-valent verbs with an NP and a PP argument, ditransitive verbs with two NPs and a PP, verbs with an NP and a clausal argument, copula verbs, modal verbs, mono-valent unaccusative verbs, and several types of light verbs.

As said, Persian is a language with a relatively free constituent order. This is captured by allowing the combination of an arbitrary element of the SUBCAT list with a head. For instance the head may be combined with the subject and then with the object or the other way round. Languages with strict constituent order like English do not allow this but require the combination of heads with the arguments in order of their obliqueness. See [28] for further discussion of this difference.

We follow [2] in assuming that there is a special representation of valence information, nowadays called Argument Structure (ARG-ST). For all heads there is a mapping from the ARG-ST list to other valence features like COMPS and SPR. For SVO languages like English and Danish the least oblique argument of a verb is mapped to SPR and all the other arguments are mapped to COMPS[1] [29]. The verb forms a VP together with the arguments that are selected via COMPS. This VP is combined with the subject to form a sentence. In contrast, in languages like German and Persian, all arguments of finite verbs are mapped to COMPS. This makes it possible to account for orders like OSV, in which the object and the verb are not adjacent.

While the arguments can be realized in any order with respect to each other, the order with respect to their head

[1] For historical reasons COMPS is still called SUBCAT in the implementation.

is rather fixed: Persian is an SOV language, that is, the verb follows its arguments.[2] On the other hand, Persian has prepositions, that is, the adposition precedes its complement. To capture this, we assume two phrase structure rules that are instances of the general Head-Argument-Schema: one head initial and another one head final. We use a head feature INITIAL, which has the value '+' for heads that are serialized initially and '−' for heads that follow their arguments.

All tense and aspect forms of the verbal paradigm are covered. The progressive and subjunctive marking is done by inflectional lexical rules. The auxiliaries that are used for periphrastic forms are described in [17] and the description will not be repeated here. Three types of complex predicate formation are also discussed in [17] and covered in the grammar.

G. Agreement

Subject verb agreement is handled by the lexical rule that licenses finite verbs. The rule requires that the least oblique NP in the ARG-ST list that bears structural case shares its person and number values with the person and number features of the inflectional affix. This treatment of subject verb agreement is also used for the grammars of German, Maltese, and Hindi. For a similar analysis of agreement in Spanish see [30]. In comparison to the other languages, Persian allows for an additional case: Plural NPs referring to non-agentive entities may also occur with verbs inflected for third person singular.

H. Prepositions

As mentioned above, prepositions differ from verbs in governing their complement to the right.[3] This is enforced by their INITIAL value, which is '+'. The fragment currently contains prepositions that form PPs that can be used as arguments, prepositions that can be used as modifiers, and a preposition similar to the English preposition *by* that can be used in passive constructions.

I. Clitics

Pronominal clitics can be used as possessives (2). As in noun phrases with a full NP as possessor phrase, the clitic is the rightmost element in the NP. Therefore there are two possible hosts for clitic attachment: a noun as in (2a) or an adjective as in (2b):

(2) a. ketāb=aš
book=3SG
'his/her book'

b. ketāb-e jadid=aš
book-EZ new=3SG
his/her new book

Clitics can also fill the slot of the direct object of a verb. In this case, the clitic attaches to the verb as in (3a) or to the preverbal element in a complex predicate construction (3b,c), or to the future auxiliary (3d):

(3) a. did-am=aš.
saw-1SG=3SG
'I saw him/her.'

b. bāz=aš kard-am.
open=3SG did-1SG
'I opened it.'

c. dust=aš dār-am.
friend=3SG have-1SG
'I love him/her.'

d. dust xāh-am=aš dāšt.
friend FUT-1SG=3SG have.SG
'I will love him/her.'

Currently these clitics are treated as postlexical clitics, that is, clitics are treated in the syntactic component. For clitics that fill the object slot of verbs, there is a special grammar rule in the phrase structure backbone used by TRALE. This rule is necessary since the order of clitic and verb is different from the usual order. Apart from these order differences, this grammar rule is an instance of the general Head-Argument-Schema. The noun possessor construction has the same structure for possessors that are realized as clitics and for possessors realized as full NPs. The only difference is the impossibility of the Ezafe when the possessor is realized as a clitic.

However, Samvelian argued for a treatment of clitics as phrasal affixes [25], and the grammar will be adapted in order to cover lexical and morphological idiosyncrasies.

J. Determiners

Persian makes use of prenominal and postnominal determiners: demonstratives like /in/ ('this') and /ān/ ('that') and the indefinite element /yek/ ('a/an') are realized prenominally and the indefinite marker /*-i*/ occurs postnominally.

The prenominal determiners are treated as dependents of the noun and are selected via the SPR feature as suggested by [2]. To establish the semantic relation between the quantifier and the noun, the determiner is able to access the variable of the nominal projection via the SPEC feature.

The indefinite marker /*-i*/ is currently not covered but the implementation of a lexical rule that has the desired phonological, syntactic, and semantic effects is straight forward.

K. Direct Object Marker

The particle /rā/ is used as a marker of a noun in object position. Since object marking is optional, we treat /rā/ as an adjunct. The following is the lexical entry of the particle /rā/:

$$
\left[\begin{array}{ll}
\text{PHON} & \langle \textit{rā} \rangle \\
\text{SS}\,|\,\text{L} & \left[\begin{array}{ll}
\text{CAT}\,|\,\text{HEAD}\,|\,\text{MOD}\,|\,\text{L} & \left[\begin{array}{ll}
\text{CAT} & \left[\begin{array}{ll}
\text{HEAD} & \left[\begin{array}{l} \text{CASE}\,|\,\text{MORPH-CASE}\ \textit{acc} \\ \textit{noun} \end{array}\right] \\
\text{SPR} & \langle\rangle
\end{array}\right] \\
\text{CONT} & \boxed{1} \left[\begin{array}{ll} \text{LTOP} & \boxed{2} \\ \text{IND} & \boxed{3} \end{array}\right]
\end{array}\right] \\
\text{CONT} & \boxed{1}
\end{array}\right] \\
\text{RELS} & \left\langle \left[\begin{array}{ll} \text{LTOP} & \boxed{2} \\ \text{ARG1} & \boxed{3} \\ \textit{specific} & \end{array}\right] \right\rangle
\end{array}\right]
$$

The modified linguistic object is required to have accusative case. This ensures that /*rā*/ attaches to an object. In addition, it is required to have an empty SPR list, i.e. it has to be an NP. We assume that the list of relations that is contributed by signs is not part of CONT, but rather represented at the top level of

[2] We are aware of the fact that arguments may be realized postverbally. Currently, only the postverbal realization of clausal arguments is implemented. We leave the other serialization options to further research.

[3] This shows that the assumption of a headedness parameter that has the same value for verbs and prepositions makes the wrong predictions for Persian. Therefore such a parameter should not be part of an innately specified UG, if there is such a thing at all. See [31] on a detailed discussion of language acquisition including a discussion of Principle & Parameter approaches.

the feature description. This yields a more restrictive theory as far as the locality of selection is concerned [32]. Since RELS is not part of CONT, the CONT values of the modifier and the modified element can be shared. */rā/* is treated as an intersective modifier, and hence the LTOP of the modified noun is identified with the LTOP of the specificity relation contributed by */rā/*. Of course, the argument of the specificity relation is identical to the referential index of the NP ($\boxed{3}$).

L. Complementizers for Subordinated Clauses and Relative Clauses

Embedded clauses are introduced by the complementizer */ke/* ('that'). We treat the complementizer as a head that selects a finite clause. In addition to this complementizer, we have another lexical entry for */ke/* which is used in relative clauses [33]. The grammar covers relative clauses with and without resumptive pronouns. Free relatives will be implemented in the near future. The analysis is described in detail in Taghvaipour's thesis about relative clauses.

M. Pro Drop

Initially pro drop was handled by a unary branching rule that discharges the subject from a valence representation after all other arguments of a finite verb have been saturated. While this rule works correctly, it is inefficient since it contributes edges to the bottom up chart parser even if an overt subject is present in a clause. We, therefore, implemented an underspecification analysis that originally was used for imperatives in BerliGram, an implementation of a grammar of German [7]. The grammars use a valence representation that comes with an additional feature REALIZED [34], [35]. Arguments that are not realized have the value '$-$' and once they are realized, the value is changed to '+' in the valence representation at the mother node. This representation can be used to represent optional arguments by underspecification (see [36] for a different solution using a binary feature): For the subject of finite verbs, the REALIZED value can remain unspecified. The unspecified value is compatible with '+' in the pro drop case and '$-$' in the case when a verbal projection is combined with an overt subject. During a parse of the 165 test sentences from the PerGram Test Suite (see Section V), the grammar version with unspecified REALIZED value licensed 12.7 % less passive edges in comparison to the one that uses the unary branching rule. This resulted in a reduction of parse times of 30.6 % in average.

N. Coordination

The grammar handles symmetric coordinations. The coordination of two or more NPs with */va/* ('and') results in a plural NP, so that the agreement facts are captured correctly. The analysis of coordination is basically the one suggested by [2], that is, the CAT and NONLOCAL values of the conjuncts are shared. However, there is a slight complication: since we use a non-cancellation approach to valence, examples like the one in (4) are problematic.

(4) Ali [[mard rā did] va [xandid]].
Ali man RA saw.3SG and laughed.3SG
'Ali saw the man and laughed.'

In (4), a VP with a transitive verb and one with an intransitive verb is coordinated. The valence representations of the respective VPs is shown in (5):

(5) a. mard rā did: SUBCAT ⟨ NP, ~~NP~~ ⟩
b. xandid: SUBCAT ⟨ NP ⟩

The valence representation of the VP with a transitive verb contains an NP that is marked as realized. Meurers [34] called realized arguments *spirit*. In comparison, the VP with the intransitive verb does not contain such an NP. The consequence is that the SUBCAT values of the conjuncts cannot be unified since the lengths of the lists are different. This problem is solved in the implemented grammar by using a relational constraint that returns all unrealized elements in a valence list. This constraint is applied to both conjuncts and the respective result lists are unified and represented at the mother node. As a result, the valence representation of [[mard rā did] va [xandid]] is ⟨ NP ⟩. The spirit NP (~~NP~~) is not represented at the mother node. The subject NPs of */didan/* ('see') and */xandan/* ('laugh') are unified and hence it is explained why *Ali* fills the respective slots of both verbs.

This analysis captures a lot of complicated coordination phenomena like Across the Board Extraction in unbounded dependencies (questions and relative clauses) and also interacts nicely with resumptive pronouns in relative clauses. However, there is one problem left: the analysis does not extend to German fronting data and case assignment for which it was introduced originally. Meurers [34] and Przepiórkowski [35] suggested representing the saturated complements at the VP level. Auxiliary verbs attract the arguments of the verb they embed. Case is assigned to arguments that are not raised by higher verbs. In the analysis of the sentences in (6), the auxiliary attracts the arguments of *gelesen* ('read') and assigns nominative to the subject and accusative to the object.

(6) a. [Einen Aufsatz gelesen] hat er nicht.
an essay.ACC read has he.NOM not
'He did not read an essay.'
b. [Ein Aufsatz gelesen] wurde nicht.
an essay.NOM read was not
'An essay was not read.'

The important point is that the case is not determined locally in the fronted VP, but assigned by the finite verb. In order to assign case, the finite verb has to raise the realized argument (the spirit) and hence it has to be accessible at the VP node.

With this background, the problem of the coordination analysis is obvious: We can coordinate two VPs and front them. According to the coordination analysis sketched above, realized arguments are not contained in the valence lists of the mother nodes of coordinated structures. Since these spirits are needed for case assignment, we have conflicting demands: coordination requires VPs with verbs of different arity to be syntactically parallel and case assignment (in German) requires all arguments to be present at the VP node.

(7) a. [[Einen Aufsatz gelesen] und [einen Report
an essay.ACC read and a report.ACC
geschrieben]] hat er nicht.
written has he.NOM not
'He neither read an essay nor did he write a report.'

b. [[Ein Aufsatz gelesen] und [ein Report
an essay.NOM read and a report.NOM
geschrieben]] wurde nicht.
written was not
'An essay was not read and a report was not written either.'

It remains to be seen if it is possible to develop a consistent analysis of coordination and non-cancellation approaches to valence.

O. Empty Elements

Currently, two types of analyses of unbounded dependency phenomena are entertained in the HPSG framework: one assumes an empty element for the introduction of a nonlocal dependency [2] and the other one introduces nonlocal dependencies lexically [37]. See [38] for an extended discussion. In our implementation we adopt a trace-based approach.

In addition to the empty element that is used in nonlocal dependencies, we also use an empty determiner for the analysis of NPs that do not have an overt determiner. The empty determiner introduces the quantifier relation that is needed for the interpretation of the NP. The alternative to this treatment would be a unary branching ID schema that discharges the valence requirement represented under SPR and introduces the appropriate semantics. By adopting this solution, one would miss the generalization about determiners. In our approach, the type definitions for overt determiners can be used for the covert determiner as well. No idiosyncratic ID rule is needed. There is just one place in the grammar where it is said that the phonology of a determiner may be empty.

We agree that empty elements should be avoided wherever possible and that they should not be stipulated on a cross-linguistic basis but rather be motivated by evidence from within the language under consideration. That is, a topic morpheme in Japanese should not be seen as evidence for an empty topic head in German, English, and Danish. If empty elements are assumed based on the evidence from the language under consideration, the language acquisition model does not have to assume a rich UG, but is compatible with data-driven approaches like the one that is entertained by Tomasello [39].

As is known from research on formal grammars [40], phrase structure grammars with empty elements can be converted into grammars without empty elements. This result does not transfer directly to grammars with typed feature structures, but most of the grammars that are currently suggested can be converted into grammars without empty elements by applying the techniques developed for PSGs (See [24], [31] for examples). TRALE does this kind of grammar conversion for the relevant cases automatically and transparently for the user and hence the grammars can be specified in a more compact way. For example, in the case of the empty determiner, a special variant of the Head-Specifier-Schema is created that has no daughter for the determiner. That is, TRALE compiles the grammar into one that has the unary branching rule that was mentioned above. See [41] for further discussion.

V. The Test Suites

During the development of the Persian grammar, we put together two test suites that are used for systematic testing and grammar profiling [42]. The first one contains examples from the linguistic literature that are relevant for the phenomena that are covered by the grammar. In addition, it contains ungrammatical examples that were constructed in the development process in order to rule out overgeneration of the grammar which was detected by systematic testing. This test suite consists of 165 sentences including both positive (132 sentences) and negative (33 sentences) examples.

To be able to test the coverage of the grammar, we randomly selected 130 sentences from a Persian corpus called 'Peykare'. Peykare [44] is a big Persian corpus provided by the University of Tehran and the Higher Council for Informatics of Iran. It contains texts from various data sources, both written and spoken.The part-of-speech of each word is annotated according to the EAGLES guidelines. The sentences were selected randomly in such a way that the balancedness of the original corpus is kept in the subcorpus; as a result, the 130 sentences have the variability of the existing registers in the Peykare corpus. Currently, most of the sentences do not parse because of missing lexical entries. We added lexical items for proper nouns, common nouns, and adjectives to the lexicon, but there are other missing lexical items (verbs, clitic forms of the copula, numerals, adverbs, adjectives derived from nouns) that affect 98 of the 130 sentences.

VI. Summary and Conclusion

In this paper, we briefly described some of the phenomena that are part of an implemented HPSG grammar of Persian. A full description of the phenomena and the analyses will be provided in [43]. The grammar covers the core aspects of Persian syntax and morphology and provides semantic representations in the form of MRS. The grammar is evaluated with respect to the example sentences that were collected from the linguistic literature and ungrammatical examples that were constructed during the development process. In addition, we started experiments with naturally occurring data that was selected from the Peykare corpus.

Acknowledgment

We thank Pollet Samvelian, Olivier Bonami, and three anonymous reviewers for comments on an earlier version of this paper. Of course all shortcomings of the paper and the implementation are entirely our fault.

The first author thanks Mina Esmaili, Pollet Samvelian, Yasser Shakeri, and Mehran A. Taghvaipour for help with the

Persian data. We thank Emily M. Bender, Felix Bildhauer, Olivier Bonami, Bob Borsley, Karine Megerdoomian, Ivan Sag, Pollet Samvelian, and several anonymous reviewers, for comments on and discussion of the complex predicate paper and/or Persian grammar in general.

This work was supported by a grant from Agence Nationale de la Recherche and Deutsche Forschungsgemeinschaft to the Franco-German project *PER-GRAM: Theory and Implementation of a Head-driven Phrase Structure Grammar for Persian* (DFG MU 2822/3-1).

REFERENCES

[1] C. J. Pollard and I. A. Sag, *Information-Based Syntax and Semantics*. Stanford: CSLI Publications, 1987.

[2] ——, *Head-Driven Phrase Structure Grammar*. University of Chicago Press, 1994.

[3] D. P. Flickinger, A. Copestake, and I. A. Sag, "HPSG analysis of English," in *Verbmobil: Foundations of Speech-to-Speech Translation*, W. Wahlster, Ed. Berlin: Springer Verlag, 2000, pp. 254–263.

[4] S. Müller, "The Babel-System—an HPSG Prolog implementation," in *Proceedings of the Fourth International Conference on the Practical Application of Prolog*, London, 1996, pp. 263–277.

[5] S. Müller and W. Kasper, "HPSG analysis of German," in *Verbmobil: Foundations of Speech-to-Speech Translation*, W. Wahlster, Ed. Berlin: Springer Verlag, 2000, pp. 238–253.

[6] B. Crysmann, "On the efficient implementation of German verb placement in HPSG," in *Proceedings of RANLP 2003*, Borovets, Bulgaria, 2003, pp. 112–116.

[7] S. Müller, *Head-Driven Phrase Structure Grammar: Eine Einführung*, 2nd ed. Tübingen: Stauffenburg Verlag, 2008.

[8] S. Mahootiyan, *Persian*. Routledge, 1997.

[9] Z. A. Chime, "An account for compound preposition in Farsi," in *Proceedings of the COLING/ACL 2006*, 2006, pp. 113–119.

[10] Z. A. Chime and M. Ghayoomi, "Incorporation: Word production of persian prepositions and its application in computational linguistics," in *Proceedings of the 2nd Workshop on the Persian Language and Computer*, 2006, pp. 16–24.

[11] W. D. Meurers, G. Penn, and F. Richter, "A web-based instructional platform for constraint-based grammar formalisms and parsing," in *Proceedings of the Effective Tools and Methodologies for Teaching NLP and CL*, 2002, pp. 18–25.

[12] G. Penn, "Balancing clarity and efficiency in typed feature logic through delaying," in *Proceedings of ACL 2004*, 2004, pp. 239–246.

[13] S. Müller, "The Grammix CD Rom. a software collection for developing typed feature structure grammars," in *Proceedings of the Grammar Engineering Across Frameworks Workshop 2007*, ser. Studies in Computational Linguistics ONLINE, T. H. King and E. M. Bender, Eds. Stanford: CSLI Publications, 2007.

[14] B. Carpenter and G. Penn, "Efficient parsing of compiled typed attribute value logic grammars," in *Recent Advances in Parsing Technology*, H. Bunt and M. Tomita, Eds., no. 1. Dordrecht: Kluwer Academic Publishers, 1996, pp. 145–168.

[15] G. Penn and B. Carpenter, "ALE for speech: a translation prototype," in *Proceedings of the 6th Conference on Speech Communication and Technology (EUROSPEECH)*, G. Gordos, Ed., Budapest, Hungary, 1999.

[16] J. Dellert, K. Evang, and F. Richter, "Kahina, a debugging framework for logic programs and TRALE," 2010, presentation at the HPSG 2010 Conference.

[17] S. Müller, "Persian complex predicates and the limits of inheritance-based analyses," *Journal of Linguistics*, vol. 46, no. 3, To Appear 2010, http://hpsg.fu-berlin.de/~stefan/Pub/persian-cp.html.

[18] A. Copestake, D. Flickinger, C. Pollard, and I. A. Sag, "Minimal recursion semantics: An introduction," *Research on Language and Computation*, vol. 4, no. 3, pp. 281–332, 2006.

[19] A. Koller and S. Thater, "Efficient solving and exploration of scope ambiguities," in *Proceedings of the ACL Interactive Poster and Demonstration Sessions*. Ann Arbor: ACL, 2005, pp. 9–12.

[20] S. Müller and J. Lipenkova, "Serial verb constructions in Mandarin Chinese," in *Proceedings of the 16th International Conference on Head-Driven Phrase Structure Grammar*, S. Müller, Ed. Stanford: CSLI Publications, 2009, pp. 234–254.

[21] B. Ørsnes, "Preposed sentential negation in Danish," in *Proc. of the 16th International Conference on Head-Driven Phrase Structure Grammar*, S. Müller, Ed.Stanford: CSLI Publications, 2009, pp. 255–275.

[22] S. Müller, "Towards an HPSG analysis of Maltese," in *Introducing Maltese Linguistics*, B. Comrie, R. Fabri, B. Hume, M. Mifsud, T. Stolz, and M. Vanhove, Eds. Amsterdam, Philadelphia: John Benjamins Publishing Co., 2009, pp. 83–112.

[23] E. W. Hinrichs and T. Nakazawa, "Linearizing AUXs in German verbal complexes," in *German in Head-Driven Phrase Structure Grammar*, J. Nerbonne, K. Netter, and C. J. Pollard, Eds. Stanford: CSLI Publications, 1994, pp. 11–38.

[24] S. Müller, *Complex Predicates: Verbal Complexes, Resultative Constructions, and Particle Verbs in German*. Stanford: CSLI Publications, 2002.

[25] P. Samvelian, "A (phrasal) affix analysis of the Persian Ezafe," *Journal of Linguistics*, vol. 43, pp. 605–645, 2007.

[26] A. Kahnemuyipour, "Persian ezafe construction revisited: Evidence for modifier phrase," in *Proceedings of the 21st Conference of the Canadian Linguistic Association*, 2000, pp. 173–185.

[27] W. Wahlster, Ed., *Verbmobil: Foundations of Speech-to-Speech Translation*. Berlin: Springer Verlag, 2000.

[28] S. Müller, "Head-Driven Phrase Structure Grammar," in *Syntax – Ein internationales Handbuch zeitgenössischer Forschung*, 2nd ed., A. Alexiadou and T. Kiss, Eds. Berlin: Walter de Gruyter Verlag, in Preparation, http://hpsg.fu-berlin.de/~stefan/Pub/hpsg-hsk.html.

[29] ——, "On predication," in *Proceedings of the 16th International Conference on Head-Driven Phrase Structure Grammar*, S. Müller, Ed. Stanford: CSLI Publications, 2009, pp. 213–233, http://hpsg.fu-berlin.de/~stefan/Pub/predication.html.

[30] C. Vogel and B. Villada, "Spanish psychological predicates," in *Grammatical Interfaces in HPSG*, R. Cann, C. Grover, and P. Miller, Eds. Stanford: CSLI Publications, 2000, pp. 251–266.

[31] S. Müller, *Grammatiktheorie: Von der Transformationsgrammatik zur beschränkungsbasierten Grammatik*. Tübingen: Stauffenburg Verlag, To Appear, http://hpsg.fu-berlin.de/~stefan/Pub/grammatiktheorie.html. English translation in preparation.

[32] M. Sailer, "Local semantics in Head-Driven Phrase Structure Grammar," in *Empirical Issues in Formal Syntax and Semantics*, O. Bonami and P. C. Hofherr, Eds. Online, 2004, vol. 5, pp. 197–214. [Online]. Available: http://www.cssp.cnrs.fr/eiss5/sailer/index_en.html

[33] M. A. Taghvaipour, "Persian relative clauses in Head-driven Phrase Structure Grammar," Ph.D. dissertation, Department of Language and Linguistics, University of Essex, 2005.

[34] W. D. Meurers, "Raising spirits (and assigning them case)," *Groninger Arbeiten zur Germanistischen Linguistik (GAGL)*, vol. 43, pp. 173–226, 1999, http://www.sfs.uni-tuebingen.de/~dm/papers/gagl99.html.

[35] A. Przepiórkowski, "On case assignment and "adjuncts as complements"," in *Lexical and Constructional Aspects of Linguistic Explanation*, G. Webelhuth, J.-P. Koenig, and A. Kathol, Eds. Stanford: CSLI Publications, 1999, pp. 231–245.

[36] D. P. Flickinger, "On building a more efficient grammar by exploiting types," *Natural Language Engineering*, vol. 6, no. 1, pp. 15–28, 2000.

[37] G. Bouma, R. Malouf, and I. A. Sag, "Satisfying constraints on extraction and adjunction," *Natural Language and Linguistic Theory*, vol. 19, no. 1, pp. 1–65, 2001.

[38] R. D. Levine and T. E. Hukari, *The Unity of Unbounded Dependency Constructions*. Stanford: CSLI Publications, 2006.

[39] M. Tomasello, *Constructing a Language: A Usage-Based Theory of Language Acquisition*. Cambridge: Harvard University Press, 2003.

[40] Y. Bar-Hillel, M. A. Perles, and E. Shamir, "On formal properties of simple phrase-structure grammars," *Zeitschrift für Phonetik, Sprachwissenschaft und Kommunikationsforschung*, vol. 14, no. 2, pp. 143–172, 1961.

[41] S. Müller, "Elliptical constructions, multiple frontings, and surface-based syntax," in *Proceedings of Formal Grammar 2004, Nancy*, G. Jäger, P. Monachesi, G. Penn, and S. Wintner, Eds. Stanford: CSLI Publications, To Appear. [Online]. Available: http://hpsg.fu-berlin.de/~stefan/Pub/surface.html

[42] S. Oepen and D. P. Flickinger, "Towards systematic grammar profiling. Test suite technology ten years after," *Journal of Computer Speech and Language*, vol. 12, no. 4, pp. 411–436, 1998.

[43] O. Bonami, S. Müller, and P. Samvelian, *Persian in Head-Driven Phrase Structure Grammar*, In Preparation.

[44] M. Bijankhan, "The role of corpora in writing grammar," *Journal of Linguistics,* vol. 19, no. 2, pp. 48-67, 2004, Tehran: Iran University Press.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 469–476

ISBN 978-83-60810-27-9
ISSN 1896-7094

WordnetLoom: a Graph-based Visual Wordnet Development Framework

Maciej Piasecki, Michał Marcińczuk, Adam Musiał, Radosław Ramocki, Marek Maziarz
Institute of Informatics, Wrocław University of Technology, Poland
Email: {maciej.piasecki,michal.marcinczuk}@pwr.wroc.pl

***Abstract*—The paper presents WordnetLoom – a new version of an application supporting the development of the Polish wordnet called plWordNet. The primary user interface of WordnetLoom is a graph-based, graphical, active presentation of wordnet structure. Linguist can directly work on the structure of synsets linked by relation links. The new version is compared with the previous one in order to show the lines of development and to illustrate the introduced difference. A new version of WordnetWeaver – a tool supporting semi-automated expansion of wordnet is also presented. The new version is based on the same user interface as WordnetLoom, utilises all types of wordnet relations and is tightly integrated with the rest of the wordnet editor. The role of the system in the wordnet development process, as well as experience from its application, are discussed. A set of WWW-based tools supporting coordination of team work and verification is also presented.**

I. An Editor for Wordnet

A LARGE wordnet[1] is a very complex graph of many thousands vertices and arcs, where vertices represent lexical units and sets of lexical units, and arcs represent lexico-semantic relations of several types. A wordnet can be expressed in a simple formal language, cf the language defined for the needs of Princeton WordNet [3] and used in "lexical source files". However, the size and the complexity of a large wordnet makes manual encoding of the wordnet structure error prone. Error probability can be reduced by the introduction of specialised wordnet editors that free linguists from learning a formal language of description and provide some forms of error checking. Several wordnet editors have been proposed.

Two applications were constructed for the EuroWordNet project [4]: Polaris — an editing tool, and Periscope — a graphical database viewer. However, Polaris is a commercial tool, its development has been closed, and it is not commonly available for research. Moreover, it is a modification of a previous ontology editor, is tightly associated with the EuroWordNet structure and was constructed for a limited number of platforms.

Because of the Polaris limitations, a new tool called VisDic was created for the Czech WordNet [5]. In VisDic the relation definitions are still written in text windows, but an XML based format is used and some immediate browsing is possible in the tool, e.g. bi-directional browsing of graphs of semantic relations. VisDic was used in the Slovene Wordnet [6] and was available for research. VisDic was a monolithic application directly working on XML files, contrary to its direct descendant DEBVisDic [7] – a client-server, lexical database editor. DEBVisDic reimplements and extends the functionality of VisDic and is based on the client-server architecture and an XML database server. DEBVisDic has become a popular tool used in several projects.

Both tools are oriented on editing a wordnet synchronized with wordnets for other languages by the Interlingua Index [4], that complicates their basic structure and user interface.

For the needs of the GermaNet[2] development a dedicated wordnet editor called GernEdiT was constructed [8]. GernEdiT stores data in a relational database and provides concurrent access for many users at the same time. GernEdiT offers graph-based, graphical presentation of the wordnet structure, but limited to the hypernymic links only. Other types of relations, including relations between synsets (called *conceptual relations* in GermaNet) are not shown. Moreover, direct editing of the relation graph is not possible. Data must be first changed or added via dialogue forms. This makes the association between an action and the effect more remote from the user point of view. Direct editing of the graph is more compatible with the idea of GUI (Graphical User Interface).

The majority of existing tools for visualising a wordnet focus only on the hypernymy structure, e.g. [9], [10], and only some of the graph-based tools offer possibilities of editing, e.g. [11].

When the project on the construction of plWordNet[3] started in 2005, DEBVisDic [7] was not available and VisDic [5] was not an alternative, as it is a single machine tool. We needed a tool supporting a group of linguists working in a distributed environment. A dedicated wordnet editor called plWordNetApp was built. It was not as universal and flexible as DEBVisDic, but it implemented the assumed procedure of linguistic work. Its interface was designed according to the estimated ways of performing linguistic tasks. plWordNetApp enabled cooperation of a linguist team on the basis of a central database accessed via Internet. A brief characteristic of the editor is given in Sec. II.

plWordNetApp has been used by the Polish wordnet development team since the year 2006. During this period we

[1] A wordnet is a large electronic thesaurus following the basic construction features of Princeton WordNet [1], [2]. In WordNet, one and multi-word lexical units are grouped into *synsets* – "sets of near synonyms", that are linked by semantic relations derived from lexico-semantic relations.

[2] GermaNet is a wordnet for German, the second largest wordnet in the world.

[3] The first publicly available wordnet for Polish – Polish name: *Słowosieć*

978-83-60810-27-9/09/$25.00 © 2010 IEEE

collected rich experience concerning usability drawbacks of the editor. Editing and browsing complex graphs of hypernymy appeared to be the biggest problem. The problem was gradually rising with the increasing complexity of the plWordNet hypernymy structure[4]. Linguists requested visualisation of the hypernymy graphs which would be improved in the comparison to simple tree-based, static visualisation offered by plWordNetApp. Thus an improved version of the editor was constructed with the graphical presentation of the wordnet structure as a basis for all kinds of interaction.

The goal of this paper is to present a new wordnet editor called WordnetLoom, which is built around the idea of visual, direct editing of the wordnet graph structure. The editor is integrated with a tool called Wordnet Weaver, which delivers automatic support for extending the wordnet.

In the rest of the paper we will first briefly describe plWordNetApp – the basis for the present version of the application. Next the core graph-based view will be introduced. Full integration of WordnetWeaver with the rest of the application is discussed. The paper is concluded with the presentation of a set of web pages enabling browsing of data important for the coordinators of a linguistic group and plans for further development.

II. plWordNet 1.0 Development Tools

From the very beginning of designing plWordNetApp we were trying to find a solution for the necessity of presenting and editing complex wordnet structures and, at the same time, fitting the resulting complex structures of user interface onto one screen. In order to avoid cluttering the main screen, we decided to divide the user interface of the application into two main screens, presenting two main perspectives:

- the *perspective of lexical units*[5]
- and the *perspective of synsets*.

plWordNetApp GUI lets the lexicographers avoid the use of an artificial language for the description of semantic relations, starting with introduction of a new lexical unit (LU) and its description; this improves on the practice in WordNet and GermaNet [1], [15]. Both browsing and making decisions (during editing) are done via GUI screen controls and transparently recorded in the server database.

The perspective of lexical units, cf [13, pp. 39], was intended to support grouping lexical units into synsets. The lexical unit list, present in the system, can be filtered according to several criteria, e.g. a selected domain. To facilitate searching, each synset is also automatically assigned to the domain of its first lexical unit. Domains are the main tool in organising the work of the linguist team: linguists are assigned complete domains to process by the coordinator. Once a lexical unit has been selected, its properties are presented and can be edited. A new lexical unit can be added in the lexical unit perspective (but in some other parts of plWNApp GUI, too).

All synsets including a given lexical unit are shown in the tabbed panel below the lexical unit property panel, cf [13, pp. 41]. Editing of the selected synset is possible in the tabbed panel to the right of it — the second, hidden tab pane contains the synset properties. We assumed that first a selected lexical unit is assigned to an existing synset or freshly created one, and then the synset is edited.

In the hidden tab pane of the synset list panel, one can browse and edit a list of lexical relations[6] of the selected lexical unit, i.e. between pairs of lexical units, e.g. antonymy or derivational relations. Derivational relations play an important role in the structure of plWordNet [16].

In order to support the consistency among the linguists' decisions a *substitution test* defined for the given relation is presented to the user, prior to adding any new instance of a semantic relation. The test templates are defined and can be edited by coordinators in a dedicated window. On the basis of a template, a test instance is generated from the template by filling it with the inflection forms of the tested lexical units — the morphological analyser *Morfeusz* [17] was applied. The inflection form properties are specified in templates by IPIC codes. The mechanism of the tests makes plWordNetApp different from other wordnet editing tools, e.g. DEBVisDIC.

From the property panel of the selected synset, the user can switch to the synset perspective set to this synset as the *source synset*.

The five panels of this synset perspective, cf [13, pp. 41], can be divided into the following groups:

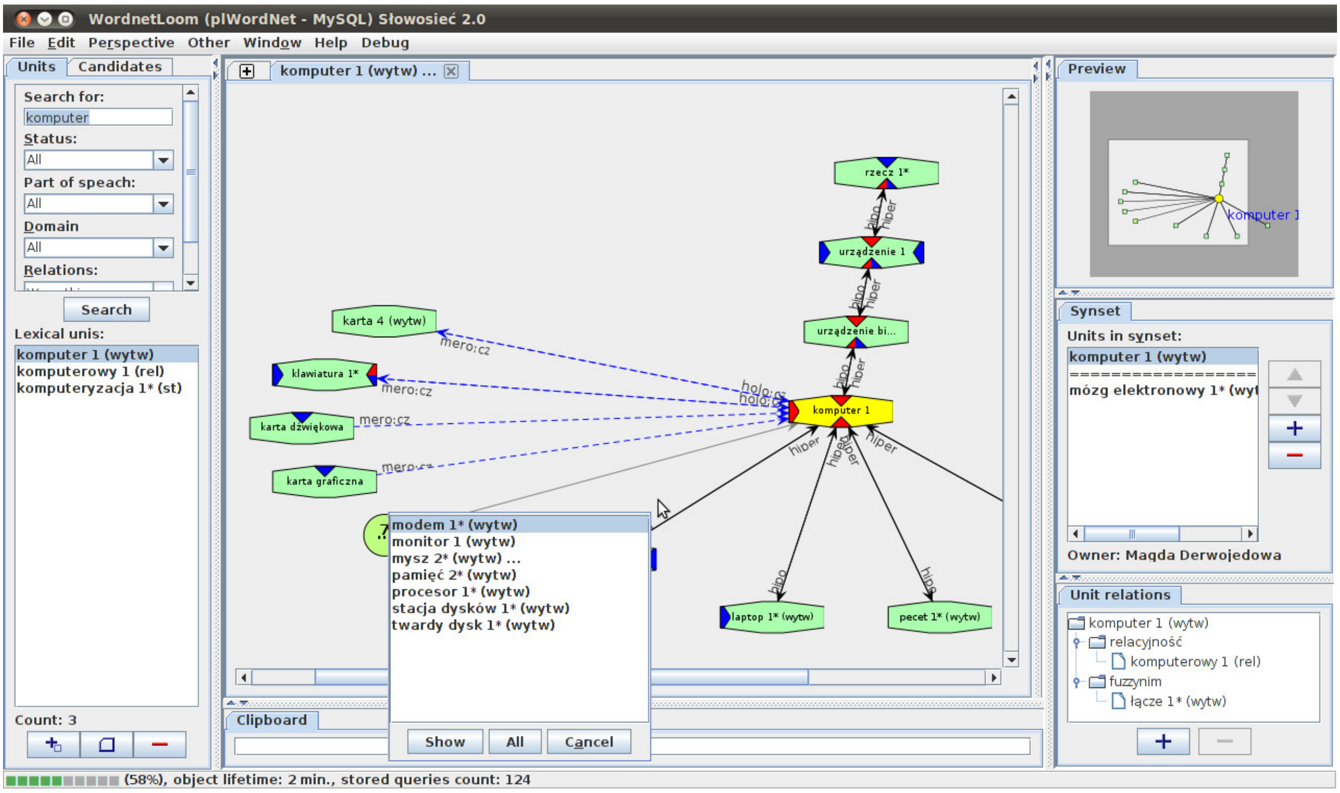
- selection and editing of a *source synset* – the synset for which we are going to define a relation or whose relations we are going to browse and edit,
- selection and editing of a *target synset* of a relation to be defined.
- browsing of existing relations.

There are two possible views of synset relations: a tabular one, cf [13, pp. 41] and a tree one (the hidden tab pane). According to the linguists' demands, the initial browsing facility was extended with editing synset relations directly in this view. The browsing panel also enables the navigation along the graph of relations. The possibility of editing synsets directly in this perspective was introduced in order to facilitate

[4] The hypernymy is a lexico-semantic relation between a lexical unit of a more general meaning and a semantically subordinated lexical unit of a more specific meaning – a relation similar to the superclass-subclass relation – e.g. 'plane figure' is a hypernym of 'tree diagram'.

[5] Informally, lexical unit is a semantically disambiguated word in a broad sense. Technically, a *lexical unit* is a pair: *lemma* and one of the *meanings* represented across different occurrences of this lemma in language utterances, cf [12], [13], e.g. a pair: the lemma *kolejka*, [polysemous] 'narrow-gauge railway', 'train', 'round' or 'queue', and its meaning represented in *plWordNet* 1.0 [14] by the symbol `kolejka 3` – a kind of railway, means of transport.
A *lemma* is a morphological word form selected as an exponent of the whole set of word forms of the same Part of Speech, such that all word forms from the set are described by the same shared grammatical categories and the word forms differ only in the values of the subsequent categories, e.g. *program* 'program' as an exponent of a set: *programu*, *programie*, *programach*, ...or *maszyna parowa* 'steam engine' (multiword lemma) as an exponent of the set: *maszyny parowej*, *maszynie parowej*, *maszyn parowych* ...; we assume that the choice of a particular word form as a lemma is arbitrary and constrained only by some tradition.

[6] By lexical relation we mean a lexico-semantic relation described by pair of lexical units, in contrary to synset relations that link synsets and originate from the lexical relations defined for the lexical units belonging to them.



Search for: komputer
Status: All
Part of speach: All
Domain: All
Relations:
Lexical unis:
komputer 1 (wytw)
komputerowy 1 (rel)
komputeryzacja 1* (st)
Count: 3

modem 1* (wytw)
monitor 1 (wytw)
mysz 2* (wytw) ...
pamięć 2* (wytw)
procesor 1* (wytw)
stacja dysków 1* (wytw)
twardy dysk 1* (wytw)

Units in synset:
komputer 1 (wytw)
====================
mózg elektronowy 1* (wyt
Owner: Magda Derwojedowa

komputer 1 (wytw)
relacyjność
komputerowy 1 (rel)
fuzzynim
łącze 1* (wytw)

(58%), object lifetime: 2 min., stored queries count: 124

Fig. 1. Screenshot of the active graph-based presentation of the wordnet structure (Glosses for synsets: *rzecz* 'thing', *urządzenie* 'device', *urządzenie biurowe* 'office device', *komputer* 'computer', *karta* 'card', *karta dźwiękowa/graficzna* 'audio/graphic card', *klawiatura* 'keyboard' *modem* 'modem', *monitor* 'monitor', *mysz* 'mouse', *pamięć* 'memory', *procesor* 'processor', *stacja dysków* 'disk drive', *twardy dysk* 'hard disk').

the correction of the initial synsets, e.g. it is possible to extract some lexical units from the source synset and to create a new hypernym synset.

Whenever a user wants to introduce an instance of a synset relation, the appropriate substitution test is presented. According to the definition of the synset relation introduced in plWN, it must be valid for any pair of lexical units from both synsets: the source and the target one. Thus, lexical semantic relations are extrapolated from relations on lexical units to relations on synsets. The substitution test window facilitates selecting all possible pairs of lexical units (from both synsets) and generating instances of the test.

Besides the main screen of the synset perspective, cf [13, pp. 41], an additional screen of synset editing was introduced during the plWordNetApp development. This screen is used for browsing synsets and have a layout similar to the lexical unit perspective: a large list of synsets on the left (rich filtering possibilities), and the tabular view of the selected synset relations, plus all synset editing panels on the right. The screen and the lexical unit perspective are synchronised, i.e. the filter setting and the selected synset and/or lexical unit are transferred to and from the lexical unit perspective during switching. It facilitates browsing the lexical unit relations of the lexical units belonging to the given synset.

III. Graph-based Wordnet Editor

The construction of plWordNetApp was focused on its use on slower computers as were used in 2005 and effectiveness of the primarily keyboard-based interaction. A new version of the application, called *WordnetLoom*[7], is built around the active, graphical presentation of the wordnet structure as a central element of the user interface. The presentation is described as 'active', as it enables not only browsing but also full editing of the structure. An example screenshot is presented in Fig. 1.

The organisation of the presentation is a mixture of the often used radial layout and a vertical tree-like scheme. The latter is often used for presenting large graphs and also wordnets, e.g. [9]–[11]. In the radial layout one synset (node of a graph) is selected as the central one (the present location of the user focus) and other synsets are presented around it in circular layers defined by the number of links from the centre.

According to our analysis, the radial layout expresses several drawbacks from the point of view of the active wordnet presentation. There is only one central element and simultaneous presentation of several sub-graphs sharing some nodes on one screen is difficult. A situation in which there are several

[7] The name refers to a static role played by the wordnet editor – it is a tool, which enables manual construction of a wordnet, while WordnetWeaver is a semi-automatic tool, which is a kind of 'active participant' of the wordnet development process.

paths from the central node to a node shared by the sub-graphs causes deformation of the layout. This is a problem for any layout, but the radial one is especially prone to this issue.

We modified this scheme in order to preserve presentation of the hypernymy in the form of a tree-like scheme. Thus, a synset which is selected by the user becomes the synset in focus. Along the vertical dimensions its hypernyms (direct and indirect) and hyponyms are presented, respectively above and below the central synset.

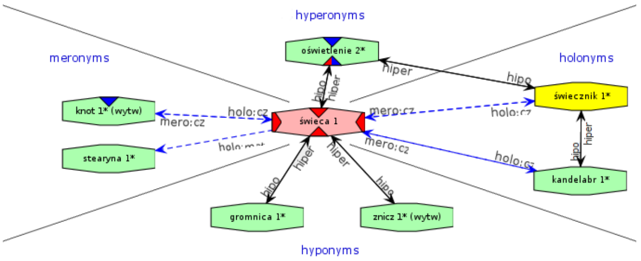


Fig. 2. Layout of the presentation of relation links (Glosses for synsets: *oświetlenie* 'lighting', *knot* 'wick', *świeca* 'candle', *kandelabr* 'candelabra', *stearyna* 'stearin', *znicz* 'candle', *świecznik* 'candlestick').

Meronymy and holonymy are shown along the horizontal dimensions: meronyms to the left and holonyms to the right, respectively. An example of the layout of relation presentation is shown in Fig. 2. Lines dividing the space and the relation names written in a blue bold face font are not a part of the screen, but they have been added for the presentation purpose only. Other types of synset relations, like near-synonymy[8], dweller (e.g. *mieszczanin* 'burgher' – *miasto* 'city'), markedness[9] are presented below the central synset, as they are close in their character to the hyponymy. Relations are visually differentiated by line style, line colour and labels. The definition of labels follows symbols used in the source files of Princeton WordNet. Optionally, shorten names of relations can be used instead, as it is shown in Fig.2.

Interaction starts with searching for lexical units in the left panel. Next, after a lexical unit has been selected the corresponding synset is shown as the central synset. Initially only synsets linked by the direct links to the central one are presented. However, the user can navigate by pressing triangular buttons on the sides of the octagonal synset symbols. The colour of the triangular buttons signals whether none, some or all links of the given type have been opened for the given synset.

A synset can contain several lexical units[10], each lexical unit can participate in several lexical relations, a synset can have several dozen relation links and there can be many synsets – it is impossible to show legibly all this information on one computer screen at the same time. Thus we had to remove a lot of data from the main panel and leave only the minimum of information required for the general understanding of the structure on it. We left only one lexical unit presented per synset, this is compatible with a linguistic practice of assigning the lexical unit which is the most representative for the given synset to first position. Moreover, in the case of a large wordnet the vast majority of synsets include only one lexical unit.

The whole set of lexical units of the central synset can be browsed and edited on the `Synset` panel – the pane in the middle of the right most column in Fig. 1.

Lexical relations have been moved to the separate pane in the bottom-right corner – `Lexical relations` in Fig. 1. Only relations for a lexical unit which is selected in the `Synset` panel are presented. Keeping the detailed information concerning synsets and lexical units in the separate panels defers access to this information and increases the load of the user's short-term memory, but it is a compromise with the possibility of having a more broad view on the wordnet structure.

Synsets are assigned a minimal representation in the graph, but still a wordnet structure of relations can be quite dense in some parts, and many synsets located in the upper parts of the hypernymy structure can have several dozens or even hundreds[11] of outgoing hyponymic links. Thus, by default, WordnetLoom presents only up to $k = 5$ (a parameter) links of the given synset relation (presently only hyponymy). If the number of links is greater, than only the first $k - 1$ are automatically presented and the rest is folded and hidden under the symbol of a circle with the number of hidden synsets shown on it. After clicking the circle a small list-window is shown, the user can browse across hidden links and unfold any sub-group of them – this situation is presented near the bottom of the main panel in Fig. 1. In any moment a link can be folded back into the group of the hidden links.

In large graphs the user can quickly get lost seeing only a small portion of a graph. As a support for navigating in the space of a large graph we introduced a mini-map overview panel in the top-right corner in Fig. 1. Synsets are presented only as point-size circles, but colours are used to distinguish the central synset and selected synsets. Moreover, synset descriptions are given in tooltips shown in response to holding the mouse pointer over some circle.

Visual, graph presentation gives an opportunity to perform editing directly on the relation structure. The context and potential consequences of an action to the presented part of the wordnet structure are clearly visible on the screen. However, only a subgraph can be presented and the biggest problem was to find a way for adding new relation links from synsets presented on the screen to those not. A gradual enlargement of the subgraph being unfolded by clicking triangular buttons is always possible, but only to some space limits. Unfolding of huge graphs is not an option if one has to look for the target

[8] Near-synonymy is different than synonymy expressed by synsets, and is used to link lexical units of the very close meaning but belonging to the different language registers, e.g. *chłopiec* 'boy' – *gówniarz* 'squirt'.

[9] Markedness is relation originating from the regular derivational associations and encompasses several sub-types like dimunitives, augmentative, expressive forms etc.

[10] There are a few synsets which include about 20 LUs each.

[11] E.g. *czynność* 'activity' has 376 hyponyms, mostly gerunds.

synset of a link in some different part of the wordnet. In order to facilitate simultaneous browsing of different parts of the wordnet the central panel was implemented as a multi-panel window enabling opening each sub-graph in a new sub-panel of the central panel. However, switching between sub-panels complicates the way in which selection of a target synset is performed. The user has to remember the state of the user interface as "in selecting a target". In order to make both the source and the target of the action accessible without switching between sub-panels, we introduced a *clipboard* – and additional panel – in which the user can keep references to the synsets that he is working on, are interesting to him, etc. `Clipboard` is located below the main panel, see Fig. 1. When adding a new relation link the user first selects the source synset by right clicking it, and then selects the target synset from the active sub-panel or from the clipboard. Only synset symbols are shown in the clipboard but one can activate a sub-panel, which is already open, or open a new sub-panel for a synset in the clipboard with mouse double clicking on its symbol.

The functionality of simultaneous opening several graphs in separate sub-panels supports comparing different parts of the wordnet structure – a challenging task but very important in the case of a large wordnet developed by a team of several linguists. Erroneously separated parts of the hypernymy structure can be immediately linked, and other corrections can be introduced, too.

It is worth to emphasise here that the graph-based editing and presentation is a significant advantage offered by *WordnetLoom* in comparison to *plWordnetApp*, at least according to the opinions of the linguist team members. The most important improvement is that a linguist can work on all relations (linking synsets and lexical units) using one set of panels without the necessity of switching to other screens. Moreover, he can still follow the general wordnet structure as it is defined by the synset relations and can be perceived only from the perspective of groups of inter-linked synsets not only singular synsets.

IV. Integrated Semi-automated Support

WordnetLoom includes a new version of WordnetWeaver as its integrated part [13]. WordnetWeaver is a tool supporting semi-automated expansion of plWordNet. It consists of the two parts:

- the algorithm of Activation-Area Attachment (henceforth, AAA) which for a new lemma from the outside of plWordNet generates a set of suggested new lexical units on the basis of several knowledge sources,
- and an user interface based on the active presentation of hypernymy subgraphs as descriptions of the suggested relations.

The latest version of the AAA algorithm was described in details in [18], here only a brief characteristics is given below. AAA utilises several knowledge sources describing semantic association of lemma pairs. The used set of five knowledge sources includes: Measure of Semantic Relatedness (1), which has been constructed automatically on the basis of a large corpus, lemma pairs – possibly hypernymic – extracted from the corpus by hand-written patterns (2) and also patterns learned automatically (3). Moreover, lemma pairs for which the measure returns high values are filtered by a classifier (4), which has been trained for the recognition of wordnet relations and finally lemma pairs in which both lemmas are mutually high on the their lists of the mutually close semantically related according to the measure (5), cf [13].

AAA works in three steps, described below in the perspective of adding one new lemma to plWordNet.

1) Semantic fit between a new lemma and each lemma in the wordnet is calculated on the basis of weighted voting applied to information concerning the new lemma from the knowledge sources.
2) Semantic fit between the new lemma and each synset is calculated:
    a) on the basis of cumulated fit between the new lemma and the synset lemmas,
    b) and also on the basis of the fit between the new lemma and synsets that are located in the close distance to the given synset.
3) Connected hypernymic sub-graphs of synsets expressing higher semantic fit to the new lemma are identified and presented as descriptions of potential new lexical units for the new lemma.

Connected sub-graphs identified by AAA are presented graphically to the user, e.g. in Fig. 3 for the new lemma *flinta* ('shotgun', 'flint-lock gun') one lexical unit is suggested and presented as a graph with the highest fit for the synset including *strzelba* 'shotgun'. On the screen, the purple oval represents the new lemma, synsets of the colours from yellow to red are elements of the identified sub-graph and green synsets do not belong to the suggestion but have been unfolded manually by the user. Synset colour expresses the strength of semantic fit between it and the new lemma: light yellow means weak strength of fit, darker yellow and red represents increasing strength. In the case of synsets presented as rectangles the fit was calculated on the basis of the Measure of Semantic Relatedness only ("weak fit" in [13], [18]), while octagons signal stronger fit supported by data from several knowledge sources ("strong fit" in [13], [18]).

The main change in the WordnetWeaver user interface is its full integration with WordnetLoom. Thus, suggestions are shown as hypernymic sub-graphs, but the user can browse relations of any type and add a new lexical unit for the new lemma with the help of any relation. In the previous version of WordnetWeaver [13] only hypernymic links could be browsed. A new lemma could be added to an existing synset member or as a new hypo/hypernym, but any other relation could not be directly used. WordnetWeaver was initially focused on expanding a wordnet along synonymy and hypernymy dimensions. That is why its effective use was limited to nouns, and the first attempts to apply it to verbs failed, as the verbal hypernymy structure is very limited. However knowledge sources assign high values of semantic association to lemma pairs that are linked not only by synonymy/hypernymy, but also by other lexico-semantic relations, e.g. meronymy/holonymy. Thus new LUs suggested for a lemma can be also motivated by relations other than synonymy/hypernymy.

In contrast to plWordnetApp, the user interface of new version of WordnetWeaver, presented here, has been changed to the new one developed for WordnetLoom. As a results, see e.g. Fig. 3, the user now has access to any relation defined in the wordnet and new lexical units can be attached by any relation. A new algorithm of the layout generation prevents overlapping presentation of synsets on the screen.

WordnetWeaver was intended to be a tool for a team of linguists working simultaneously, rather than for an individual linguist. Thus the work procedure is based on assigning disjoint sets of lemmas, called 'packages', to particular linguists. Packages are numbered and a package can be selected using the text edit called `Package number`. Instead of dividing lemmas into packages on the basis of their alphabetic order, lemmas are semantically grouped by the clustering algorithm implemented in CLUTO system [19] applied to the results of the Measure of Semantic Relatedness, cf [13]. We did not expect perfect clustering, e.g. in terms of cluster purity, we only wanted to achieve a practical effect of groups generally semantically coherent. Created lemma groups appeared to be quite coherent, on average 2–3 semantic domains (e.g. names of professions, food, plants, minerals, etc.) can be noticed. During two years of intensive practical use this method based on the off-the-shelf clustering tool proved its value. In the new version of the WordnetWeaver we added a mechanism of temporal blocking a package for the linguist editing it. The mechanism is transparent to the users and blocking starts with the first editing actions and is automatically removed after the linguist has closed the client application or switched to editing of another package.

Quality of suggestions generated by the AAA depends on the local quality of the wordnet structure, e.g. if for there are no appropriate hypernyms for the new lemmas in the wordnet structure, the generated suggestion can be accidental. Thus, WordnetWeaver is equipped with mechanism of re-computing suggestions, which is activated on the user request (the button `Re-compute`). This facility can be especially helpful in case of constructing a new hypernymy sub-graph for some domain which was scarcely populated at the beginning. In the older version of WordnetWeaver the re-computation was run for all packages, in the present version the process is limited to one package only and can serve simultaneously several requests putting them into an internal queue.

V. Management Tools

The linguistic team is organised into two subgroups: wordnet editors and coordinators. The task of coordinators is to take care of the consistency of decisions made by different editors and to check the quality of changes introduced. In order to support the work of coordinators and facilitate the team cooperation a set of tools was developed. The tools are accessible via web pages in order to make usable even in the case of low bandwidth network connections.

First, browsing all changes is possible, e.g. adding a new wordnet element (like lexical unit, synset, relation link, etc.), deleting and modifying. Each change is registered together with information concerning the exact time and person who made it. An example screenshot is presented in Fig. 4. The filtering functionality – the top-right part of the screen – enables limiting the data to particular period, linguist or domain. Moreover, each team member can search for the source of the observed wordnet element, e.g. in order to get explanations concerning the reason of the introduction of some specific lexical unit (word sense).

On the additional web page, coordinators can observe the pace of work in relation to particular linguists, and on the basis of the collected data they are able also to plan the distribution of work among linguists.

Some diagnostic reports have been introduced, as well. Coordinators can obtain data concerning synsets without hypernyms and/or any other relation link[12]. These reports are the basis for correcting the structure and joining separated hypernymic sub-graphs into larger structures in those cases where it is supported by the linguistic data. The whole hypernymy graph is being constructed in plWordNet in the bottom-up direction, i.e. starting with more specific lexical units and their relations.

The set of management tools is being continuously extended but even its present version is frequently used by the coordinators and the linguistic team.

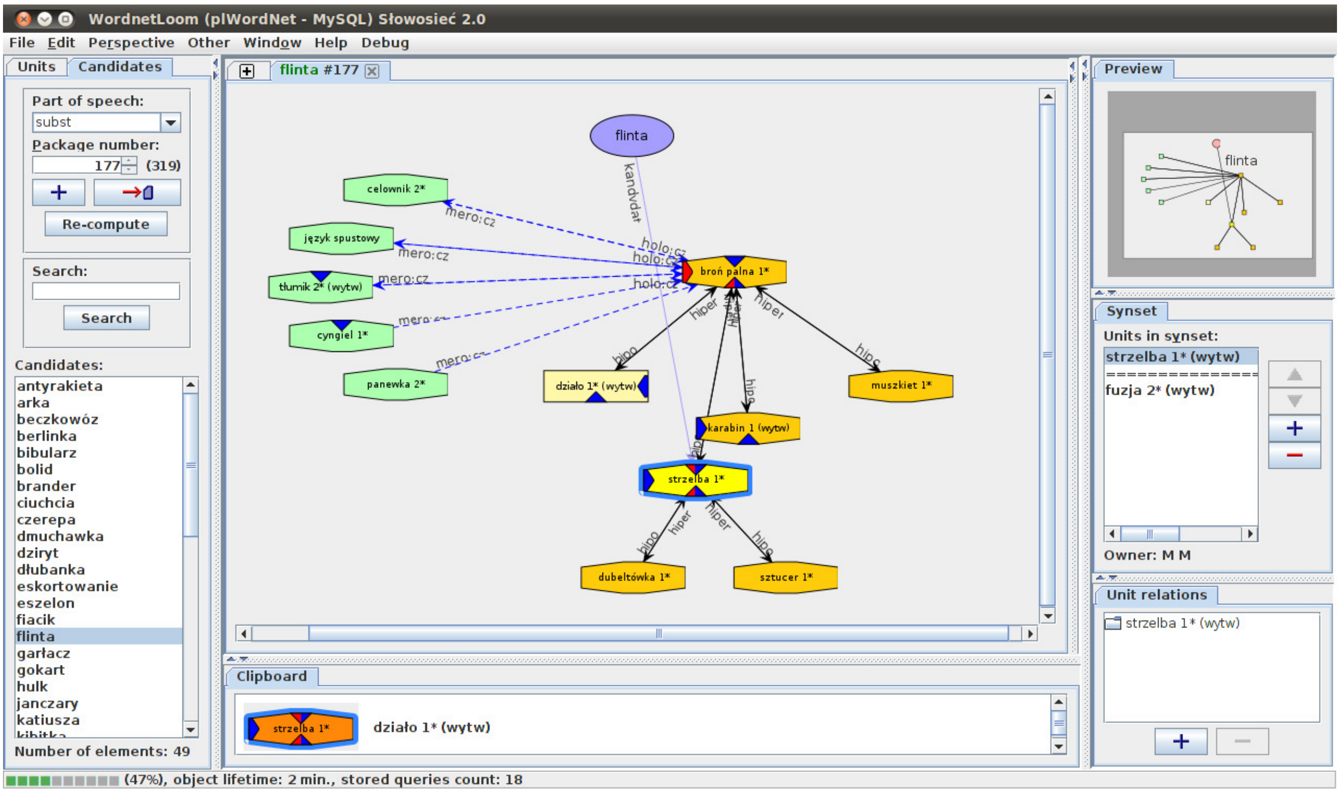
VI. Further Research

The vast majority of WordnetLoom is now implemented. Its new version described in this paper has been used for the last three months and about 9000 LUs and almost 20000 relation links have been added into plWordNet with its help. The idea of the primarily graph-based user interface in a wordnet editing tool seems to work well in practice, especially in the case of group work on expanding larger wordnet in the process driven by data extracted from large corpora. In such a process different parts of the wordnet are being developed simultaneously, and the linguists must follow the changes in an appropriate way.

Concerning the application development, we concentrate now on fixing problems which appeared during the work of linguists, as well, as minor improvements introduced in response to requests of linguists.

We work now on a significant extension of the AAA algorithm which is the basis for WordnetWeaver. We aim for the use of all types of relation links as a basis for expansion (now, only the hypernymy structure is used) and generating suggestions targeted at particular relations as description of the

[12] The presence of a synset which is not linked to any hypernym does not necessarily mean an error in the wordnet. In plWordNet every relation link must be supported by the linguistic analysis. Thus it is only required that there must be at least one relation link for a synset, and each lexical unit (but not lemma) must belong to exactly one synset.



Part of speech: subst

Package number: 177 (319)

Search:

Candidates:

antyrakieta
arka
beczkowóz
berlinka
bibularz
bolid
brander
ciuchcia
czerepa
dmuchawka
dziryt
dłubanka
eskortowanie
eszelon
fiacik
flinta
garłacz
gokart
hulk
janczary
katiusza

Number of elements: 49

strzelba 1* działo 1* (wytw)

Units in synset:

strzelba 1* (wytw)
fuzja 2* (wytw)

Owner: M M

strzelba 1* (wytw)

(47%), object lifetime: 2 min., stored queries count: 18

Fig. 3. Screenshot of Wordnet Weaver tool supporting semi-automated wordnet expansion (Glosses for synsets: *flinta* ‘shotgun’, *celownik* ‘sight’, *cyngiel* ‘trigger’, *język spustowy* ‘trigger’, *panewka* ‘pan’, *tłumik* ‘silencer’, *broń palna* ‘firearm’, *działo* ‘gun’, *karabin* ‘rifle’, *muszkiet* ‘musket’), *strzelba* ‘shotgun’, *dubeltówka* ‘double-barrelled gun’, *sztucer* ‘rifle’)

attachment place, i.e. a synset selected as a point of attachment will be described by the type of relation by which a new lexical unit should be linked to it.

WordnetLoom can be used for editing other wordnets than plWordNet. The application uses UTF encoding for characters, relation types are defined in the XML-based format of wordnet data and the list of domains can be easily extended (now we use the Princeton WordNet list of domains). Lists of synset relations that are presented in vertical and horizontal directions are defined in the set-up file. Import and export to the Princeton WordNet file format has been implemented (but the use of WordnetWeaver requires preparation of the appropriate data knowledge sources). The suggestion presentation mechanism can work on the basis of any list of triples: word, synset, fit value. However, the AAA algorithm needs tuning to knowledge sources other than those used by us. WordnetLoom can be used for editing any network of lexical semantic relations or even an ontology. However, in the latter case, the limited, fixed set of the LU properties can be a strong limitation: concepts in an ontology are mostly described by an expandable attribute-value structures while lexical units are attached a limited, fixed set of properties in a wordnet.

Acknowledgment

Work financed by Innovative Economy Programme project POIG.01.01.02-14-013/09.

References

[1] C. Fellbaum, Ed., *WordNet — An Electronic Lexical Database*. The MIT Press, 1998.

[2] G. A. Miller and C. Fellbaum, “Wordnet then and now,” *Lang Resources & Evaluation*, vol. 41, pp. 209–214, 2007.

[3] R. I. Tengi, *Design and Implementation of the WordNet Lexical Database and Searching Software*. The MIT Press, 1998, ch. 4, pp. 105–127.

[4] P. Vossen, “EuroWordNet general document version 3,” University of Amsterdam, Tech. Rep., 2002.

[5] A. Horák and P. Smrž, “New features of wordnet editor VisDic,” *Romanian Journal of Information Science and Technology*, vol. 7, no. 1–2, pp. 201–213, 2004.

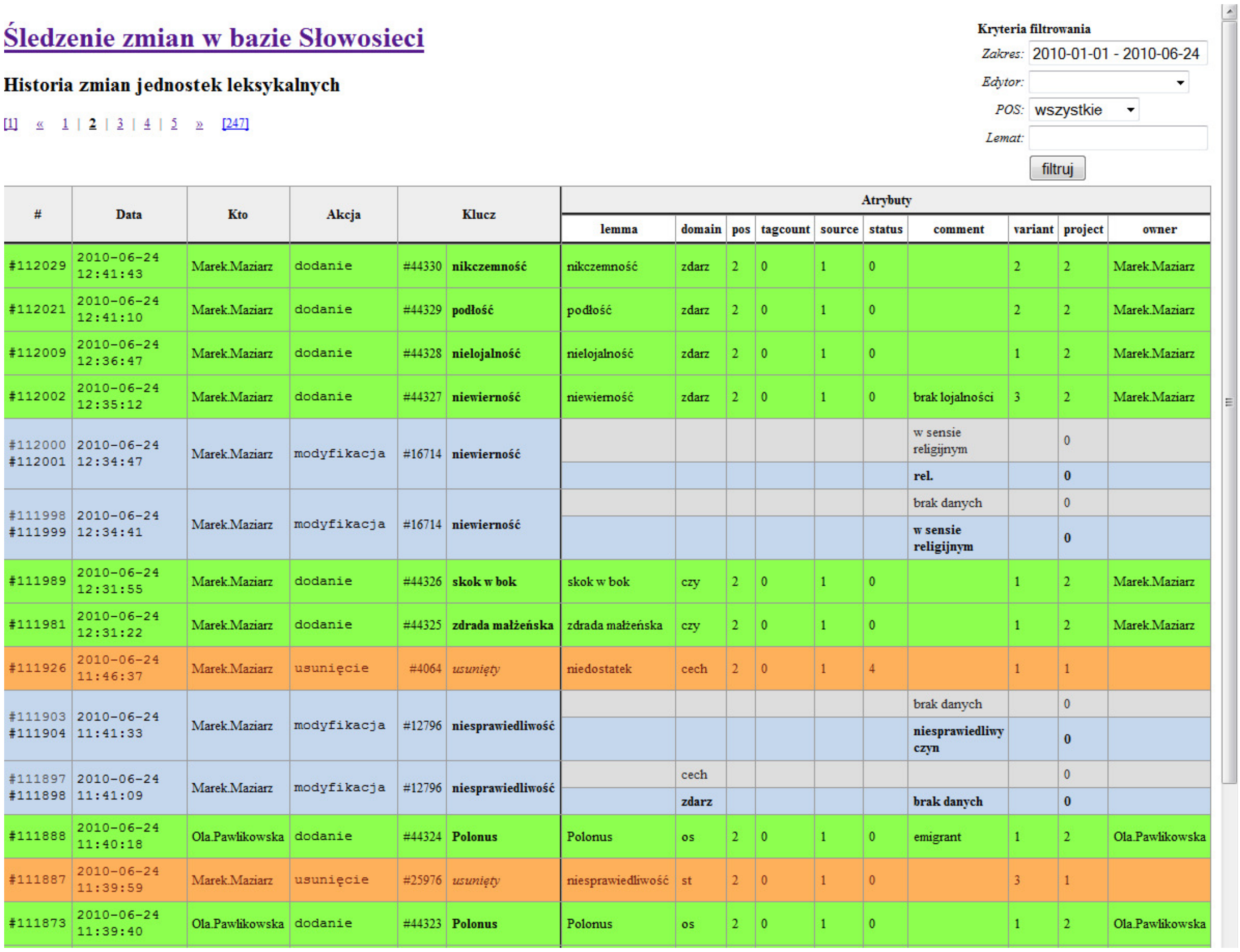
[6] T. Erjavec and D. Fišer, “Building the Slovene Wordnet: first steps, first problems,” in *Proceedings of the Third International WordNet Conference — GWC 2006*, 2006.

[7] A. Horák, K. Pala, A. Rambousek, and M. Povolný, “DEBVisDic — first version of new client-server wordnet browsing and editing tool,” in *Proceedings of the Third International WordNet Conference — GWC 2006*. Masaryk University, 2006, pp. 325–328. [Online]. Available: http://nlp.fi.muni.cz/publications/gwc2006_hales_pala_etal/gwc2006_hales_pala_etal.pdf

[8] V. Henrich and E. Hinrichs, “Gernedit - the germanet editing tool,” in *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC’10)*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odjik, S. Piperidis, M. Rosnera, and D. Tapias, Eds. Valletta, Malta: European Language Resources Association (ELRA), may 2010.

[9] J. Kamps and M. Marx, “Visualizing WordNet structure,” in *Proceedings of the 1st International Conference on Global WordNet,*, Mysore, India, 2002, pp. 182–186.

[10] C. Collins, “WordNet Explorer: Applying visualization principles to lexical semantics,” Computational Linguistics Group, Department of Computer Science, University of Toronto, Toronto, Ontario, Canada,



Śledzenie zmian w bazie Słowosieci

Historia zmian jednostek leksykalnych

[1] « 1 | 2 | 3 | 4 | 5 » [247]

Kryteria filtrowania

Zakres: 2010-01-01 - 2010-06-24

Edytor:

POS: wszystkie

Lemat:

#	Data	Kto	Akcja	Klucz		Atrybuty									
						lemma	domain	pos	tagcount	source	status	comment	variant	project	owner
#112029	2010-06-24 12:41:43	Marek.Maziarz	dodanie	#44330	**nikczemność**	nikczemność	zdarz	2	0	1	0		2	2	Marek.Maziarz
#112021	2010-06-24 12:41:10	Marek.Maziarz	dodanie	#44329	**podłość**	podłość	zdarz	2	0	1	0		2	2	Marek.Maziarz
#112009	2010-06-24 12:36:47	Marek.Maziarz	dodanie	#44328	**nielojalność**	nielojalność	zdarz	2	0	1	0		1	2	Marek.Maziarz
#112002	2010-06-24 12:35:12	Marek.Maziarz	dodanie	#44327	**niewierność**	niewierność	zdarz	2	0	1	0	brak lojalności	3	2	Marek.Maziarz
#112000 #112001	2010-06-24 12:34:47	Marek.Maziarz	modyfikacja	#16714	**niewierność**							w sensie religijnym		0	
												rel.		**0**	
#111998 #111999	2010-06-24 12:34:41	Marek.Maziarz	modyfikacja	#16714	**niewierność**							brak danych		0	
												w sensie religijnym		**0**	
#111989	2010-06-24 12:31:55	Marek.Maziarz	dodanie	#44326	**skok w bok**	skok w bok	czy	2	0	1	0		1	2	Marek.Maziarz
#111981	2010-06-24 12:31:22	Marek.Maziarz	dodanie	#44325	**zdrada małżeńska**	zdrada małżeńska	czy	2	0	1	0		1	2	Marek.Maziarz
#111926	2010-06-24 11:46:37	Marek.Maziarz	usunięcie	#4064	*usunięty*	niedostatek	cech	2	0	1	4		1	1	
#111903 #111904	2010-06-24 11:41:33	Marek.Maziarz	modyfikacja	#12796	**niesprawiedliwość**							brak danych		0	
												niesprawiedliwy czyn		**0**	
#111897 #111898	2010-06-24 11:41:09	Marek.Maziarz	modyfikacja	#12796	**niesprawiedliwość**		cech							0	
							zdarz					**brak danych**		**0**	
#111888	2010-06-24 11:40:18	Ola.Pawlikowska	dodanie	#44324	**Polonus**	Polonus	os	2	0	1	0	emigrant	1	2	Ola.Pawlikowska
#111887	2010-06-24 11:39:59	Marek.Maziarz	usunięcie	#25976	*usunięty*	niesprawiedliwość	st	2	0	1	0		3	1	
#111873	2010-06-24 11:39:40	Ola.Pawlikowska	dodanie	#44323	**Polonus**	Polonus	os	2	0	1	0		1	2	Ola.Pawlikowska

Fig. 4. Screenshot of the WWW-based browser of changes introduced in plWordNet.

Technical Report, 2008. [Online]. Available: http://faculty.uoit.ca/collins/publications/docs/TR-wordnet-explorer.pdf

[11] J. Szymański, "Wordventure – cooperative wordnet editor. Architecture for lexical semantic aquisition," in *Proceedings of the international conference on Knowledge Engineering and ontology developement, INSTICC*, Funchal-Madeira, Portugal, 2009.

[12] M. Derwojedowa, M. Piasecki, S. Szpakowicz, M. Zawisławska, and B. Broda, "Words, concepts and relations in the construction of Polish WordNet," in *Proceedings of the Global WordNet Conference, Seged, Hungary January 22–25 2008*, A. Tanâcs, D. Csendes, V. Vincze, C. Fellbaum, and P. Vossen, Eds. University of Szeged, 2008, pp. 162–177. [Online]. Available: http://www.plwordnet.pwr.wroc.pl/main/content/files/publications/gwc08-plWN-paper.pdf

[13] M. Piasecki, S. Szpakowicz, and B. Broda, *A Wordnet from the Ground Up*. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2009. [Online]. Available: http://www.plwordnet.pwr.wroc.pl/main/content/files/publications/A_Wordnet_from_the_Ground_Up.pdf

[14] M. Derwojedowa, M. Głabska, M. Piasecki, J. Rabiega-Wiśniewska, S. Szpakowicz, and M. Zawisławska, "plWordNet 1.0 — The Polish Wordnet," April 2009, online access to the database of plWordNet 1.0: www.plwordnet.pwr.wroc.pl. [Online]. Available: http://plwordnet.pwr.wroc.pl/browser/?lang=en

[15] B. Hamp and H. Feldweg, "GermaNet — a lexical-semantic net for German," in *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications, Madrid*. ACL, 1997.

[16] M. Derwojedowa, M. Piasecki, S. Szpakowicz, and M. Zawisławska, "Polish WordNet on a shoestring," in *Proceedings of Biannual Conference of the Society for Computational Linguistics and Language Technology, Tübingen, April 11-13 2007*. Universität Tübingen, 2007, pp. 169–178. [Online]. Available: http://www.plwordnet.pwr.wroc.pl/main/content/files/publications/piasecki-gldv07-final.pdf

[17] M. Woliński, "Morfeusz – a practical tool for the morphological analysis of Polish." ser. Advances in Soft Computing, M. A. Kłopotek, S. T. Wierzchoń, and K. Trojanowski, Eds. Berlin: Springer, 2006, pp. 511–520.

[18] M. Piasecki, B. Broda, M. Głąbska, M. Marcińczuk, and S. Szpakowicz, "Semi-automatic expansion of polish wordnet based on activation-area attachment," in *Recent Advances in Intelligent Information Systems*. Academic Publishing House EXIT, 2009, pp. 247–260. [Online]. Available: http://iis.ipipan.waw.pl/2009/proceedings/iis09-25.pdf

[19] G. Karypis, "CLUTO - a clustering toolkit," Tech. Rep. #02-017, nov 2003.

[20] A. Zgrzywa, Ed., *Proceedings of Multimedia and Network Information Systems*. Oficyna Wydawnicza Politechniki Wrocławskiej, 2006.

[21] M. A. Kłopotek, S. T. Wierzchoń, and K. Trojanowski, Eds., *Intelligent Information Processing and Web Mining – Proceedings of the International IIS: IIPWM'06 Conference held in Wisła, Poland, June 2006*, ser. Advances in Soft Computing. Berlin: Springer, 2006.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 477–484

ISBN 978-83-60810-27-9
ISSN 1896-7094

Building and Using Existing Hunspell Dictionaries and TEX Hyphenators as Finite-State Automata

Tommi A Pirinen, Krister Lindén
University of Helsinki,
Department of Modern Languages
Unionkatu 40, FI-00014 University of Helsinki, Finland
Email: {tommi.pirinen,krister.linden}@helsinki.fi

***Abstract*—There are numerous formats for writing spell-checkers for open-source systems and there are many descriptions for languages written in these formats. Similarly, for word hyphenation by computer there are TEX rules for many languages. In this paper we demonstrate a method for converting these spell-checking lexicons and hyphenation rule sets into finite-state automata, and present a new finite-state based system for writer's tools used in current open-source software such as Firefox, OpenOffice.org and enchant via the spell-checking library voikko.**

I. Introduction

CURRENTLY there is a wide range of different free open-source solutions for spell-checking and hyphenation by computer. For hyphenation the ubiquitous solution is the original TEX algorithm described in [1]. The most popular of the spelling dictionaries are the various instances of *spell software, i.e. ispell[1], aspell[2], myspell and hunspell[3] and other *spell derivatives. The TEX hyphenation patterns are readily available on the Internet to cover some 49 languages. The hunspell dictionaries provided with the OpenOffice.org suite cover 98 languages.

The program-based spell-checking methods have their limitations because they are based on specific program code that is extensible only by coding new features into the system and getting all users to upgrade. E.g. hunspell has limitations on what affix morphemes you can attach to word roots with the consequence that not all languages with rich inflectional morphologies can be conveniently implemented in hunspell. This has already resulted in multiple new pieces of software for a few languages with implementations to work around the limitations, e.g. emberek (Turkish), hspell (Hebrew), uspell (Yiddish) and voikko (Finnish). What we propose is to use a generic framework of finite-state automata for these tasks. With finite-state automata it is possible to implement the spell-checking functionality as a one-tape weighted automaton containing the language model and a two-tape weighted automaton containing the error model. This also allows simple use of unigram training for optimizing spelling suggestion results [2]. With this model, extensions to context-based n-gram models for real-word spelling error problems [3] are also possible.

We also provide a method for integrating the finite-state spell-checking and hyphenation into applications using an open-source spell-checking library voikko[4], which provides a connection to typical open-source software, such as Mozilla Firefox, OpenOffice.org and the Gnome desktop via enchant.

II. Definitions

In this article we use weighted two-tape finite-state automata—or weighted finite-state transducers—for all processing. We use the following symbol conventions to denote the parts of a weighted finite-state automaton: a transducer $T = (\Sigma, \Gamma, Q, q_0, Q_f, \delta, \rho)$ with a semi-ring $(S, \oplus, \otimes, \overline{0}, \overline{1})$ for weights. Here Σ is a set with the input tape alphabet, Γ is a set with the output tape alphabet, Q a finite set of states in the transducer, $q_0 \in Q$ is an initial state of the transducer, $Q_f \subset Q$ is a set of finite states, $\delta : Q \times \Sigma \times \Gamma \times S \to Q$ is a transition relation, $\rho : Q_f \to S$ is a final weight function. A successful path is a list of transitions from an initial state to a final state with a weight different from $\overline{0}$ collected from the transition function and the final state function in the semi-ring S by the operation $\otimes$. We typically denote a successful path as a concatenation of input symbols, a colon and a concatenation of output symbols. The weight of the successful path is indicated as a subscript in angle brackets, *input:output*$_{<w>}$. A path transducer is denoted by subscripting a transducer with the path. If the input and output symbols are the same, the colon and the output part can be omitted.

The finite-state formulation we use in this article is based on Xerox formalisms for finite-state methods in natural language processing [4], in practice lexc is a formalism for writing right linear grammars using morpheme sets called lexicons. Each morpheme in a lexc grammar can define their right follower lexicon, creating a finite-state network called a *lexical transducer*. In formulae, we denote a lexc style lexicon named X as Lex_X and use the shorthand notation $Lex_X \cup$ *input:output* Y to denote the addition of a lexc string or morpheme, `input:output Y ;` to the `LEXICON X`. In the same framework, the twolc formalism is used to describe context restrictions for symbols and their realizations in the form of parallel rules as defined in the appendix of [4]. We use $Twol_Z$ to denote the rule set Z and use the shorthand

[1] http://www.lasr.cs.ucla.edu/geoff/ispell.html
[2] http://aspell.net
[3] http://hunspell.sf.net
[4] http://voikko.sf.net

notation $Twol_Z \cap a{:}b \leftrightarrow l\ e\ f\ t\_r\ i\ g\ h\ t$ to denote the addition of a rule string `a:b <=> l e f t _ r i g h t ;` to the rule set Z, effectively saying that *a:b* only applies in the specified context.

A spell-checking dictionary is essentially a single-tape finite-state automaton or a language model T_L, where the alphabet $\Sigma_L = \Gamma_L$ are characters of a natural language. The successful paths define the correctly spelled word-forms of the language [2]. If the spell-checking automaton is weighted, the weights may provide additional information on a word's correctness, e.g. the likelihood of the word being correctly spelled or the probability of the word in some reference corpus. The spell-checking of a word s is performed by creating a path automaton T_s and composing it with the language model, $T_s \circ T_L$. A result with the successful path $s_{<W>}$, where W is greater than some threshold value, means that the word is correctly spelled. As the result is not needed for further processing as an automaton and as the language model automaton is free of epsilon cycles, the spell-checking can be optimized by performing a simple traversal (lookup) instead, which gives a significant speed-advantage over full composition [5].

A spelling correction model or an error model T_E is a two-tape automaton mapping the input text strings of the text to be spell-checked into strings that may be in the language model. The input alphabet Σ_E is the alphabet of the text to be spell-checked and the output alphabet is $\Gamma_E = \Sigma_L$. For practical applications, the input alphabet needs to be extended by a special any symbol with the semantics of a character not belonging to the alphabet of the language model in order to account for input text containing typos outside the target natural language alphabet. The error model can be composed with the language model, $T_L \circ T_E$, to obtain an error model that only produces strings of the target language. For space efficiency, the composition may be carried out during runtime using the input string to limit the search space. The weights of an error model may be used as an estimate for the likelihood of the combination of errors. The error model is applied as a filter between the path automaton T_s compiled from the erroneous string, $s \notin T_L$, and the language model, T_L, using two compositions, $T_s \circ T_E \circ T_L$. The resulting transducer consists of a potentially infinite set of paths relating an incorrect string with correct strings from L. The paths, $s : s^i_{<w_i>}$, are weighted by the error model and language model using the semi-ring multiplication operation, $\otimes$. If the error model and the language model generate an infinite number of suggestions, the best suggestions may be efficiently enumerated with some variant of the n-best-paths algorithm [6]. For automatic spelling corrections, the best path may be used. If either the error model or the language model is known to generate only a finite set of results, the suggestion generation algorithm may be further optimized.

A hyphenation model T_H is a two-tape automaton mapping input text strings of the text to be hyphenated to possibly hyphenated strings of the text, where the input alphabet, Σ_E, is the alphabet of the text to be hyphenated and the output alphabet, Γ_E, is $\Sigma_E \cup H$, where H is the set of symbols marking hyphenation points. For simple applications, this equals hyphens or discretionary (soft) hyphens $H = -$. For more fine-grained control over hyphenation, it is possible to use several different hyphens or weighted hyphens. Hyphenation of the word s is performed with the path automaton T_s by composing, $T_s \circ T_H$, which results in an acyclic path automaton containing a set of strings mapped to the hyphenated strings with weights $s : s^h_{<w_h>}$. Several alternative hyphenations may be correct according to the hyphenation rules. A conservative hyphenation algorithm should only suggest the hyphenation points agreed on by all the alternatives.

III. Material

In this article we present methods for converting the hunspell and TEX dictionaries and rule sets for use with open-source finite-state writer's tools. As concrete dictionaries we use the repositories of free implementations of these dictionaries and rule sets found on the internet, e.g. for the hunspell dictionary files found on the OpenOffice.org spell-checking site[5]. For hyphenation, we use the TEX hyphenation patterns found on the TEXhyphen page[6].

In this section we describe the parts of the file formats we are working with. All of the information of the hunspell format specifics is derived from the `hunspell(4)`[7] man page, as that is the only normative documentation of hunspell we have been able to locate. For TEX hyphenation patterns, the reference documentation is Frank Liang's doctoral thesis [1] and the TEXbook [7].

A. Hunspell File Format

A hunspell spell-checking dictionary consists of two files: a dictionary file and an affix file. The dictionary file contains only root forms of words with information about morphological affix classes to combine with the roots. The affix file contains lists of affixes along with their context restrictions and effects, but the affix file also serves as a settings file for the dictionary, containing all meta-data and settings as well.

The dictionary file starts with a number that is intended to be the number of lines of root forms in the dictionary file, but in practice many of the files have numbers different from the actual line count, so it is safer to just treat it as a rough estimate. Following the initial line is a list of strings containing the root forms of the words in the morphology. Each word may be associated with an arbitrary number of classes separated by a slash. The classes are encoded in one of the three formats shown in the examples of Figure 1: a list of binary octets specifying classes from 1–255 (minus octets for CR, LF etc.), as in the Swedish example on lines 2–4, a list of binary words, specifying classes from 1–65,535 (again ignoring octets with CR and LF) or a comma separated list of numbers written in digits specifying classes 1–65,535 as in the North Sámi examples on lines 6–8. We refer to all of

[5] http://wiki.services.openoffice.org/wiki/Dictionaries

[6] http://www.tug.org/tex-hyphen/

[7] http://manpages.ubuntu.com/manpages/dapper/man4/hunspell.4.html

```
 1  # Swedish
    abakus/HDY
 3  abalienation/AHDvY
    abalienera/MY
 5  # Northern Sámi
    okta/1
 7  guokte/1,3
    golbma/1,3
 9  # Hungarian
    üzér/1    1
11  üzletág/2        2
    üzletvezetö/3    1
13  üzletszerzö/4    1
```

Fig. 1. Excerpts of Swedish, Northern Sl-á-lmi and Hungarian dictionaries

these as continuation classes encoded by their numeric decimal values, e.g. ’abakus’ on line 2 would have continuation classes 72, 68 and 89 (the decimal values of the ASCII code points for H, D and Y respectively). In the Hungarian example, you can see the affix compression scheme, which refers to the line numbers in the affix file containing the continuation class listings, i.e. the part following the slash character in the previous two examples. The lines of the Hungarian dictionary also contain some extra numeric values separated by a tab which refer to the morphology compression scheme that is also mentioned in the affix definition file; this is used in the hunmorph morphological analyzer functionality which is not implemented nor described in this paper.

The second file in the hunspell dictionaries is the affix file, containing all the settings for the dictionary, and all non-root morphemes. The Figure 2 shows parts of the Hungarian affix file that we use for describing different setting types. The settings are typically given on a single line composed of the setting name in capitals, a space and the setting values, like the NAME setting on line 6. The hunspell files have some values encoded in UTF-8, some in the ISO 8859 encoding, and some using both binary and ASCII data at the same time. Note that in the examples in this article, we have transcribed everything into UTF-8 format or the nearest relevant encoded character with a displayable code point.

The settings we have used for building the spell-checking automata can be roughly divided into the following four categories: meta-data, error correction models, special continuation classes, and the actual affixes. An excerpt of the parts that we use in the Hungarian affix file is given in Figure 2.

The meta-data section contains, e.g., the name of the dictionary on line 6, the character set encoding on line 8, and the type of parsing used for continuation classes, which is omitted from the Hungarian lexicon indicating 8-bit binary parsing.

The error model settings each contain a small part of the actual error model, such as the characters to be used for edit distance, their weights, confusion sets and phonetic confusion sets. The list of word characters in order of popularity, as seen on line 12 of Figure 2, is used for the edit distance model. The keyboard layout, i.e. neighboring key sets, is specified for the substitution error model on line 10. Each set of the characters, separated by vertical bars, is regarded as a possible slip-of-the-finger typing error. The ordered confusion set of possible spelling error pairs is given on lines 19–27, where each line is a pair of a ‘mistyped’ and a ‘corrected’ word separated by whitespace.

The compounding model is defined by special continuation classes, i.e. some of the continuation classes in the dictionary or affix file may not lead to affixes, but are defined in the compounding section of the settings in the affix file. In Figure 2, the compounding rules are specified on lines 14–16. The flags in these settings are the same as in the affix definitions, so the words in class 118 (corresponding to lower case v) would be eligible as compound initial words, the words with class 120 (lower case x) occur at the end of a compound, and words with 117 only occur within a compound. Similarly, special flags are given to word forms needing affixes that are used only for spell checking but not for the suggestion mechanism, etc.

The actual affixes are defined in three different parts of the file: the compression scheme part on the lines 1–4, the suffix definitions on the lines 30–33, and the prefix definitions on the lines 35–37.

The compression scheme is a grouping of frequently co-occurring continuation classes. This is done by having the first AF line list a set of continuation classes which are referred to as the continuation class 1 in the dictionary, the second line is referred to the continuation class 2, and so forth. This means that for example continuation class 1 in the Hungarian dictionary refers to the classes on line 2 starting from 86 (V) and ending with 108 (l).

The prefix and suffix definitions use the same structure. The prefixes define the left-hand side context and deletions of a dictionary entry whereas the suffixes deal with the right-hand side. The first line of an affix set contains the class name, a boolean value defining whether the affix participates in the prefix-suffix combinatorics and the count of the number of morphemes in the continuation class, e.g. the line 35 defines the prefix continuation class attaching to morphemes of class 114 (r) and it combines with other affixes as defined by the Y instead of N in the third field. The following lines describe the prefix morphemes as triplets of removal, addition and context descriptions, e.g., the line 31 defines removal of ’ö’, addition of ’ős’ with continuation classes from AF line 1108, in case the previous morpheme ends in ’ö’. The context description may also contain bracketed expressions for character classes or a fullstop indicating any character (i.e. a wild-card) as in the POSIX regular expressions, e.g. the context description on line 33 matches any Hungarian vowel except a, e or ö, and the 37 matches any context. The deletion and addition parts may also consist of a sole ‘0’ meaning a zero-length string. As can be seen in the Hungarian example, the lines may also contain an additional number at the end which is used for the morphological analyzer functionalities.

```
AF 1263
AF VË–jxLnÓéè3ÄäTtYc,41 # 1
AF UmÖyiYcÇ # 2
AF ÖCWRÍ–jþÓíyÉÁÿYc2 # 3

NAME Magyar Ispell helyesírási szótár
LANG hu_HU
SET UTF–8
KEY öüó|qwertzuiopőú| # wrap
     asdfghjkléáűíyxcvbnm
TRY íóútaeslzánorhgkié # wrap
     dmyőpvöbucfjüyxwq–.á

COMPOUNDBEGIN v
COMPOUNDEND x
ONLYINCOMPOUND |
NEEDAFFIX u

REP 125
REP í i
REP i í
REP ó o
REP oliere oliére
REP cc gysz
REP cs ts
REP cs ds
REP ccs ts
# 116 more REP lines

SFX ? Y 3
SFX ? ö ős/1108 ö 20973
SFX ? 0 ös/1108 [^aáeéiíoóöőuüű] 20973
SFX ? 0 s/1108 [áéiíoóúőuúüű–] 20973

PFX r Y 195
PFX r 0 legújra/1262 . 22551
PFX r 0 legújjá/1262 . 22552
# 193 more PFX r lines
```

Fig. 2. Excerpts from Hungarian affix file

```
    \patterns{
 2     .ach4
       .ad4der
 4     .af1t
       .al3t
 6     .am5at
       f5fin.
 8     f2f5is
       f4fly
10     f2fy
    }
12  \hyphenation{
       as–so–ciate
14
       project
16     ta–ble
    }
```

Fig. 3. Excerpts from English TeX hyphenation patterns

B. TeX Hyphenation Files

The TeX hyphenation scheme is described in Frank Liang's dissertation [1], which provides a packed suffix tree structure for storing the hyphenation patterns, which is a special optimized finite-state automaton. This paper merely reformulates the finite-state form of the patterns, for the purpose of obtaining a general finite-state transducer version of the rules to be combined with other pieces of the finite-state writer's tools. In principle, the TeX hyphenation files are like any TeX source files, they may contain arbitrary TeX code, and the only requirement is that they have the 'patterns' command and/or the 'hyphenation' command. In practice, it is a convention that they do not contain anything else than these two commands, as well as a comment section describing the licensing and these conventions. The patterns section is a whitespace separated list of hyphenation pattern strings. The pattern strings are simple strings containing characters of the language as well as numbers marking hyphenation points, as shown in Figure 3. The odd numbers add a potential hyphenation point in the context specified by non-numeric characters, and the even numbers remove one, e.g. on line 8, the hyphen with left context 'f' and right context 'fis' would be removed, and a hyphen with left context 'ff' and right context 'is' is added. The numbers are applied in ascending order. The full-stop character is used to signify a word boundary so the rule on line 2 will apply to 'ache' but not to 'headache'. The hyphenation command on lines 13–16 is just a list of words with all hyphenation points marked by hyphens. It has higher precedence than the rules and it is used for fixing mistakes made by the rule set.

IV. METHODS

This article presents methods for converting the existing spell-checking dictionaries with error models, as well as hyphenators to finite-state automata. As our toolkit we use the free open-source HFST toolkit[8], which is a general purpose API for finite-state automata, and a set of tools for using legacy data, such as Xerox finite-state morphologies. For this reason this paper presents the algorithms as formulae such that they can be readily implemented using finite-state algebra and the basic HFST tools.

The lexc lexicon model is used by the tools for describing parts of the morphotactics. It is a simple right-linear grammar for specifying finite-state automata described in [4], [8]. The twolc rule formalism is used for defining context-based rules with two-level automata and they are described in [9], [8].

[8] http://HFST.sf.net

This section presents both a pseudo-code presentation for the conversion algorithms, as well as excerpts of the final converted files from the material given in Figures 1, 2 and 3 of Section III. The converter code is available in the HFST SVN repository[9] , for those who wish to see the specifics of the implementation in lex, yacc, c and python.

A. Hunspell Dictionaries

The hunspell dictionaries are transformed into a finite-state transducer language model by a finite-state formulation consisting of two parts: a lexicon and one or more rule sets. The root and affix dictionaries are turned into finite-state lexicons in the lexc formalism. The Lexc formalism models the part of the morphotax concerning the root dictionary and the adjacent suffixes. The rest is encoded by injecting special symbols, called flag diacritics, into the morphemes restricting the morpheme co-occurrences by implicit rules that have been outlined in [10]; the flag diacritics are denoted in lexc by at-sign delimited substrings. The affix definitions in hunspell also define deletions and context restrictions which are turned into explicit two-level rules.

The pseudo-code for the conversion of hunspell files is provided in Algorithm 1 and excerpts from the conversion of the examples in Figures 1 and 2 can be found in Figure 4. The dictionary file of hunspell is almost identical to the lexc root lexicon, and the conversion is straightforward. This is expressed on lines 1–1 as simply going through all entries and adding them to the root lexicon, as in lines 6—10 of the example result. The handling of affixes is similar, with the exception of adding flag diacritics for co-occurrence restrictions along with the morphemes. This is shown on lines 1—1 of the pseudo-code, and applying it will create the lines 17—21 of the Swedish example, which does not contain further restrictions on suffixes.

To finalize the morpheme and compounding restrictions, the final lexicon in the lexc description must be a lexicon checking that all prefixes with forward requirements have their requiring flags turned off.

B. Hunspell Error Models

The hunspell dictionary configuration file, i.e. the affix file, contains several parts that need to be combined to achieve a similar error correction model as in the hunspell lexicon.

The error model part defined in the KEY section allows for one slip of the finger in any of the keyboard neighboring classes. This is implemented by creating a simple homogeneously weighted crossproduct of each class, as given on lines 2–2 of Algorithm 2. For the first part of the example on line 10 of Figure 2, this results in the lexc lexicon on lines 11–18 in Figure 5.

The error model part defined in the REP section is an arbitrarily long ordered confusion set. This is implemented by simply encoding them as increasingly weighted paths, as shown in lines 2–2 of the pseudo-code in Algorithm 2.

Algorithm 1 Extracting morphemes from hunspell dictionaries

```
    $finalflags \leftarrow \epsilon$
 2: **for all** lines $morpheme/Conts$ in dic **do**
        $flags \leftarrow \epsilon$
 4:     **for all** $cont$ in $Conts$ **do**
            $flags \leftarrow flags + @C.cont@$
 6:         $Lex_{Conts} \leftarrow Lex_{Conts} \cup 0{:}[{<}cont]\ cont$
        **end for**
 8:     $Lex_{Root} \leftarrow Lex_{Root} \cup flags + morpheme\ Conts$
    **end for**
10: **for all** suffixes $lex, deletions, morpheme/Conts, context$ in aff **do**
        $flags \leftarrow \epsilon$
12:     **for all** $cont$ in $Conts$ **do**
            $flags \leftarrow flags + @C.cont@$
14:         $Lex_{Conts} \leftarrow Lex_{Conts} \cup 0\ cont$
        **end for**
16:     $Lex_{lex} \leftarrow Lex_{lex} \cup flags + [< lex] + morpheme\ Conts$
        **for all** $del$ in $deletions$ **do**
18:         $lc \leftarrow context + deletions$ before del
            $rc \leftarrow deletions$ after del $+ [< lex] + morpheme$
20:         $Twol_d \leftarrow Twol_d \cap del{:}0 \Leftrightarrow lc\ \_\ rc$
        **end for**
22:     $Twol_m \leftarrow Twol_m \cap [< lex] : 0 \Leftrightarrow context\ \_\ morpheme$
    **end for**
24: **for all** prefixes $lex, deletions, morpheme/conts, context$ in aff **do**
        $flags \leftarrow @P.lex@$
26:     $finalflags \leftarrow finalflags + @D.lex@$
        $lex \rightarrow prefixes$ {othewise as with suffixes, swapping left and right}
28: **end for**
    $Lex_{end} \leftarrow Lex_{end} \cup finalflags\ \#$
```

The TRY section such as the one on line 12 of Figure 2, defines characters to be tried as the edit distance grows in descending order. For a more detailed formulation of a weighted edit distance transducer, see e.g. [2]). We created an edit distance model with the sum of the positions of the characters in the TRY string as the weight, which is defined on lines 2–2 of the pseudo-code in Algorithm 2. The initial part of the converted example is displayed on lines 20–27 of Figure 5.

Finally to attribute different likelihood to different parts of the error models we use different weight magnitudes on different types of errors, and to allow only correctly written substrings, we restrict the result by the root lexicon and morfotax lexicon, as given on lines 1–9 of Figure 5. With the weights on lines 1–5, we ensure that KEY errors are always suggested before REP errors and REP errors before TRY errors. Even though the error model allows only one error of any type, simulating the original hunspell, the resulting transducer can be transformed into an error model accepting multiple errors by a simple FST algebraic concatenative n-closure, i.e. repetition.

[9] http://hfst.svn.sourceforge.net/viewvc/hfst/trunk/conversion-scripts/

```
LEXICON Root
        HUNSPELL_pfx ;
        HUNPELL_dic ;

! swedish lexc
LEXICON HUNSPELL_dic
@C.H@@C.D@@C.Y@abakus HDY ;
@C.A@@C.H@@C.D@@C.v@@C.Y@abalienation
        HUNSPELL_AHDvY ;
@C.M@@C.Y@abalienera MY ;

LEXICON HDY
 0:[<H]      H ;
 0:[<D]      D ;
 0:[<Y]      Y ;

LEXICON H
 er   HUNSPELL_end ;
 ers   HUNSPELL_end ;
 er   HUNSPELL_end ;
 ers   HUNSPELL_end ;

LEXICON HUNSPELL_end
@D.H@@D.D@@D.Y@@D.A@@D.v@@D.m@ # ;

 ! swedish twolc file
 Rules
 "Suffix H allowed contexts"
%[%<H%]: 0 <=> \ a _ e r ;
        \ a _ e r s ;
        a:0 _ e r ;
        a:0 _ e r s ;

 "a deletion contexts"
 a:0 <=> _ %[%<H%]:0 e r ;
        _ %[%<H%]: e r s ;
```

Fig. 4. Converted dic and aff lexicons and rules governing the deletions

C. TEX Hyphenation

The formulation of hyphenation as finite-state transducers is simple. We use the hyphenation alphabet $\Sigma_H = -$. To model the context-based deletions and additions of hyphenation patterns, we use twol rules with centers of $\epsilon : -$ for addition and $- : \epsilon$ for deletion. The algorithm for creating the rule sets described in Algorithm 3 simply goes through the patterns, and for each hyphenation point of each pattern extracts left and right context strings and adds them to the contexts of a rule. The result is exemplified in Figure 6. There is one rule for each of the hyphenation point numbers. The rules may be composed into one single transducer at compile time or applied as cascade at runtime.

The TEX hyphenation pattern also contains explicit exceptions to the hyphenation patterns, which are simply specific word forms with hyphenations, and can be compiled as simple paths: e.g. for the pattern{as-so-ciate} we create a path $as\epsilon so\epsilon ciate : as-so-ciate$

Algorithm 2 Extracting patterns for hunspell error models

```
for all neighborsets $ns$ in KEY do
    for all character $c$ in $ns$ do
        for all character $d$ in $ns$ such that $c! = d$ do
            $Lex_{KEY} \leftarrow Lex_{KEY} \cup c : d_{<0>}\#$
        end for
    end for
end for
$w \leftarrow 0$
for all pairs $wrong, right$ in REP do
    $w \leftarrow w + 1$
    $LEX_{REP} \leftarrow LEX_{REP} \cup wrong : right_{<w>}\#$
end for
$w \leftarrow 0$
for all character $c$ in TRY do
    $w \leftarrow w + 1$
    $Lex_{TRY} \leftarrow Lex_{TRY} \cup c : 0_{<w>}\#$
    $Lex_{TRY} \leftarrow Lex_{TRY} \cup 0 : c_{<w>}\#$
    for all character $d$ in TRY such that $c! = d$ do
        $Lex_{TRY} \leftarrow Lex_{TRY} \cup c : d_{<w>}\#$ {for swap: replace
        $\#$ with $cd$ and add $Lex_{cd} \cup d : c_{<0>}\#$}
    end for
end for
```

Algorithm 3 Extracting hyphenation patterns from TEX

```
for all patterns $p$ do
    for all digits $d$ in $p$ do
        $l, r \leftarrow$ split $p$ on $d$
        if $d$ odd then
            $l, r \leftarrow l, r << 0 : \epsilon$
            $Twol_d \leftarrow Twol_d \cap 0 : \epsilon \leftrightarrow l\_r;$
        else
            $l, r \leftarrow l, r << \epsilon : 0$
            $Twol_d \leftarrow Twol_d \cap \epsilon : 0 \leftrightarrow l\_r;$
        end if
    end for
end for
for all hyphenations $h$ do
    $word \leftarrow h - hyphens$
    $Lex_{exceptions} \leftarrow Lex_{exceptions} \cup$ word:h #
end for
```

V. Implementation and Tests

We have implemented the spell-checkers and hyphenators as finite-state transducers using program code and scripts with a Makefile. To test the code, we have converted 49 hyphenation pattern files and more than 42 hunspell dictionaries from various language families. They consist of the dictionaries that were accessible from the aforementioned web sites at the time of writing. The Tables I and II gives an overview of the sizes

```
LEXICON HUNSPELL_error_root
< ? > HUNSPELL_error_root ;
HUNSPELL_KEY "weight: 0" ;
HUNSPELL_REP "weight: 100" ;
HUNSPELL_TRY "weight: 1000" ;

LEXICON HUNSPELL_errret
< ? > HUNSPELL_errret ;
# ;

LEXICON HUNSPELL_KEY
ö:ü HUNSPELL_errret "weight: 0" ;
ö:ó HUNSPELL_errret "weight: 0" ;
ü:ö HUNSPELL_errret "weight: 0" ;
ü:ó HUNSPELL_errret "weight: 0" ;
ó:ö HUNSPELL_errret "weight: 0" ;
ó:ü HUNSPELL_errret "weight: 0" ;
! same for other parts

LEXICON HUNSPELL_TRY
í:0 HUNSPELL_errret "weight: 1" ;
0:í HUNSPELL_errret "weight: 1" ;
í:ó HUNSPELL_errret "weight: 2" ;
ó:í HUNSPELL_errret "weight: 2" ;
ó:0 HUNSPELL_errret "weight: 2" ;
0:ó HUNSPELL_errret "weight: 2" ;
! same for rest of the alphabet

LEXICON HUNSPELL_REP
í:i HUNSPELL_errret "weight: 1" ;
i:í HUNSPELL_errret "weight: 2" ;
ó:o HUNSPELL_errret "weight: 3" ;
oliere:olière HUNSPELL_errret "weight: 4" ;
cc:gysz HUNSPELL_errret "weight: 5" ;
cs:ts HUNSPELL_errret "weight: 6" ;
cs:ds HUNSPELL_errret "weight: 7" ;
ccs:ts HUNSPELL_errret "weight: 8" ;
! same for rest of REP pairs...
```

Fig. 5. Converted error models from aff file

```
"Hyphen insertion 1"
0:%− <=> # (0:%−) a (0:%−) f _ t ;
...
```

Fig. 6. Converted hyphenation models from TeXexamples

of the compiled automata. The size is given in binary multiples of bytes as reported by `ls -hl`.

In the hyphenation table, the second column gives the number of patterns in the rules. The total size is a result of composing all hyphenation rules into one transducer; it may be noted that both separate rules and a single composed transducer are equally usable at runtime. The separated version requires less memory whereas the single composed version is faster. For large results, such as the Norwegian[10] one, it may still be beneficial to keep the rules separated. In the Norwegian case, the four separately compiled rules are each of sizes between 1.2 MiB and 9.7 MiB.

For the hunspell automata in Table II, we also give the number of roots in the dictionary file and the affixes in affix file. These numbers should also help with identifying the version of the dictionary, since there are multiple different versions available in the downloads.

The resulting transducers were tested by hand using the results of the corresponding TeX `hyphenate` command and `hunspell -d` as well as the authors' language skills to judge errors. As testing material, a wikipedia article on the Finnish language[11] were used for most languages, and some arbitrary articles where this particular article was not found. In both tests, the majority of differences come from the lack of normalization or case folding. E.g. this resulted in our converted transducers failing to hyphenate words where uppercase letters would have been equal to their lowercase variants.

The hunspell model was built incrementally starting from the basic set of affixes and dictionary, and either adding or skipping all directive types of the file format as found in the wild. Some of the omissions show up e.g. in the English results, where omitting of the PHONE directive for the suggestion mechanism results in some of the differing suggestions in English tests, e.g. first suggestion for *calqued* in hunspell is *catafalqued*. Without implementing the phonetic folding, we get no results within 1 hunspell error, and get word forms like *chalked*, *caulked*, and so forth, within 2 hunspell errors. No other language has .aff files with PHONE rules, e.g. in French *comitatif* gets the suggestions *commutatif* and *limitatif* as the first ones in both systems.

VI. Conclusion

We have demonstrated a method and created the software to convert legacy spell-checker and hyphenation data to a more general framework of finite-state automata and used it in a real-life application. We have also referred to methods for extending the system to more advanced error models and the inclusion of other more complex models in the same system. We are currently developing a platform for finite-state based spell-checkers for open-source systems in order to improve the front-end internationalization.

The next obvious development for the finite-state spell checkers is to apply the unigram training [2] to the automata,

[10] both Nynorsk and bokmøal `input` the same patterns

[11] http://en.wikipedia.org/wiki/Finnish+language and its international links

TABLE I
COMPILED HYPHENATION AUTOMATA SIZES

Language	Hyphenator total	Number of patterns
Norwegian	978 MiB	27,166
German (Germany, 1996)	72 MiB	14,528
German (Germany, 1901)	66 MiB	14,323
Dutch	58 MiB	12,742
English (Great Britain)	38 MiB	8,536
Irish	20 MiB	6,046
English (U.S.)	19 MiB	4,948
Hungarian	15 MiB	13,469
Swedish	12 MiB	4,717
Icelandic	12 MiB	4,199
Estonian	8.8 MiB	3,701
Russian	4.2 MiB	4,820
Czech	3.1 MiB	3,646
Ancient Greek	2.4 MiB	2,005
Ukrainian	1.5 MiB	1,269
Danish	1.4 MiB	1,153
Slovak	1.1 MiB	2,483
Slovenian	939 KiB	1,086
Spanish	546 KiB	971
French	521 KiB	1,184
Interlingua	382 KiB	650
Greek (Polyton)	325 KiB	798
Upper Sorbian	208 KiB	1,524
Galician	160 KiB	607
Romanian	151 KiB	665
Mongolian	135 KiB	532
Finnish	111 KiB	280
Catalan	95 KiB	231
Greek (Monoton)	91 KiB	429
Serbian	76 KiB	2,681
Serbocroatian	56 KiB	2,681
Sanskrit	32 KiB	550
Croatian	32 KiB	1,483
Coptic	30 KiB	128
Latin	26 KiB	87
Bulgarian	24 KiB	1,518
Portuguese	19 KiB	320
Basque	15 KiB	49
Indonesian	14 KiB	46
Turkish	8 KiB	602
Chinese (Pinyin)	868	202

TABLE II
COMPILED HUNSPELL AUTOMATA SIZES

Language	Dictionary	Roots	Affixes
Portugese (Brazil)	14 MiB	307,199	25,434
Polish	14 MiB	277,964	6,909
Czech	12 MiB	302,542	2,492
Hungarian	9.7 MiB	86,230	22,991
Northern Sámi	8.1 MiB	527,474	370,982
Slovak	7.1 MiB	175,465	2,223
Dutch	6.7 MiB	158,874	90
Gascon	5.1 MiB	2,098,768	110
Afrikaans	5.0 MiB	125,473	48
Icelandic	5.0 MiB	222087	0
Greek	4.3 MiB	574,961	126
Italian	3.8 MiB	95,194	2,687
Gujarati	3.7 MiB	168,956	0
Lithuanian	3.6 MiB	95,944	4,024
English (Great Britain)	3.5 MiB	46,304	1,011
German	3.3 MiB	70,862	348
Croatian	3.3 MiB	215,917	64
Spanish	3.2 MiB	76,441	6,773
Catalan	3.2 MiB	94,868	996
Slovenian	2.9 MiB	246,857	484
Faeroese	2.8 MiB	108,632	0
French	2.8 MiB	91,582	507
Swedish	2.5 MiB	64,475	330
English (U.S.)	2.5 MiB	62,135	41
Estonian	2.4 MiB	282,174	9,242
Portugese (Portugal)	2 MiB	40.811	913
Irish	1.8 MiB	91,106	240
Friulian	1.7 MiB	36,321	664
Nepalese	1.7 MiB	39,925	502
Thai	1.7 MiB	38,870	0
Esperanto	1.5 MiB	19,343	2,338
Hebrew	1.4 MiB	329237	0
Bengali	1.3 MiB	110,751	0
Frisian	1.2 MiB	24,973	73
Interlingua	1.1 MiB	26850	54
Persian	791 KiB	332,555	0
Indonesian	765 KiB	23,419	17
Azerbaijani	489 KiB	19,132	0
Hindi	484 KiB	15,991	0
Amharic	333 KiB	13,741	4
Chichewa	209 KiB	5,779	0
Kashubian	191 KiB	5,111	0

and extend the unigram training to cover longer n-grams and real word error correction.

ACKNOWLEDGMENT

The authors would like to thank Miikka Silfverberg and others in the HFST research team as well as the anonymous reviewers for useful comments on the manuscript.

REFERENCES

[1] F. M. Liang, "Word hy-phen-a-tion by com-pu-ter," Ph.D. dissertation, Stanford University, 1983. [Online]. Available: http://www.tug.org/docs/liang/

[2] T. A. Pirinen and K. Lindén, "Finite-state spell-checking with weighted language and error models," in *Proceedings of the Seventh SaLTMiL workshop on creation and use of basic lexical resources for less-resourced languagages*, Valletta, Malta, 2010, pp. 13–18. [Online]. Available: http://siuc01.si.ehu.es/~jipsagak/SALTMIL2010_Proceedings.pdf

[3] L. A. Wilcox-O'Hearn, G. Hirst, and A. Budanitsky, "Real-word spelling correction with trigrams: A reconsideration of the mays, damerau, and mercer model," in *CICLing*, ser. Lecture Notes in Computer Science, A. F. Gelbukh, Ed., vol. 4919. Springer, 2008, pp. 605–616.

[4] K. R. Beesley and L. Karttunen, *Finite State Morphology*. CSLI publications, 2003.

[5] M. Silfverberg and K. Lindén, "Hfst runtime format—a compacted transducer format allowing for fast lookup," in *FSMNLP 2009*, B. Watson, D. Courie, L. Cleophas, and P. Rautenbach, Eds., 13 July 2009. [Online]. Available: http://www.ling.helsinki.fi/~klinden/pubs/fsmnlp2009runtime.pdf

[6] M. Mohri and M. Riley, "An efficient algorithm for the n-best-strings problem," 2002.

[7] D. Knuth, *The TeXbook*. Oxford Oxfordshire: Oxford University Press, 1986.

[8] K. Lindén, M. Silfverberg, and T. Pirinen, "Hfst tools for morphology—an efficient open-source package for construction of morphological analyzers," in *sfcm 2009*, ser. Lecture Notes in Computer Science, C. Mahlow and M. Piotrowski, Eds., vol. 41. Springer, 2009, pp. 28—47.

[9] K. Koskenniemi, "Two-level morphology: A general computational model for word-form recognition and production," Ph.D. dissertation, University of Helsinki, 1983. [Online]. Available: http://www.ling.helsinki.fi/~koskenni/doc/Two-LevelMorphology.pdf

[10] K. R. Beesley, "Constraining separated morphotactic dependencies in finite-state grammars." Morristown, NJ, USA: Association for Computational Linguistics, 1998, pp. 118–127.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 485–492

ISBN 978-83-60810-27-9
ISSN 1896-7094

The Polish Cyc lexicon as a bridge between Polish language and the Semantic Web

Aleksander Pohl
Computational Linguistics Department,
Jagiellonian University, Cracow, Poland
Email: aleksander.pohl@uj.edu.pl

***Abstract*—In this paper we discuss the problem of building the Polish lexicon for the Cyc ontology. As the ontology is very large and complex we describe semi-automatic translation of part of it, which might be useful for tasks lying on the border between the fields of Semantic Web and Natural Language Processing.**

We concentrate on precise identification of lexemes, which is crucial for tasks such as natural language generation in massively inflected languages like Polish, and we also concentrate on multiword entries, since in Cyc for every 10 concepts, 9 of them is mapped to expressions containing more than one word.

I. Introduction

THE FACT that linguistic resources play a key role in any Natural Language Processing undertaking is well established. Abstract theoretical problems such as word sense disambiguation and parsing as well as practical, such as machine translation, information extraction and question answering, are insolvable without large set of fine grained rules, large semantic dictionaries or huge collections of hand-annotated texts.

When a researcher works on a language having much less available resources than English, she always has to decide, whether to create them from scratch employing the best available techniques or to adopt some of the already available lexicons, ontologies, etc. As the adoption of the WordNet lexical database in the GlobalWordNet project shows, there is no obvious answer for this question.

Considering Polish, which is a language with a constantly growing set of linguistic resources (there are at least several complete or semi-complete Polish inflectional dictionaries, two growing WordNets and one large national corpus containing hand annotated samples of syntactic structures) one has to decide, whether it makes sense to wait for other researchers to complete their undertakings or to start the construction or adaptation of other resources.

Considering semantics, which is our primary field of interest, we have to agree, that the most advanced Polish resource is the Polish WordNet [1][1]. Since it is available for the Polish research community without restrictions and is created according to the state-of-the-art techniques of building WordNets, it doesn't make sense to spend time and money, on the creation of another, similar resource.

The Polish lexicon for Cyc [2], is a mapping between Cyc concepts and their Polish lexical representations. Since the mapping does not have to be isomorphic and each concept might have many mappings, the set of mappings for a give concept might be considered as a synset. What is more, the taxonomy of concepts in Cyc, in its structure, is quite similar to the taxonomy of WordNet synsets. At the first glance it seems, that the Polish lexicon is much similar to the Polish WordNet and, as a result, it seems to be a fruitless effort. Thus the question arises: what are the special properties of Cyc and what are the design goals of the Polish lexicon, which make the decision of creating it valuable?

II. Motivation

Our primary concern is to build algorithms and tools which bridge the gap between Polish language and the Semantic Web, thus bringing the benefits of the technology to the Polish speaking community.

Even though the fields of the Semantic Web and Natural Language Processing have much in common, there are certain problems, which have to be solved, before the data available in the Semantic Web and the data made available by NLP techniques is fully translatable. This stems from the fact, that the reference resources for the Semantic Web are ontologies, while the Princeton WordNet and its incarnations for languages other than English, serve as the *de facto* standard for NLP. Yet, there exist mappings between concepts of ontologies and WordNets (e.g. there is a mapping between Cyc and Princeton WordNet 2.0), but these mappings have certain limitation, stemming from the fact, that the logical structures of ontologies and WordNets is different.

The most problematic difference, in our opinion, is the huge discrepancy between the number and semantics of the types[2] of relations employed in both types of resources. In ontologies, the number of relations is not restricted *a priori* – it is only limited by the complexity of the domain of the ontology and by the desired level of detail. For instance, the old version of Dublin Core[3] defined 15 relations[4], while the latest defines

[1] Available at `http://plwordnet.pwr.wroc.pl/browser/?lang=en`.

[2] From here, by relation we mean both type of a relation and instance of a relation. We hope this inadequacy will not introduce ambiguities, since in most cases the types of relations are discussed.

[3] http://dublincore.org/documents/dcmi-terms/

[4] In RDF/OWL oriented ontologies the relations are always binary and are called properties.

approx. 50; the Music Ontology[5] defines approx. 120 relations, DBpedia[6] approx. 1200 and Cyc approx. 17000 relations[7].

On the other hand, most of the WordNets is created in accordance with the original Princeton WordNet idea refraining from using cross-part-of-speech relations. What is more, the set of relations was primarily limited to these, which were well accepted by the linguistic researchers community. Even though there are exceptions to these rules (e.g. there are cross-part-of-speech relations in the Polish WordNet), and there are plans and proposals to extend the set of relations (see [3]), it is unimaginable that the set of relations will grow to the size observed in moderately complicated ontologies.

To explain why we have to bother with that difference, let us consider a prototypical scenario, in which a music information extraction application utilizes data available both in the Semantic Web and made available by WordNet-based NLP algorithms. Let us assume, that the NLP module is able to fully disambiguate the common concepts (common nouns, verbs, adjectives, etc.) which appear in a certain text and the Semantic Web module is connected with an ontology containing massive amount of information about music[8]. The system should be able to answer questions such as „Have Tool already released the Ten thousand days album?", by parsing the question and consulting the database or recent press releases. However, it is unlikely, that the NLP module would recognize Tool as a name of a music group, and it is even less likely, that the phrase „Ten thousand days" would be recognized as a title of a music product, since the NLP dictionaries should capture general linguistic knowledge. But the biggest problem lies in the fact that the information is intransferable from the NLP module to the Semantic Web module – the former doesn't capture the relations between the release event (in which a music entity makes some music product available to the audience), the music entity and the music product. It might capture a notion of an event's actor and object, but such an information is too vague for the ontology.

We argue, that in such an application the NLP module should be designed in such a way, that the ontology contents is directly available in it. This is why we think, that building the Polish lexicon for Cyc, is worth its effort. The other advantages of using Cyc as the primary resource for NLP-enabled Semantic Web applications are as follows: there exists a Semantic Web endpoint which is linked to other Linked Open Data resources[9], it has probably the largest number of relations employed to describe the stored and processed knowledge, CycL – the language of Cyc is very expressive (e.g. allows for expressing relations between relations) and the ontology is shipped with an efficient inference engine, allowing for not only accessing, but also processing the knowledge in a consistent manner. And the last, but not the least, the relations in Cyc (and other ontologies) have formal definitions, which means, among others, that their arguments are restricted to concepts defined in the ontology (e.g. the first argument of the relation `#$weaponTypeCanDestroyTargetType` is restricted to `#$Weapon` and the second to `#$SolidTangibleThing`).

As it was stated, our primary concern is to bridge the gap between Polish language and the Semantic Web. Our final goal is to create a system, which is able to recognize ontological relations with their arguments in Polish texts, as well as, being able to produce well-formed Polish sentences, on the basis of the contents of the ontology. So, besides the adoption of a large number of relations provided by Cyc, we have to embrace the second important phenomenon – multi-word expressions. The reason why they are so common in ontologies stems from the fact, that the ontologies contain two types of entities, which are mostly represented by multi-word expressions: proper names and „artificial" concepts.

Proper names are the primary means for describing particular things, and we think that they are quite valuable, since they might be used to automatically pick training examples for the relations, we have to embrace them. The „artificial" concepts, are concepts which are used to properly structure the contents of the ontology – e.g. in Cyc there are concepts such as `#$Agent-Generic`, `#$Agent-PartiallyTangible` and `#$Agent-Underspecified`, which are used to capture certain properties of various types of agents. They shall not be mapped to the same word – *agens* – since that would introduce false ambiguity. It is better to provide descriptive, distinct mappings for these concepts (e.g. *agens*, *agens materialny*, *uogólniony agens*), but multi-word expressions are indispensable here. This is why we pay special attention to the multi-word expressions.

III. Related work

In our work we use the transfer approach to translate the compound expressions. On the other hand there has been much research in the field of statistical machine translation of these expressions (see [4]) and there are commercial machine translation systems available, like Google Translate[10]. The problem we encountered, refraining us from using this approach, was the lack of resources – we didn't find any publicly available aligned English-Polish corpus, while using resources such as Wikipedia seemed to be too demanding for this task. What is more – direct application of Google Translate (which is discussed later) was too erroneous.

IV. Methodology

A. Goals

As it was stated in the Motivation section, our primary goal is to bridge the gap between Polish language, and the Semantic Web, using the Cyc ontology as the primary resource, by providing a tool which is capable of recognizing its relations in Polish texts, and by generating Polish paraphrases for them. The first step to achieve this goal is to build the Polish lexicon for the ontology. At the first glance, it seems that we should

[5] http://musicontology.com/

[6] http://wiki.dbpedia.org/Ontology?v=zj4

[7] ResearchCyc, system: 10.126767, KB: 7141, http://research.cyc.com

[8] e.g. http://dbtune.org/musicbrainz/

[9] http://sw.opencyc.org

[10] http://translate.google.com

build a full lexicon, that is a lexicon covering all symbols available in Cyc, but we think that this is not needed. The proposed algorithm would utilize the definitions of relations, the argument constraints in particular. It appears, that only approx. 4 thousands of concepts are used as the argument constraints and we propose to translate only *these* symbols (it might appear in the future, that translating more symbols, would bring better extraction results).

This assumption seems to be an oversimplification – even though we would be able to train the algorithm to recognize these relations with these concepts as their arguments on the basis of that mapping, we won't be able to recognize other concepts. E.g. we would be able to recognize the `#$releases-Underspecified` relation, in a sentence „Zespół wydał nową płytę" („A music group released a new album"), assuming that „music group" and „album" are the argument constrains of that relation, but we won't be able to recognize it in a sentence like „Tool wydał wczoraj CD Ten thousand days" („Tool released the Ten thousand days CD yesterday"), since „Tool", „Ten thousand days" and „CD" won't be recognized as the proper specializations of „music group", „album title", etc.

We agree, that this is a problem, but we won't resolve it by translating the full Cyc taxonomy. Instead, we are going to use the results of a project aiming at the extraction of the hyperonymy relation from the Polish Wikipedia, which was carried out in our research group [5]. In short, the results cover several hundreds of thousands of concepts, grouped within several thousands of semantic categories.

The particular goals we are going to achieve are as follows:

1) create translation for all the concepts which are used as the argument constraints in the Cyc relations (approx. 4 thousands)
2) map these concepts to the semantic categories extracted from the Polish Wikipedia

Achieving these two goals would allow us to:

1) automatically pick training examples for the Cyc relations
2) build linguistic models of these relations
3) build algorithms extracting these relations from Polish texts

The text generation feature of the designed system is not covered in this document, but the prototype applications utilize it.

B. The algorithm

The general algorithm for building the Polish lexicon is as follows – for each Cyc concept which is used as an argument constraint:

1) *translate* the English mapping of the concept into Polish (many results might be produced)
2) *map* the words of each translation to the entries of Polish inflectional dictionary
3) *transform* the translations to match syntax constraints
4) *rank* the translations
5) *present* the results to the human operator
6) *store* the selected result in the database
7) *search* for semantic categories extracted from the Polish Wikipedia, corresponding to the translation
8) *merge* or *link* the selected categories with the Cyc concept

C. Translation

The first necessary step in the creation of the Polish lexicon, is the translation or pairing of lexical units. This might be done by utilization of (in a transfer approach) a machine readable English-Polish dictionary or (in a statistical approach) a bilingual corpus. The latter approach is quite popular in the on-line translation systems, such as Google Translate and it has certain advantages – namely the translation algorithm is generic and the bilingual dictionary is not needed. Still, it seems that this approach is not well suited for the taxonomy translation task. It is due to the fact, that in most cases, the available training bilingual corpuses cover only texts containing regular sentences, having at least the SVO structure, while in most cases the Cyc concepts are described as a mere nominal expressions. What is more, the obtained translation should be canonical, that is, the head phrase should be in singular[11] and nominal case.

This prediction was verified on the Google Translate system. Out of 118 Cyc concepts, only 22 were translated exactly the same as by the human translator. 20 of them had certain syntax errors (e.g. „tangible agent" – „rzeczowe agent" where the adjective does not agree with the noun on case and gender), 40 of them had certain translation errors (e.g. „acquiring" – „przejmującej" where the concept denotes an event, while the translation is an adjective, thus a property), 64 of them were not in a canonical form (e.g. „animal" – „zwierząt" where the translation is in plural and in dative case, while it should be in singular and in nominal case), and 62 were translated differently, due to the general design principles of the lexicon (avoidance of ambiguities, among the others)[12].

This is why we choose the transfer approach, based on a large machine readable English-Polish dictionary „Wielki Słownik Multimedialny Polsko-Angielski/ Angielsko-Polski Oxford-PWN". Although the information which is available in such a dictionary is lexically rich – it signals the grammatical category of the entries, includes limited syntactical, semantical and pragmatical information – these features are not provided consequently and it is really hard to obtain precise mapping between Cyc concepts and the dictionary entries on the one hand, and translated entries and Polish inflectional dictionary entries on the other hand.

The translation strategy is as follows – when we translate some Cyc concept, which is represented by S_i^{en} character string, there might be the following general cases:

1) The character string is a single word, which *is not present* in the dictionary – we try to apply some trans-

[11] or plural for *plurale tantum* nouns

[12] The number of errors does not sum to 118, since one translation could be marked as invalid more than once.

formation to it, such as stemming, but if the result is not present as well, we have to ignore it. If it is present, this situation is reduced to the next one.

2) The character string is a single word, which *is present* in the dictionary – we pass the list of translations $(S^{pl}_{i,1}, S^{pl}_{i,2}, S^{pl}_{i,3}, \ldots)$ to the next step of the algorithm.
3) The character string is a multi-word expression, which *has direct* representation in the dictionary – since it seems to be a compound expression, we process it if it was a single word – pass the whole list $(S^{pl}_{i,1}, S^{pl}_{i,2}, S^{pl}_{i,3}, \ldots)$ to the next step.
4) The character string is a multi-word expression, which *doesn't have direct* representation in the dictionary – we divide the S^{en}_i string into single words: $W^{en}_{i,1}, W^{en}_{i,2}, W^{en}_{i,3}, \ldots$, which might be represented by the following character strings $S^{en}_k, S^{en}_l, S^{en}_m, \ldots$ in the dictionary. Then we remove stop words (such as determiners or prepositions) form the list. For each element of the resulting list we take the corresponding Polish strings and create a vector of lists, where each position is occupied by the corresponding translations, and order of the list reflects the order of the source words: $\left[(S^{pl}_{k,1}, S^{pl}_{k,2}, \ldots)_1, (S^{pl}_{l,1}, \ldots)_2, (S^{pl}_{m,1}, \ldots)_3, \ldots\right]$. The lower index attached to parentheses indicates the position of the source word in the source expression. This vector is passed to the next step of the algorithm.

To sum-up – the translation of a Cyc concept produces a vector of Polish words or lists of Polish words, and since a single word might be considered as a single-entry list, we might simplify the description, by assuming, that always a vector of lists containing Polish words is produced, where each element of the vector corresponds to one word in the English mapping of the concept, and each element of the list corresponds to one possible translation of the word.

For instance: if we translate `#$AddictiveSubstance`, which is mapped to the `addictive substance` expression in English, we might receive the following result: `[(uzależniający, wciągający)`$_1$`,(substancja, istota, ciężar, waga, podstawa, treść, realność, majątek)`$_2$`]`

D. Mapping to inflectional dictionary

Since the next step of the algorithm will transform the obtained translations to match the syntax constrains of the Polish language, the result of the previous step has to be mapped to Polish inflectional dictionary, such as one described in [6] or [7]. This dictionary should have at least two functionalities:

1) lemmatization – recognition of the lemma based on any of the inflected forms of a lexeme
2) inflection – production of an inflected form based on a provided set of tags

Since the first functionality might introduce ambiguity (e.g. the character string *goli* is an inflected form of lexemes having the following lemmas: *gol* (goal), *golić* (to shave), *golić się* (to shave oneself), *goły* (naked), *Gola* (a Polish surname) and *Goły* (a Polish surname)), for each character string $S^{pl}_{i,j}$ we might receive many lemmas:

$$S^{pl}_{i,j} \to \left[L^{pl}_a, L^{pl}_b, \ldots\right]_{i,j} \qquad (1)$$

where L^{pl}_a stands for the lexeme with an index a. The i, j indices indicate, that given vector corresponds to the $S^{pl}_{i,j}$ Polish character string.

The indexing of the lexemes in the dictionary needs some special attention – in general we would like to avoid the situation in which human interpretation of each lexeme requires looking it up in the index, so the lexeme should be at least represented by its lemma.

As it is discussed in detail in [8], there are no better means of differentiating the lexemes with the same lemma and different inflection, than by introducing some arbitrary marking of the homonymuous lemmas. [8] proposes numbering of the lemmas, while we think that the approach proposed in [7], namely to attach an inflectional label to each lemma, is better, since, provided that the person who looks at the mapping, knows the labeling system, doesn't have to check the ordering of these lemmas to determine, what is the inflectional paradigm of the lexeme, represented by the $< lemma, inflectional\ label >$ pair[13].

In the system described in [7] the inflectional label consist of capital letters and is constructed in such a way, that the most significant morphological distinctions are placed at the beginning of the label, thus the first letter determines the grammatical category of the lexeme (A – noun, B – verb, etc.) the second letter (in the case of nouns) determines their gender and so on. This idea has another advantage, that only the lemma, the inflectional label and the inflectional scheme is necessary to produce all the forms of a given lexeme, without the direct intervention of the dictionary, so it's easier to port across operating systems and versions of the dictionary.

In fact, the label could be replaced by a number, but the most important difference between theses systems is that, in the first case, the index indicates the position of the lexeme among other lexemes with the same lemma, while in the second case, it indicates position of its inflectional paradigm among other inflectional paradigms. This means that in the second case, the index is much less likely to change.

Since we assumed that the output produced by the previous step is always a vector, the result of this step is a vector of mappings obtained by merging the vectors produced for a given position in given single or multi-word entry:

$$\left[(S^{pl}_{k,1}, S^{pl}_{k,2}, \ldots)_1, (S^{pl}_{l,1}, \ldots)_2, (S^{pl}_{m,1}, \ldots)_3, \ldots\right] \to \qquad (2)$$

$$\left[\left([L^{pl}_a, L^{pl}_b, \ldots]_{k,1}, [L^{pl}_c, \ldots]_{k,2}, \ldots\right)_1, \left([L^{pl}_d, \ldots]_{l,1}, \ldots\right)_2, \left([L^{pl}_e, \ldots]_{m,1}, \ldots\right)_3, \ldots\right] \to \qquad (3)$$

[13] We have to mention, that in our formalization of lexemes $L^t_a, L^t_b, \ldots$ we keep abstract indices a, b, which should be interpreted as distinct $< lemma, inflectional\ label >$ pairs.

$$\Big[(L_a^{pl}, L_b^{pl}, L_c^{pl}, \ldots)_1, (L_d^{pl}, \ldots)_2, (L_e^{pl}, \ldots)_3, \ldots\Big] \quad (4)$$

where the vector $[L_a^{pl}, L_b^{pl}, \ldots]_{k,1}$ is merged with the vector $[L_c^{pl}, \ldots]_{k,2}$ producing the list present at the first position of the equation 4.

We have to mention that, some of the elements of the character string lists might be removed completely, when they are not recognized by the inflectional dictionary. Due to the productive character of languages, there are always some words, which are missing in such dictionaries. For example, the latest version of the dictionary described in [7] doesn't recognize forms such as *konfigurować* (configure) or *opcjonalny* (optional), which are recognized by modern general purpose Polish dictionaries such as the online version of the most popular Polish dictionary accessible on the website `http://sjp.pwn.pl`[14].

We also have to say that there are character strings, which are not recognized by the dictionary due to the fact, that they are multi-word expressions. It's because some English words might be translated as Polish multi-word expressions. In such a case we have two options: to ignore theme or to split them into single words. In our case we took the first approach in cases, when given Polish translation corresponded to one word in an English multi-word expression and the second approach in the opposite cases.

Considering the example from the previous step, the vector would be translated into the following result: `[(<uzależniać,BDA>, <uzależniać się,BDA>, <uzależniający,CAA>, <uzależniający się,CAA>, <wciągać,BDA>, <wciągać się,BDA> <wciągający,CAA>, <wciągający się,CAA>)`$_1$`, (<substancja,ADACBAA>, <istota,ADAAA>, <ciężar,ACAAAAA>, <Ciężar,AAAAD>, <waga,ADAB>, <Waga,AABACC>, <podstawa,ADAAA>, <treść,ADCCA>, <realność,ADCCA>, <majątek,ACABA>)`$_2$`]`

E. Transformation

The mapping step might produce tens or even hundreds of interpretations for a single Cyc concept, thus some transformations have to be applied, to reduce these numbers. There is also another problem, which stems from the fact, that for the translation to be consistent, certain features of the lexemes (such as gender of a noun and an adjective) have to be accommodated.

The general idea of the transformation step, is to look at the grammatical categories of the lexemes corresponding to the Polish character strings. It is observed, that for resources such as Cyc, many of the source entities were two-word expressions or three-word expressions containing a preposition or a determiner. Because determiners are not present in Polish, and preposition often disappear in the translation (e.g. „of" is replaced by the genitive case of the dependent nominal phrase), we had to deal with restricted number of grammatical category combinations and it was quite easy to order them in an effective, yet not much restricting manner:

1) noun + adjective
2) noun + noun
3) noun + verb
4) noun + other
5) other

Having such an ordering, the lexeme tuples taken from the Cartesian product of the vector from the equation 4, were partitioned into five sets and only the non-empty partition with the highest rank was selected. All the other lexeme pairs were dismissed. We haven't defined rules for triples of lexemes, since it turned out, that such complex expressions were rarely translated correctly, and they were processed rather slowly.

After this reduction, the inflectional forms of the lexemes were adjusted to fulfil Polish syntactic rules. For the first 3 cases the schema was as follows:

1) noun + adjective: the base form tagging for the noun was determined, which could be a nominal case of a singular or a plural form (the latter for *plurale tantum* nouns), then the form of the adjective was selected accordingly (its number, case and gender being taken directly from the noun form and its gender).
2) noun + noun: for each of the nouns the number was determined as in the previous rule. Then two pairs of forms where added: in the first, a nominal case for the first and genitive case for the second noun was selected, in the second – a genitive case for the first and nominal for the second lexeme[15].
3) noun + verb: the infinitive form of the verb and the accusative case[16] for the noun were selected.

In the other cases only the base forms were selected.

When these transformations had been completed the number of lexemes' pairs were significantly reduced. But what is even more important, for most of the cases, the obtained expressions were grammatically correct.

This step would transform example from the previous step as follows: `[uzależniająca substancja, uzależniająca się substancja, wciągająca substancja, wciągająca się substancja, uzależniająca istota, uzależniająca się istota, wciągająca istota, wciągająca się istota, uzależniający ciężar, uzależniający się ciężar, wciągający ciężar, wciągający się ciężar, ...]`

F. Ranking

When the transformation step is finished, the translations are ranked according to the number of their occurrences in a large corpus.

[14] Checked on the 7^{th} of March 2010.

[15] This idea was supported by the fact, that in nominal phrases consisting of two nouns, the subordinate noun, always has the genitive case, e.g. *wąsy kota* (whisker of a cat) – `plural:nominal singular:genitive`. In other words: noun governs genitive case.

[16] Since the syntactic features of Polish verbs are not yet fully described in a form of an electronic dictionary, we've taken this assumption, although it could produced many incorrect translations.

In general three different cases might appear as the result of the previous step:

1) there are only single words in the vector
2) there are pairs of lexemes produced by the transformation step
3) there is the original vector from the equation 4, if it contained more than two positions

In the first case, the results are ranked simply according to the number of their occurrences in the corpus. It is quite important, that there is one big lemmatized corpus of Polish available for free – the IPI PAN corpus described in [9].

In the second case, the results are ranked according to the number of occurrences of bigrams in the corpus. Even though the IPI PAN corpus is not well balanced, the mere fact that given pair of words occurs in it, is sufficient to properly order them and reject uncommon multi-word translations. In general case, if ngrams were employed in the transformation step, any kind of smoothing or backoff algorithm could be employed (see [10, pages 83–122] for details).

In the last case, when the complex transformation was not applied to the original vector, the translations should be ranked as in the first case, but separately for each position. It is not practical to check the number of occurrences of each combination of the character strings, as it might be really huge, and would significantly slow down the translation process (while the whole methodology is devised for its speed-up). Nevertheless, the ordering of translations for each position is still important, since it provides the human operator with translation hints and also signifies, which translation might be the most natural.

It might seem that the IPI PAN corpus is too small for such a task, and tools such as search engine should be consulted. In practice, this is not the best idea, since the number of queries which would have to be send to the server is quite large (at least tens for single concept), and as the tool is designed for interactive usage, this would slow down the process – simple looking through the list of results would be more efficient[17].

For the example provided above, only the „substancja uzależniająca" is recorded in the corpus and only this proper translation is presented to the user.

G. Selection

When the translations are ranked they might be presented to the human operator. If the number of translations is too large, they might be cut at some level (e.g. only 15 top ranked translations appear), since it doesn't make sense to go through all of theme (this might be more time consuming than figuring out the translation from scratch).

The selection should be as easy as clicking a button next to the correct translation. We think that the user should not be disrupted by the precise lexical information at the moment, so simple character strings should be displayed. This might introduce some ambiguity, but since the user always should have the option to enter the translation manually[18] the interpretation step is necessary anyway.

H. Interpretation

The last necessary step in creating the accurate translation is the correct interpretation of the character string selected by the user. Although if he selects one of the translations that was suggested by the system, the necessary information is available at hand. But when he enters the translation manually, the correct lexemes and taggings should be selected.

This is done with a support of a simple parsing algorithm. There is not enough space to discuss it in detail, that's why we give just its simple characteristic. In general the algorithm is based on the concept of unification, with features attached to the grammatical categories [10, pages 489–528]. Each grammatical category defines what values of features are required from the other grammatical categories if their instances are subordinate elements of the instances of the former category in the abstract syntax tree (e.g. genitive case for the noun which is a subordinate of some other noun). Some of the categories are supplemented with the information, that their instances require obligatory subordinate elements (like reflexive pronoun *się* for reflexive verbs). For given interpretation of the expression – if there exist tree of nodes constructed according to the optional requirements, and for each node all of its obligatory requirements are met, the interpretation is marked as valid.

Still, although for simple expressions, this algorithm produces many unambiguous results, there are cases[19], for which even the most clever algorithm would produce more than one interpretation. So in such cases, the human operator should be able to select the correct lexemes and taggings by hand.

As a final result, the Cyc concept is mapped to the list (in the simplest case – single entry list) of Polish lexemes with taggings[20] attached:

$$L_i^s \to \left[(L_a^t, T_{a,k})_1, (L_b^t, T_{b,l}), \ldots\right] \tag{5}$$

I. Searching for semantic categories

Since we have the proper mapping of the Cyc concept, we might search for the semantic categories extracted from the Polish Wikipedia, which are most similar to the translation. So far this algorithm is not much complicated – all the entires which contain the lexemes which appear in the translation are selected, and then they are ranked according to the following equation:

$$R_i = \frac{cm_{i,j}}{cl_i} * \frac{cm_{i,j}}{cl_j} * children_i \tag{6}$$

where

[17] On the other hand, if the results were cached, the search-engine approach would be much better and it is considered for the further development of the system.

[18] The rationale is that there are many cases, like compound metaphoric expressions not recorded in the bilingual dictionary, or entries containing many words (in our case, more than two) that will never appear as the result of the complex processing.

[19] E.g. the lexemes with lemma *zamek* differentiated by the `singular:genitive` *zamka:zamku* or the expression *akt własności*, where the second lexeme might be in singular or in plural.

[20] Indicating the selected forms.

1) R_i – is the rank of semantic category i
2) $cm_{i,j}$ – is the number of common lexemes in the semantic category i and the translation of the Cyc concept j
3) cl_i – is the number of lexemes in the name of the semantic category i
4) cl_j – is the number of lexemes in the translation of Cyc concept j
5) $children_i$ – is the number of instances in the semantic category i

This equation favors categories with many instances on the one hand, and also the categories, whose names are similar to the translation of the concept on the other.

J. Merging and linking the categories

As the last step of the algorithm the user might make the following decisions:

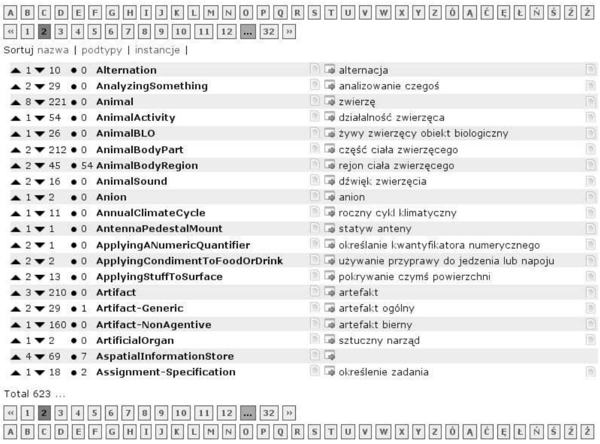
1) She might merge zero or more semantic categories with the Cyc concept. The result of the operation will be a direct attachment of all the instances of the category to the Cyc concept – the instances of the category will become the instances of the concept. E.g. the „miasto" (city) category might be merged with the `#$City` concept.
2) She might link zero or more semantic categories with the Cyc concept. In this case the category will become a specialization of the concepts, and its instances will be treated as indirect instances of the concept. E.g. the „miasto wojewódzkie" (provincial city) might be linked with the `#$City` concept.

The user is not restricted to merging one category, due to the fact, that the semantic category extraction algorithm produces many over specified categories (e.g. „miasto położone" (city situated), „miasto znajdujące" (city situated), which should be merged with the `#$City` concept).

Even though the previous step might produce many results, it is not needed to merge or link all of them – the less instances given semantic category contains, the less time should be spend for its analysis and some of them might be simply skipped.

V. Application

As a part of the research we constructed an application[21], along the lines of the described methodology. In our earlier research we found out, that although rough automatic translation of single-word entries produces promising results, the syntactic information, which is indispensable for the natural language production capability, has to be entered by the human operator. What is even more important, most of the entries which represent concepts of the ontology are multi-word expressions, which are mistakenly translated by the leading statistics based machine translation systems, like the above mentioned Google Translate.

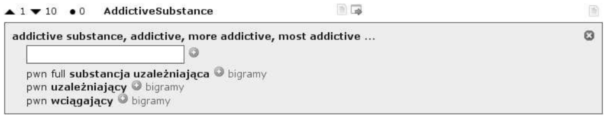

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ó Ą Ć Ę Ł Ń Ś Ź Ż
« 1 2 3 4 5 6 7 8 9 10 11 12 ... 32 »
Sortuj nazwa | podtypy | instancje |

▲ 1	▼ 10	● 0	Alternation	alternacja
▲ 2	▼ 29	● 0	AnalyzingSomething	analizowanie czegoś
▲ 8	▼ 221	● 0	Animal	zwierzę
▲ 1	▼ 54	● 0	AnimalActivity	działalność zwierzęca
▲ 1	▼ 26	● 0	AnimalBLO	żywy zwierzęcy obiekt biologiczny
▲ 2	▼ 212	● 0	AnimalBodyPart	część ciała zwierzęcego
▲ 2	▼ 45	● 54	AnimalBodyRegion	rejon ciała zwierzęcego
▲ 2	▼ 16	● 0	AnimalSound	dźwięk zwierzęcia
▲ 1	▼ 2	● 0	Anion	anion
▲ 1	▼ 11	● 0	AnnualClimateCycle	roczny cykl klimatyczny
▲ 1	▼ 1	● 0	AntennaPedestalMount	statyw anteny
▲ 2	▼ 1	● 0	ApplyingANumericQuantifier	określanie kwantyfikatora numerycznego
▲ 2	▼ 2	● 0	ApplyingCondimentToFoodOrDrink	używanie przyprawy do jedzenia lub napoju
▲ 2	▼ 13	● 0	ApplyingStuffToSurface	pokrywanie czymś powierzchni
▲ 3	▼ 210	● 0	Artifact	artefakt
▲ 2	▼ 29	● 1	Artifact-Generic	artefakt ogólny
▲ 1	▼ 160	● 0	Artifact-NonAgentive	artefakt bierny
▲ 1	▼ 2	● 0	ArtificialOrgan	sztuczny narząd
▲ 4	▼ 69	● 7	AspatialInformationStore	
▲ 1	▼ 18	● 2	Assignment-Specification	określenie zadania

Total 623 ...

« 1 2 3 4 5 6 7 8 9 10 11 12 ... 32 »
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ó Ą Ć Ę Ł Ń Ś Ź Ż

Fig. 1. Main window of the application.


▲ 1 ▼ 10 ● 0 AddictiveSubstance

addictive substance, addictive, more addictive, most addictive ...

pwn full **substancja uzależniająca** bigramy
pwn **uzależniający** bigramy
pwn **wciągający** bigramy

Fig. 2. Translations suggested by the application.

The main window of the system is presented in Fig. 1 and it shows part of the list of concepts which are mapped to Polish expressions (the concept is on the left, while the translation is on the right). The user might get familiar with the meaning of the concept, by studying its hypernyms, hyponyms and directed instances (icons on the left) and by reading its comment (the left icon in the middle between the concept and its translation).

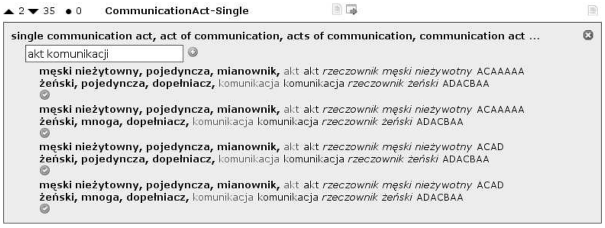
When he clicks the right icon in the middle of the table, he sees the translations suggested by the system (Fig. 2). The English mappings of the concept are present at the top of the translation box. The manual entry box and the suggested translations are below. The user might accept given suggestion by clicking the plus icon next to it or enter some other translation in the manual entry box. He might also consult statistics of given expression by clicking the `bigramy` link next to it.

In the rare case, that given expressions is morphosyntacticly ambiguous, the user is consulted once again (Fig. 3). The full tagging of each lexeme is presented, as well as the morphological information in a form of an inflectional label. If the user is not familiar with the inflectional scheme, he might click the lemma of the lexeme, and its full inflectional paradigm will be presented. The user accepts given interpretation by clicking the tick icon, which is below.

The user might validate his selection by clicking the icon which is on the left of the translation (Fig. 1). It will show Polish paraphrasing of sample of the taxonomical knowledge taken from Cyc (Fig. 4), which utilizes the morphosyntactic information attached to the translation.

Then the user might search among the semantic categories extracted from the Polish Wikipedia, to find these which could

[21] The demo of the application is available under the URL `http://klon.wzks.uj.edu.pl/cycdemo`.



▲ 2 ▼ 35 ● 0 **CommunicationAct-Single**

single communication act, act of communication, acts of communication, communication act ...

akt komunikacji

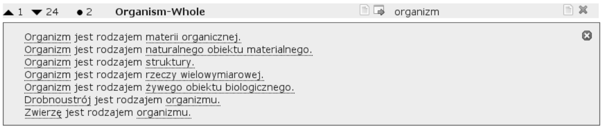
męski nieżytowny, pojedyncza, mianownik, akt akt *rzeczownik męski nieżywotny* ACAAAAA
żeński, pojedyncza, dopełniacz, komunikacja komunikacja *rzeczownik żeński* ADACBAA

męski nieżytowny, pojedyncza, mianownik, akt akt *rzeczownik męski nieżywotny* ACAAAAA
żeński, mnoga, dopełniacz, komunikacja komunikacja *rzeczownik żeński* ADACBAA

męski nieżytowny, pojedyncza, mianownik, akt akt *rzeczownik męski nieżywotny* ACAD
żeński, pojedyncza, dopełniacz, komunikacja komunikacja *rzeczownik żeński* ADACBAA

męski nieżytowny, pojedyncza, mianownik, akt akt *rzeczownik męski nieżywotny* ACAD
żeński, mnoga, dopełniacz, komunikacja komunikacja *rzeczownik żeński* ADACBAA

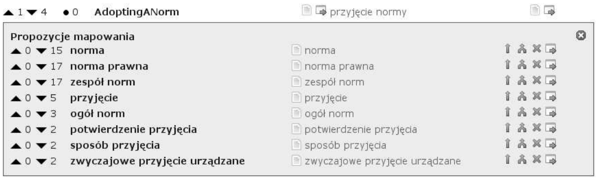
Fig. 3. Morphosyntactic ambiguity.



▲ 1 ▼ 24 ● 2 **Organism-Whole** organizm

Organizm jest rodzajem materii organicznej.
Organizm jest rodzajem naturalnego obiektu materialnego.
Organizm jest rodzajem struktury.
Organizm jest rodzajem rzeczy wielowymiarowej.
Organizm jest rodzajem żywego obiektu biologicznego.
Drobnoustrój jest rodzajem organizmu.
Zwierzę jest rodzajem organizmu.

Fig. 4. Polish paraphrases of sample of the Cyc knowledge.



▲ 1 ▼ 4 ● 0 **AdoptingANorm** przyjęcie normy

Propozycje mapowania

▲ 0 ▼ 15 **norma** norma
▲ 0 ▼ 17 **norma prawna** norma prawna
▲ 0 ▼ 17 **zespół norm** zespół norm
▲ 0 ▼ 5 **przyjęcie** przyjęcie
▲ 0 ▼ 3 **ogół norm** ogół norm
▲ 0 ▼ 2 **potwierdzenie przyjęcia** potwierdzenie przyjęcia
▲ 0 ▼ 2 **sposób przyjęcia** sposób przyjęcia
▲ 0 ▼ 2 **zwyczajowe przyjęcie urządzane** zwyczajowe przyjęcie urządzane

Fig. 5. The semantic categories which are proposed for merging and linking.

be merged or linked with the concept (Fig. 5). This is done by clicking the icon on the right of the translation.

The user might merge the category with the concept, by dragging the splitted arrow and dropping it on the name of the concept. The user might also link the category with the concept, by dragging the straight arrow and dropping it on the name of the concept.

VI. Results

So far 560[22] of Cyc concepts out of 3600 selected for translations were mapped to the Polish expressions. The translations were carried out by two independent translators, reaching the inter-translator agreement of 56%. This means that the translation task is not easy. This is due to the fact, that the concepts selected for translations are quite general, and have to be translated carefully. The general precision[23] of the translations suggested by the system was 37% and recall[24] was 88%, while precision for two-word compounds was 27%.

On the other hand 193 of the translated concepts were linked and merged with Wikipedia semantic categories, covering more than 20000 Wikipedia instances (articles), which seems to be a good result for the very limited resources available for the project.

VII. Conclusions

Although the precision of the translation algorithm is quite low, the application speeds-up the creation of the Polish lexicon for Cyc. This is due to the fact, that it integrates several resources, namely the Cyc ontology, Polish inflectional dictionary, English-Polish dictionary as well as semantic categories and concepts extracted from Wikipedia, while presenting to the user only these pieces of information, which are relevant for the task. As a result, after few months (approx. 3) of work the created resource will cover thousands of Polish linguistic units incorporated into the formal framework of the Cyc ontology.

The usefulness of this resource is verified in experiments covering the extraction of semantic relations from Polish texts as well in demo application allowing for Polish paraphrasing of the knowledge available as Open Data and the preliminary results are promising.

[22] It took two weeks to translate these concepts. The translation of the rest of them will be started in September 2010, since the funds for the task are paid in tranches.

[23] Measured as the number of concepts for which the system suggested the translation selected by the translator.

[24] Measured as the number of concepts for which translations were suggested.

References

[1] M. Piasecki, S. Szpakowicz, and B. Broda, *A Wordnet from the Ground Up*. Oficyna Wydawnicza Politechniki Wrocławskiej, 2009.

[2] D. B. Lenat, "CYC: A large-scale investment in knowledge infrastructure," *Communications of the ACM*, vol. 38, no. 11, pp. 33–38, 1995.

[3] R. Amaro, R. P. Chaves, P. Marrafa, and S. Mendes, "Enriching Wordnets with new Relations and with Event and Argument Structures," in *Seventh International Conference on Intelligent Text Processing and Computational Linguistics*, 2006, pp. 28 – 40.

[4] H. Somers, "Review Article: Example-based Machine Translation," *Machine Translation*, vol. 14, no. 2, pp. 113–157, 2005.

[5] P. Chrząszcz, "Automatyczne rozpoznawanie i klasyfikacja nazw wielosegmentowych na podstawie analizy haseł encyklopedycznych," Master's thesis, UST, Cracow, Poland, 2009.

[6] M. Woliński, "Morfeusz – a Practical Tool for the Morphological Analysis of Polish," in *Intelligent Information Processing and Web Mining, IIS:IIPWM'06 Proceedings*. Springer, 2006, pp. 503–512,.

[7] P. Pisarek, *Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu*. Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH, 2009, ch. Słownik fleksyjny, pp. 37–68.

[8] M. Woliński, "System znaczników morfosyntaktycznych w korpusie IPI PAN," *Polonica*, vol. XII, pp. 39–54, 2004.

[9] A. Przepiórkowski, "The potential of the IPI PAN corpus," *Poznań Studies in Contemporary Linguistics*, vol. 41, pp. 31–48, 2006.

[10] D. Jurafsky and J. H. Martin, *Speech and language processing (second edition)*. Prentice Hall, 2009.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Tools for Syntactic Concordancing

Violeta Seretan and Eric Wehrli
LATL - Language Technology Laboratory
Department of Linguistics, University of Geneva
Email: {violeta.seretan, eric.wehrli}@unige.ch

***Abstract*—Concordancers are tools that display the immediate context for the occurrences of a given word in a corpus. Also called KWIC – Key Word in Context tools, they are essential in the work of lexicographers, corpus linguists, and translators alike. We present an enhanced type of concordancer, which relies on a syntactic parser and on statistical association measures in order to detect those words in the context that are syntactically related to the sought word and are the most relevant for it, because together they may participate in multi-word expressions (MWEs). Our syntax-based concordancer highlights the MWEs in a corpus, groups them into syntactically-homogeneous classes (e.g., verb-object, adjective-noun), ranks MWEs according to the strength of association with the given word, and for each MWE occurrence displays the whole source sentence as a context. In addition, parallel sentence alignment and MWE translation techniques are used to display the translation of the source sentence in another language, and to automatically find a translation for the identified MWEs. The tool also offers functionalities for building a MWE database, and is available both off-line and online for a number languages (among which English, French, Spanish, Italian, German, Greek and Romanian).**

I. Introduction

Knowledge of a word means knowledge of the relations that this word establishes with other words: “You shall know a word by the company it keeps!” [1, p. 179]. Hence, the study of words in context—in order to analyse how words are actually used and what their typical contexts are—is a major concern in any field dealing with language from diverse perspectives, no matter whether it is theoretically or practically motivated.

The advent of the computer era and the ever-increasing availability of texts in digital format allow for virtually unlimited exploration. Yet, this is at the same time one of the biggest issues that users presented with automatically detected contexts inevitably have to face. The information comes to them as huge amounts of unstructured data, characterised by a high degree of redundancy.

To help them overcome the problem of information overload, a new generation of concordancers have been developed that are able to pre-process textual data such that the most relevant contextual information comes first [2]. This is achieved using lexical association measures that quantify the degree of interdependence between words, by relying on statistical hypothesis tests, on concepts from information theory, on data mining techniques, or by making use of various other methods ([3], [4]).

A representative example of such a concordancer is the Sketch Engine [5]. It analyses a preexisting corpus of text in order to produce, for a given word, a one-page summary of its grammatical and collocational behaviour. In doing so, it first performs a shallow parsing of the corpus by relying on automatically assigned POS tags for words, then it applies an association measure derived from Pointwise Mutual Information [6]. For illustration, Figure 1 shows part of the “sketch” produced for the French word *atteindre* (“to reach, to attain”). By clicking on the links in the frequency column, users have the possibility to see the actual concordance line, with a left and right context for each instance found in the corpus.

Developed more or less simultaneously with the Sketch Engine, our concordancer *FipsCo* ([7], [8], [9]) shares several similarities with it, and was primarily designed with the specific goal of being integrated as a new type of tool in the workbench of translators from an international organisation. In this paper, we describe this tool, its underlying resources and methodology, its latest developments, and we present the manner in which it is currently integrated in the larger, evolving processing environment available in our laboratory.

The paper is organised as follows. Section II provides an overview of FipsCo. Section III presents the resources and methods on which it is founded. Section IV describes in greater detail its functionalities, then Section V introduces FipsCoWeb, its recently developed online version. Section VI discusses the manner in which FipsCo and FipsCoWeb are integrated into the larger language processing environment of LATL. The last section contains concluding remarks.

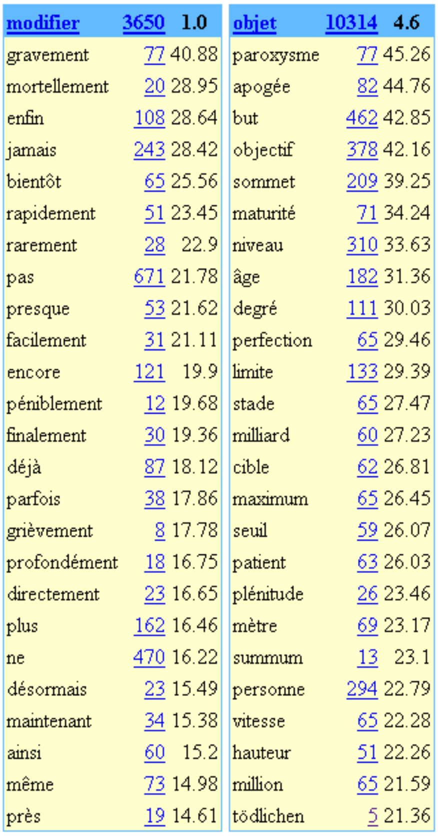
II. FipsCo: An Overview

In FipsCo, the system of syntax-based collocation extraction and concordancing developed in our laboratory, the input text is first syntactically analysed with a full parser, then it is processed with standard statistical methods which measure the strength of association between words.

Collocation, understood as “typical, specific and characteristic combination of two words” [10], is a generic term used here to encompass all syntactic word combinations found in a corpus that are relevant to the studied word, from a lexicographic point of view. As in [11], we consider that collocation refers, more generally, to “the way words combine in a language to produce natural-sounding speech and writing”.

This concept is allowed to overlap with other types of MWEs, like compounds (e.g., *wheel chair*), phrasal verbs (e.g, *to ask [somebody] out*) or certain types of less figurative idioms (*to open the door [for smth]* “to allow [smth] to happen”).

978-83-60810-27-9/09/$25.00 © 2010 IEEE



modifier	3650	1.0	objet	10314	4.6
gravement	77	40.88	paroxysme	77	45.26
mortellement	20	28.95	apogée	82	44.76
enfin	108	28.64	but	462	42.85
jamais	243	28.42	objectif	378	42.16
bientôt	65	25.56	sommet	209	39.25
rapidement	51	23.45	maturité	71	34.24
rarement	28	22.9	niveau	310	33.63
pas	671	21.78	âge	182	31.36
presque	53	21.62	degré	111	30.03
facilement	31	21.11	perfection	65	29.46
encore	121	19.9	limite	133	29.39
péniblement	12	19.68	stade	65	27.47
finalement	30	19.36	milliard	60	27.23
déjà	87	18.12	cible	62	26.81
parfois	38	17.86	maximum	65	26.45
grièvement	8	17.78	seuil	59	26.07
profondément	18	16.75	patient	63	26.03
directement	23	16.65	plénitude	26	23.46
plus	162	16.46	mètre	69	23.17
ne	470	16.22	summum	13	23.1
désormais	23	15.49	personne	294	22.79
maintenant	34	15.38	vitesse	65	22.28
ainsi	60	15.2	hauteur	51	22.26
même	73	14.98	million	65	21.59
près	19	14.61	tödlichen	5	21.36

Fig. 1. The Sketch Engine [5]: Sample (partial) output, showing collocates for the French verb *atteindre*, "to reach, to attain".

The boundaries between the different kinds of MWEs are known to be particularly difficult to be drawn, as they are rather fuzzy [12]. From a practical point of view, all MWEs pose similar processing problems, regardless of any finer-grained classification. We will henceforth refer to the output of our system as to collocations (or collocation candidates), without making further, more elaborate distinctions.

As the Sketch Engine [5], our system outputs collocation candidates grouped by types (which conflate all the instances of a specific word combination detected in the corpus), partitioned into syntactically homogeneous classes, and ranked in the reverse order of the association strength. Thus, the user may easily consult a manageable amount of contextual data, consisting of the most relevant collocates. The data presented are organised and, to a certain extent, free of redundancy.[1]

Figure 2 shows some results obtained from a French corpus by using FipsCo. These results were filtered by the user so that they contain collocations for the sought word (in this case, the verb *atteindre*, "to reach, to attain") in a specific syntactic configuration; the configuration retained here was verb-object. According to the association measure applied, the noun the most collocationally related to this word is *objectif* ("objective"). It was detected 271 times in the source corpus, and it is indeed a good collocate candidate, as one might easily agree that *atteindre un objectif* ("to reach a goal") is a collocation in French.

The concordancer displays its 100th instance in the corpus, which, as can be seen, involves a rather complex syntactic context: the order of the items in the collocation is inverted, the items are inflected and not in the base word form, and there is additional material inserted in between. Due to a grammatical transformation (passivization), the original verb-object combination is realized, at the surface level, as a subject-verb combination. The identification of these type of complex cases, which are particularly difficult to handle by pattern-based shallow parsers, is possible in our system thanks to the deep analysis provided by the parser (cf. Section III).

Among the other combinations shown in Figure 2, one might find several other MWEs with the verb *atteindre*. The tool also presents the automatically retrieved translation of the context in another language, if parallel corpora are available. In a translation environment, such corpora are typically available from translation archives. Thus, when working on a new document, translators have the possibility to see how a given expression has previous been translated in various contexts.

Figure 2 also shows the buttons *Validate*, used for manually validating the automatically extracted results, and *Translate*, used for automatically detecting a translation for the selected collocations. The *Filter* button opens the interface in Figure 3, through which the user can control which corpus results to display.

FipsCo is freely available for research as an offline tool for Windows (cf. Section IV). One of the latest developments concerned the creation of a lighter-weight online version, which has already been made available to the public. A more detailed description of FipsCo and its Web version, FipsCoWeb, is provided in Section IV.

III. Underlying Resources and Methodology

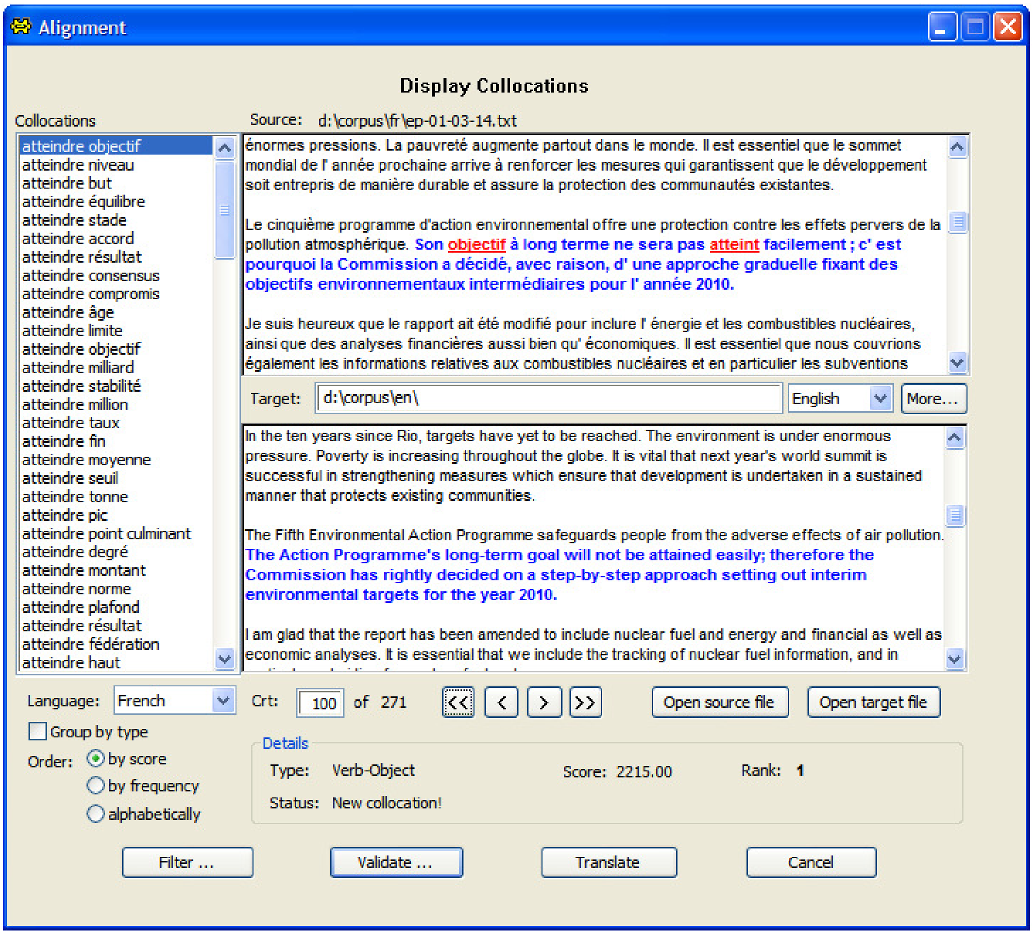
This section provides details about the resources used by FipsCo and the method used to extract from text corpora the most relevant collocates for a given word.

A. Resources

FipsCo was built as an extension of Fips, a multilingual symbolic parser based on generative grammar concepts [13]. Fips can be characterised as a strong lexicalist, bottom-up, left-to-right parser. Given a sentence, it builds a rich structural representation combining *a*) the constituent structure; *b*) the interpretation of constituents in terms of arguments; *c*) the interpretation of elements like clitics, relative and interrogative pronouns in terms of intra-sentential antecedents; and *d*) co-indexation chains linking extraposed elements (e.g., fronted NPs and wh elements) to their canonical positions.

According to the theoretical stipulations on which Fips relies, some constituents of a sentence may move from their canonical "deep" position to surface positions, due to various grammatical transformations. For instance, in the case of

[1] All the corpus instances (tokens) of a same word combination are grouped under a single entry, the corresponding collocation type.



Source: d:\corpus\fr\ep-01-03-14.txt

atteindre objectif
atteindre niveau
atteindre but
atteindre équilibre
atteindre stade
atteindre accord
atteindre résultat
atteindre consensus
atteindre compromis
atteindre âge
atteindre limite
atteindre objectif
atteindre milliard
atteindre stabilité
atteindre million
atteindre taux
atteindre fin
atteindre moyenne
atteindre seuil
atteindre tonne
atteindre pic
atteindre point culminant
atteindre degré
atteindre montant
atteindre norme
atteindre plafond
atteindre résultat
atteindre fédération
atteindre haut

énormes pressions. La pauvreté augmente partout dans le monde. Il est essentiel que le sommet mondial de l' année prochaine arrive à renforcer les mesures qui garantissent que le développement soit entrepris de manière durable et assure la protection des communautés existantes.

Le cinquième programme d'action environnemental offre une protection contre les effets pervers de la pollution atmosphérique. **Son objectif à long terme ne sera pas atteint facilement ; c' est pourquoi la Commission a décidé, avec raison, d' une approche graduelle fixant des objectifs environnementaux intermédiaires pour l' année 2010.**

Je suis heureux que le rapport ait été modifié pour inclure l' énergie et les combustibles nucléaires, ainsi que des analyses financières aussi bien qu' économiques. Il est essentiel que nous couvrions également les informations relatives aux combustibles nucléaires et en particulier les subventions

Target: d:\corpus\en\ English

In the ten years since Rio, targets have yet to be reached. The environment is under enormous pressure. Poverty is increasing throughout the globe. It is vital that next year's world summit is successful in strengthening measures which ensure that development is undertaken in a sustained manner that protects existing communities.

The Fifth Environmental Action Programme safeguards people from the adverse effects of air pollution. **The Action Programme's long-term goal will not be attained easily; therefore the Commission has rightly decided on a step-by-step approach setting out interim environmental targets for the year 2010.**

I am glad that the report has been amended to include nuclear fuel and energy and financial as well as economic analyses. It is essential that we include the tracking of nuclear fuel information, and in

Language: French

Crt: 100 of 271

☐ Group by type

Order: ● by score ○ by frequency ○ alphabetically

Type: Verb-Object Score: 2215.00 Rank: 1

Status: New collocation!

Fig. 2. FipsCo: Parallel concordancing interface, displaying filtered collocations for the French verb *atteindre* ("to reach, to attain").

the French sentence shown in Figure 2, it is considered that the noun *objectif* moved from its original position of direct object into the surface position of subject due to a passivisation transformation. The parser keeps track of this movement by linking the (empty) object position of the verb *atteint* to the extraposed noun, *objectif*. In the normalised sentence representation it builds, the parser identifies this noun as the "deep" direct object. Consequently, the combination *atteindre–objectif* can successfully be identified from this sentence as a verb-object collocation, as the parser helps abstract away from the particular surface realization.

The parser is the most important resource on which syntactic concordancers like FipsCo rely in order to filter out "noise" and to return highly accurate extraction results.[2] However, parsers are only available for a handful of languages. Fips, in particular, relies on large resources (lexica and grammars) whose construction is time-consuming.

Currently available for English, French, Spanish, Italian, Greek and German, Fips is actually conceived as a generic parsing architecture, coupling a language-independent parsing engine with language-specific extensions. The language-independent part implements the parsing algorithm, based on three main types of operations: *Project* (assignment of constituent structures to lexical entries), *Merge* (combination of adjacent constituents into larger structures), and *Move* (creation of chains by linking surface positions of "moved" constituents to their corresponding canonical positions).

The language-specific part of the Fips parser consists of grammar rules of a given language and of a detailed lexicon for that language. In the formalism used by Fips, the role of most grammar rules is to specify the conditions under which two adjacent constituents may be merged into a larger constituent by a *Merge* operation. The construction of the lexicon is supported by a morphological generation tool that creates appropriate lexical entries corresponding to a specified inflection paradigm (when applicable). Unlike other parsers, Fips does not require POS-tagged data as input; the POS is assigned to words during the analysis, based on lexical information and on the parsing hypotheses.

Given the Fips architecture and the existing tools supporting the creation of lexical resources, we believe that the effort of extending Fips to a new language is comparable to the combined effort of building POS-taggers and developing

[2] An evaluation of FipsCo is presented in [14].

shallow parsers for the same language. Our recent work on Romanian [15] confirmed that a Fips parser version that can be satisfactorily be used for the purpose of collocation extraction can be built in a reasonable amount of time, of the order of several person-months.

B. Methodology

As mentioned in Section II, the extraction of collocations from text corpora is done by using a hybrid extraction method, which combines the syntactic information provided by the Fips parser with existing statistical methods for detecting typical lexical associations in corpora.[3]

Thus, in the first step, collocation candidates are identified as combinations of lexical items in predefined syntactic configurations (for instance, verb-object) from each sentence of the corpus, by traversing the parse structures returned by the parser. In the second step, the candidates obtained are ranked according to their probability to constitute collocations, as computed with the log-likelihood ratio association measure [16]. FipsCo actually implements a wide range of other measures that the user can choose for ranking collocation candidates; log-likelihood ratio is proposed by default as it is a well-established measure for collocation extraction.

The output of FipsCo is a so-called significance list, in which one finds at the top the candidates that are most likely to actually constitute collocations. A cut-off point can be applied by the user to the results, in order to retain only the candidates with higher scores. Typically, a frequency threshold is also employed to eliminate those combinations that only occur a few times in the corpus. This is because statistical measures are unreliable on low frequency data ($f < 5$). However, we opted for keeping all the candidate data (no frequency threshold), since relevant collocations may be found among combinations occurring only a few times in the corpus. Besides, a threshold can be applied by the user afterwards. The syntactic filter applied on the otherwise huge candidate data helps our system keep the statistical computation tractable. In the systems that do not use parsed data, high frequency cut-offs are often imposed only to reduce the amount of data to process.

IV. Detailed Description of FipsCo

FipsCo is implemented in Component Pascal under BlackBox Component Builder IDE,[4] just as the syntactic parser Fips, on which it relies. It makes an extensive use of the SQL database query language in order to store the extraction results, compute the collocation scores, filter the data that will be displayed, etc.

The system has, in principle, a pipeline architecture, as the typical execution flow follows the order in which the main components of the system are described below. However, there are no restrictions to the order in which the various components can be used, since the extracted and validated results can be stored and accessed later for visualisation.

A. File Selection

The source corpus used in an extraction session is specified by selecting the folder which contains the desired files and, optionally, by applying an automatic or manual filter on its content.

The automatic filter is based on:

- file location: inclusion or exclusion of the sub-folders; exclusion of sub-folders having a specific name;
- file name: this might be required to contain a given string of characters;
- file type: this must belong to a list of allowed types. The system supports all the file formats that can be currently imported by BlackBox, e.g., `odc` – Oberon document; `txt`, `htm`, and `html` – text; `rtf`, `doc` – rich text format; and `utf` – Unicode.
- file last modification date (from `date1` to `date2`; in the last `n` days).

In addition, the selection can be further narrowed manually, as the user may select or deselect items after the automatic filter applies. For instance, it is possible to choose items (files or folders) in the first level of the source folder with a mouse click, or by using standard selection commands (check all; uncheck all; invert selection).

B. Collocation Extraction

The collocation extractor is the main component of the system. It iteratively processes all the files in the selection. The number of files that can be processed is virtually unlimited. The collocation candidates identified from the parse trees are incrementally added to previous results until an extraction session ends. They are stored either in a database or in a single text file. As an option, they can also be stored file by file in a folder whose structure mirrors the structure of the source folder.

At the end of the extraction session, several processing statistics are computed for the source corpus that are derived from parsing information (e.g., the total number of tokens, sentences, sentences with a complete parse). Then, the candidates identified are ranked according to the chosen association measure (by default, log-likelihood ratio [16]).

C. Filtering

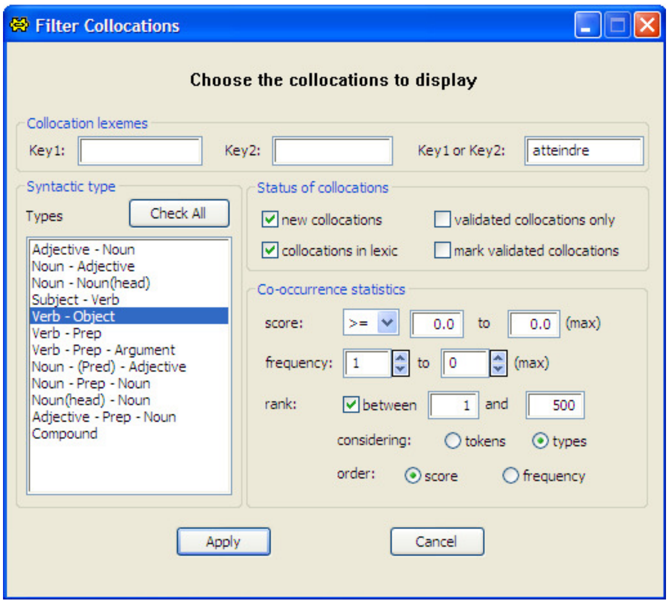
This component selects the results to be displayed in the concordancer, according to the parameters set by the user (see Figure 3). The extracted collocations can be filtered according to several criteria:

- syntactic type: the user can select one or more types from a list that is automatically built from the database containing the extracted collocations;
- collocation score: a range from `score1` to `score2`;[5]

[3] It is important to note that the method itself is not dependent on Fips or any of the specific theoretical assumptions made by Fips, but it can be used in conjunction with other parsers.

[4] BlackBox is developed by Oberon Microsystems (http://www.oberon.ch). A characteristic of this development environment is the ease of editing graphical user interfaces components, which turned into a big advantage for our system, in which visualisation plays a major role.

[5] The user is not required to know the actual maximal values; the corresponding fields can be left blank and these values will be retrieved by the system.



Filter Collocations

Choose the collocations to display

Collocation lexemes

Key1: Key2: Key1 or Key2: atteindre

Syntactic type

Types — Check All

Adjective - Noun
Noun - Adjective
Noun - Noun(head)
Subject - Verb
Verb - Object
Verb - Prep
Verb - Prep - Argument
Noun - (Pred) - Adjective
Noun - Prep - Noun
Noun(head) - Noun
Adjective - Prep - Noun
Compound

Status of collocations

☑ new collocations □ validated collocations only
☑ collocations in lexic □ mark validated collocations

Co-occurrence statistics

score: >= 0.0 to 0.0 (max)
frequency: 1 to 0 (max)
rank: ☑ between 1 and 500
considering: ○ tokens ● types
order: ● score ○ frequency

Apply Cancel

Fig. 3. FipsCo: Interface for filtering collocations.

- corpus frequency: a range from `freq1` to `freq2`;
- collocation keywords: the user can search for collocations containing a specific word.

In addition, the user can specify the range of results to display (from `rank1` to `rank2`), according to the order given by the collocation score or by the corpus frequency. The range restrictions can be applied both to collocation types and to collocation instances (tokens).

D. Concordancing

This component is responsible for the visualisation of extraction results according to the selection made by the user. The (filtered) list of collocations is displayed on the left hand-side on the concordance interface, and can be ordered by score, by frequency in the corpus, or alphabetically. On the right hand-side, a text panel displays the context of the currently selected collocation in the source document. The whole content of the document is accessible, and is automatically scrolled to the current collocation; this collocation and the sentence in which it occurs are highlighted with different colors (cf. Figure 2).

Each item in the list represents a collocation type; its corresponding instances are read from the database when the user clicks on it. The right panel automatically displays the first instance, then the user has the possibility of navigating through all the instances by using the standard browsing arrows (`<< - first`, `< - previous`, `> - next`, `>> - last`), or to skip to a given instance by entering its order number.

The visualisation interface also displays information about the rank of the currently selected collocation, its syntactic type, its score, and its status relative to the parser's lexicon (new collocation, or collocation in lexicon). The user can easily switch to a different source language in order to load the collocations already extracted for that language, if these were stored in the same database.

E. Complex Collocations

By treating already extracted collocations as single lexical items, FipsCo is able to identify complex collocations that can be seen as structures containing embedded collocations: for instance, *atteindre point culminant* ("to be at the highest level") is a complex collocation of verb-object type, which contains an embedded noun-adjective collocation, *point culminant*.

The detection of such complex collocation is particularly useful when the resulting expression constitutes a non-decomposable compound, or when it contains a nested compound. In these cases, it is important to highlight the whole expression rather than nonsensical sub-parts. For instance, *genetically modified organisms* is a compound, and it will be desirable to output it as a whole rather than only the sub-part *modified organisms*. The expression *second world war* is more compositional, as *world war* is a collocation on its own. However, it is desirable to eliminate *second war* from the extraction results, if it only occurs in the corpus in the longer expression *second world war* .

Our method of detecting complex collocations is described in [17] and [18]. FipsCo includes a concordancing interface for displaying complex collocations, which is similar to the standard interface shown in Figure 2.

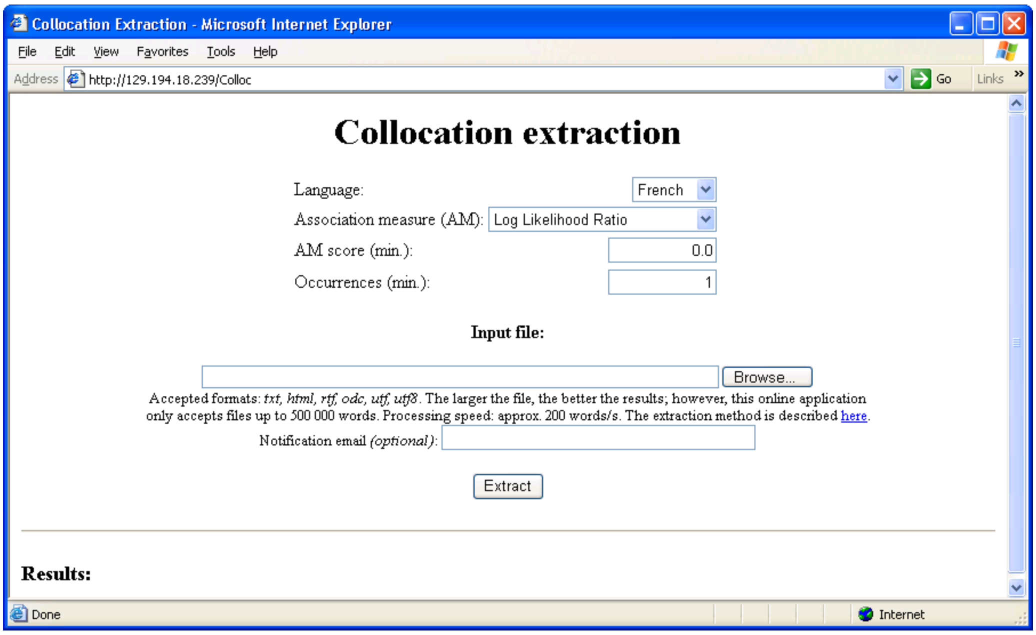
F. Sentence Alignment

When parallel corpora are available, the target sentence containing the counterpart of the source sentence can be detected and displayed in the alignment interface below the source sentence. The user selects the target language from a list of languages and specifies the path of the target corpus and the filename transformation rule needed to determine the filename of the target document (i.e., of the translation) from the filename of the source document. These rules assume that the source folder and the target folder have the same structure, and that the target filename can be obtained from the source filename by replacing the prefix and/or the suffix of the filename (which are assumed to be variable across languages), while keeping the middle part constant. For instance, `35.1.001E.txt` can be obtained from `35.1.001F.txt` by replacing the suffix `F` with `E`.

Once the target file has been found, the sentence that is likely to be the translation of the source sentence is identified using an in-house sentence alignment method ([7], [9]). The alignment component is operational both for binary collocations and for complex collocations.

G. Validation

This component provides functionalities that allow the user to create and maintain a list of manually validated collocations from the collocations visualised with the concordance and the alignment interfaces. An entry contains basic information about a collocation (such as the collocation keywords, lexeme


Collocation Extraction - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address
http://129.194.18.239/Colloc
Go
Links
Browse...
Extract
Done
Internet

Collocation extraction

Language: French

Association measure (AM): Log Likelihood Ratio

AM score (min.): 0.0

Occurrences (min.): 1

Input file:

Accepted formats: *txt, html, rtf, odc, utf, utf8*. The larger the file, the better the results; however, this online application only accepts files up to 500 000 words. Processing speed: approx. 200 words/s. The extraction method is described here.

Notification email *(optional)*:

Results:

Fig. 4. FipsCoWeb: Interface (screen capture).

indexes for the participating items, syntactic type, score and corpus frequency). A monolingual entry may also contain the source sentence of the currently visualised instance, which provides a naturally-occurring usage sample for the collocation. A bilingual entry stores, in addition, the target sentence found via alignment and the translation proposed for the collocation: the translation can be manually retrieved by the user from the target sentence.

Additional information related to the currently visualized collocation instance is stored (namely, the name of the source and target file, the file position of the collocation's items in the source and target files, and the file position of the source and target sentences). Most of this information is automatically filled in by the system. The entries in the list of collocations validated in a session can be updated, deleted, or saved—completely or in part—by the user in a monolingual and in a bilingual database.

H. Translation

This component attempts to detect a translation equivalent for the collocations visualised in the concordancer, by scanning the existing translations and using a strategy briefly described below.

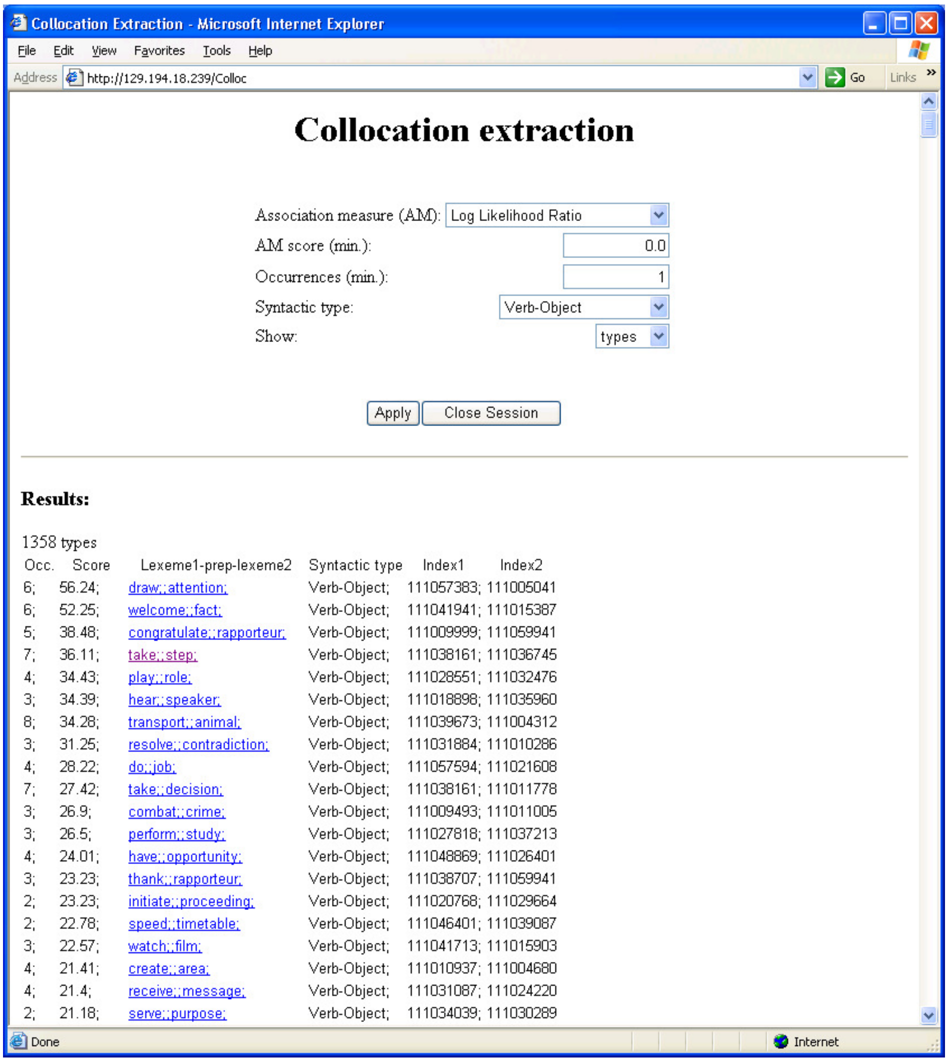
First, a limited number of corpus sentences (50 in our current experiments) in the source language is retrieved for the source collocation, based on the corpus instances detected during extraction. The alignment component is then used for finding, for each source sentence, the corresponding target sentence in the desired target language, for which a parallel corpus is available.

The target mini-corpus thus obtained is parsed, and collocations are extracted from it using the same method that was applied to the source corpus. Finally, a process of collocation matching takes place, which tries to find, among the extracted collocations, the one that is likely to represent a translation for the source collocation. The matching is performed by applying a series of filters on the extracted pairs that gradually reduce their number until a single item is retained, which will be proposed as translation. An updated description of the translation method can be found in [19].

V. Online Version: FipsCoWeb

FipsCoWeb, which is introduced for the first time in this paper, is the online version of the FipsCo system. Its current interface is shown in Figure 4. FipsCoWeb allows the user to upload a file and to set the initial processing and visualization parameters (e.g., association measure, cut-off score, frequency threshold). After the processing is done on the server side, the user is presented with the results, as shown in Figure 5. The user has then the possibility to apply different parameters, to apply a syntactic filter, and to see the actual occurrences of a collocation by clicking on the corresponding link. The words in the collocation will be presented in the sentence context, and highlighted for readability (cf. Figure 6).

FipsCoWeb currently allows users to upload files containing up to 0.5 million words. While this is a reasonable size for online corpus exploration, the processing, which is performed at an average of 200 tokens/second, might take a while to complete. Depending on the file size, users might only be able to see the results after a few minutes or a longer lapse of time (typically, half an hour). For this reason, FipsCoWeb gives users the possibility to enter the e-mail address at which the link to results is sent when the server-side computation is completed. Results are stored on the server and can be


Collocation Extraction - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://129.194.18.239/Colloc
Go
Links
Done
Internet

Collocation extraction

Association measure (AM): Log Likelihood Ratio

AM score (min.): 0.0

Occurrences (min.): 1

Syntactic type: Verb-Object

Show: types

Apply | Close Session

Results:

1358 types

Occ.	Score	Lexeme1-prep-lexeme2	Syntactic type	Index1	Index2
6;	56.24;	draw;;attention;	Verb-Object;	111057383;	111005041
6;	52.25;	welcome;;fact;	Verb-Object;	111041941;	111015387
5;	38.48;	congratulate;;rapporteur;	Verb-Object;	111009999;	111059941
7;	36.11;	take;;step;	Verb-Object;	111038161;	111036745
4;	34.43;	play;;role;	Verb-Object;	111028551;	111032476
3;	34.39;	hear;;speaker;	Verb-Object;	111018898;	111035960
8;	34.28;	transport;;animal;	Verb-Object;	111039673;	111004312
3;	31.25;	resolve;;contradiction;	Verb-Object;	111031884;	111010286
4;	28.22;	do;;job;	Verb-Object;	111057594;	111021608
7;	27.42;	take;;decision;	Verb-Object;	111038161;	111011778
3;	26.9;	combat;;crime;	Verb-Object;	111009493;	111011005
3;	26.5;	perform;;study;	Verb-Object;	111027818;	111037213
4;	24.01;	have;;opportunity;	Verb-Object;	111048869;	111026401
3;	23.23;	thank;;rapporteur;	Verb-Object;	111038707;	111059941
2;	23.23;	initiate;;proceeding;	Verb-Object;	111020768;	111029664
2;	22.78;	speed;;timetable;	Verb-Object;	111046401;	111039087
3;	22.57;	watch;;film;	Verb-Object;	111041713;	111015903
4;	21.41;	create;;area;	Verb-Object;	111010937;	111004680
4;	21.4;	receive;;message;	Verb-Object;	111031087;	111024220
2;	21.18;	serve;;purpose;	Verb-Object;	111034039;	111030289

Fig. 5. FipsCoWeb: Sample results (screen capture).

consulted later, until users explicitly decide to clear them, by clicking on the *Close Session* button. A feature that is currently unavailable in the system, but can be easily implemented, is the search for collocations with a given word.[6]

The Web version has been implemented in BlackBox (see Section II), and the Web server itself[7] runs as a BlackBox program. This made the integration between the involved software modules easier. However, since it runs as a unique Windows process, it cannot be efficiently used for the parallel processing of large files. A solution is currently being worked on to circumvent this problem. Future work will focus on implementing FipsCoWeb as a Web service.

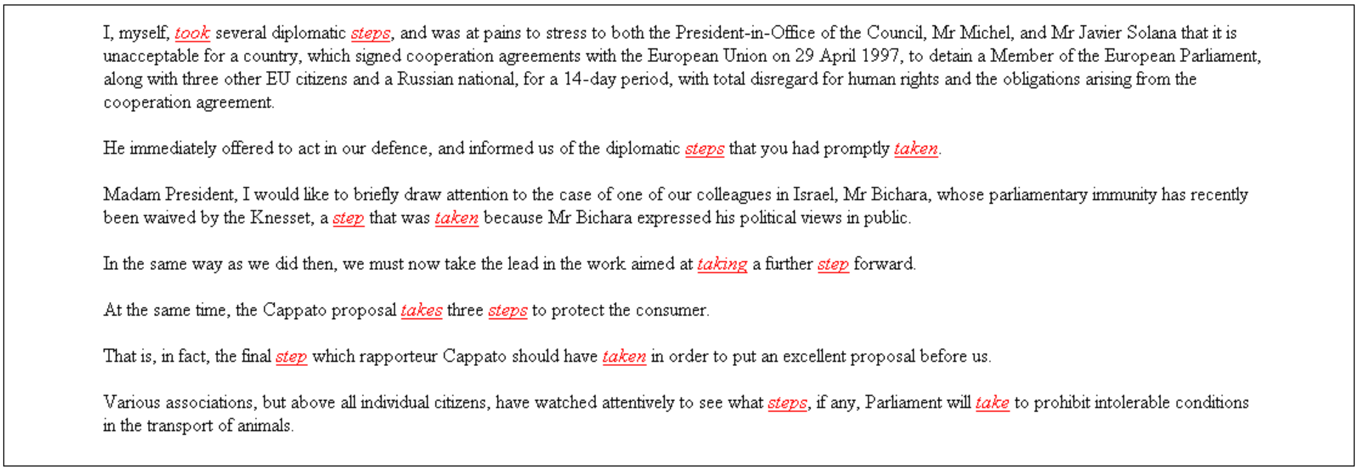
VI. Integration in the NLP Environment of LATL

LATL develops a range of NLP tools in several areas. FipsCo (and its online version, FipsCoWeb) are not isolated tools, but are part of a larger processing framework specifically dealing with MWEs, from different practical perspectives.

As a matter of fact, the corpus-based study of words and their collocates was not, in our case, a goal in itself. The collocations that lexicographers manually validate are entered into the lexical database of the parser Fips, and are used to guide future analyses performed by Fips [20]. Their translations (either manually or automatically obtained) are used to populate the bilingual lexicon of a rule-based machine translation based on Fips. The collocations added in the lexicon are further used in two applications of terminology assistance, Twic and TwicPen [21], which look up the lexicon and propose a translation for a given word that is compatible with the grammatical context. If the selected word is part of a MWE, these systems output the translation of the whole MWE, rather than a translation for the word in isolation. Work is under way to augment the MWE resources for all the languages supported by the Fips parser.

[6] Note that the online version does not aim to re-implement all the functionalities of FipsCo.

[7] O_3-WAF (Web-Application-Framework); http://o3-software.de/



I, myself, took several diplomatic steps, and was at pains to stress to both the President-in-Office of the Council, Mr Michel, and Mr Javier Solana that it is unacceptable for a country, which signed cooperation agreements with the European Union on 29 April 1997, to detain a Member of the European Parliament, along with three other EU citizens and a Russian national, for a 14-day period, with total disregard for human rights and the obligations arising from the cooperation agreement.

He immediately offered to act in our defence, and informed us of the diplomatic steps that you had promptly taken.

Madam President, I would like to briefly draw attention to the case of one of our colleagues in Israel, Mr Bichara, whose parliamentary immunity has recently been waived by the Knesset, a step that was taken because Mr Bichara expressed his political views in public.

In the same way as we did then, we must now take the lead in the work aimed at taking a further step forward.

At the same time, the Cappato proposal takes three steps to protect the consumer.

That is, in fact, the final step which rapporteur Cappato should have taken in order to put an excellent proposal before us.

Various associations, but above all individual citizens, have watched attentively to see what steps, if any, Parliament will take to prohibit intolerable conditions in the transport of animals.

Fig. 6: FipsCoWeb: Collocation instances in context (screen capture).

VII. Conclusion

In this article, we provided an updated description of FipsCo, a tool for extracting collocations (and multi-word expressions more generally) from corpora, which has been developed at LATL in the last several years. Since FipsCo is based on parsing and offers multiple visualisation functionalities, it can be seen as a tool for syntax-based corpus exploration, or syntactic concordancing.

Also, we introduced FipsCoWeb, the online version of this tool, recently developed and already functional. This version can be used to upload a user's own text corpus as a file and to consult the retrieved collocations. The two tools are part of a larger processing framework dedicated to MWEs, and are being used to provide resources for the two main long-term NLP projects pursued in our laboratory, namely, a multilingual symbolic parser and a machine translation system based on it.

Acknowledgement

The work reported in this paper has been supported by the Swiss National Science Foundation (grant no. 100012-117944). We would like to thank the four anonymous reviewers of an earlier version of this paper for useful comments and suggestions.

References

[1] J. R. Firth, *Papers in Linguistics 1934-1951*. Oxford: Oxford Univ. Press, 1957.

[2] G. Barnbrook, *Language and Computers: A Practical Introduction to the Computer Analysis of Language*. Edinburgh: Edinburgh University Press, 1996.

[3] S. Evert, "The statistics of word cooccurrences: Word pairs and collocations," Ph.D. dissertation, University of Stuttgart, 2004.

[4] P. Pecina, "Lexical association measures: Collocation extraction," Ph.D. dissertation, Charles University in Prague, 2008.

[5] A. Kilgarriff, P. Rychly, P. Smrz, and D. Tugwell, "The Sketch Engine," in *Proceedings of the Eleventh EURALEX International Congress*, Lorient, France, 2004, pp. 105–116.

[6] K. Church and P. Hanks, "Word association norms, mutual information, and lexicography," *Computational Linguistics*, vol. 16, no. 1, pp. 22–29, 1990.

[7] L. Nerima, V. Seretan, and E. Wehrli, "Creating a multilingual collocation dictionary from large text corpora," in *Companion Volume to the Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL'03)*, Budapest, Hungary, 2003, pp. 131–134.

[8] V. Seretan, L. Nerima, and E. Wehrli, "A tool for multi-word collocation extraction and visualization in multilingual corpora," in *Proceedings of the Eleventh EURALEX International Congress, EURALEX 2004*, Lorient, France, 2004, pp. 755–766.

[9] V. Seretan, "Collocation extraction based on syntactic parsing," Ph.D. dissertation, University of Geneva, 2008.

[10] F. J. Hausmann, "Kollokationen im deutschen Wörterbuch. Ein Beitrag zur Theorie des lexikographischen Beispiels," in *Lexikographie und Grammatik. Akten des Essener Kolloquiums zur Grammatik im Wörterbuch*, ser. Lexicographica. Series Major 3, H. Bergenholtz and J. Mugdan, Eds., 1985, pp. 118–129.

[11] D. Lea and M. Runcie, Eds., *Oxford collocations dictionary for students of English*. Oxford: Oxford University Press, 2002.

[12] K. R. McKeown and D. R. Radev, "Collocations," in *A Handbook of Natural Language Processing*, R. Dale, H. Moisl, and H. Somers, Eds. New York, USA: Marcel Dekker, 2000, pp. 507–523.

[13] E. Wehrli, "Fips, a "deep" linguistic multilingual parser," in *ACL 2007 Workshop on Deep Linguistic Processing*, Prague, Czech Republic, 2007, pp. 120–127.

[14] V. Seretan and E. Wehrli, "Multilingual collocation extraction with a syntactic parser," *Language Resources and Evaluation*, vol. 43, no. 1, pp. 71–85, 2009.

[15] V. Seretan, E. Wehrli, L. Nerima, and G. Soare, "FipsRomanian: Towards a Romanian version of the Fips syntactic parser," in *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, 2010.

[16] T. Dunning, "Accurate methods for the statistics of surprise and coincidence," *Computational Linguistics*, vol. 19, no. 1, pp. 61–74, 1993.

[17] V. Seretan, L. Nerima, and E. Wehrli, "Extraction of multi-word collocations using syntactic bigram composition," in *Proceedings of the Fourth International Conference on Recent Advances in NLP (RANLP-2003)*, 2003, pp. 424–431.

[18] L. Nerima, E. Wehrli, and V. Seretan, "A recursive treatment of collocations," in *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, 2010.

[19] V. Seretan, "Extraction de collocations et leurs équivalents de traduction à partir de corpus parallèles," *TAL*, vol. 50, no. 1, pp. 305–332, 2009.

[20] E. Wehrli, V. Seretan, and L. Nerima, "Sentence analysis and collocation identification," in *Proceedings of the Workshop on Multiword Expressions: from Theory to Applications (MWE 2010)*, Beijing, China, 2010, pp. 27–35.

[21] E. Wehrli, L. Nerima, V. Seretan, and Y. Scherrer, "On-line and off-line translation aids for non-native readers," in *Proceedings of the International Multiconference on Computer Science and Information Technology*, Mragowo, Poland, 2009, pp. 299–303.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Effective natural language parsing with probabilistic grammars

Paweł Skórzewski
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
ul. Umultowska 87, 61-614 Poznań, Poland
Email: pawel@st.amu.edu.pl

***Abstract*—This paper presents an example of application of a PCFG parsing algorithm based on the A* search procedure in a machine translation system. We modified the existing CYK-based parser used in the machine translation system Translatica and applied the A* parsing algorithm in order to improve the performance of the parser.**

I. Introduction

PROBABILISTIC context-free grammars are a useful tool for modeling natural languages. One of the methods commonly used for parsing PCFGs is chart parsing. In general, the time complexity of chart parsing algorithms is worst-case cubic. In practice, for large grammars and long sentences, even cubic time can be insufficient.

The process of parsing PCFGs can be viewed as a process of finding a path in a certain weighted multigraph. This means that graph search algorithms can be used for parsing. The parse which has the highest probability from all the possible parses of the sentence is called the *Viterbi parse*. Finding a Viterbi parse of a given sentence can be realized by algorithms that find the shortest path in a graph. The graph algorithms are used to accelerate the chart parsing of PCFGs.

One of the methods of speeding up the parsing is the best-first strategy, described in [1] and [2]. The order of removing edges from agenda is determined by a figure of merit. If the figure of merit is properly chosen, the number of edges processed during the parsing can be significantly reduced.

Another method is a beam-search algorithm, described in [5] and [6]. In this method, only a limited number of best parses is tracked at any time. The fewer parses are tracked, the shorter time is required to complete the algorithm.

Both best-first parsing and beam-search parsing are greedy algorithms. Unfortunately, neither of them guarantees the finding of the actual Viterbi parse of a given sentence.

The A* algorithm is used to find the shortest path between two given vertices of a graph, assuming a heuristic function is specified. The heuristic function is defined on the vertices of the graph and estimates the distance between the given vertex and the target vertex. The algorithm tries to minimize the sum of the distance covered so far and the value of the heuristic.

Klein and Manning in [3] presented the algorithm that uses the A* procedure to parse the probabilistic context-free grammar. The authors showed that if a proper heuristic function is applied, the A* parsing algorithm can significantly reduce the time required to find the Viterbi parse of a given sentence.

II. The A* parsing algorithm

The A* parser is a kind of a chart parser. The basic data structure the parser operates on is called an edge. The edge is a triple $X[i,j]$, where X is a symbol of the grammar and (i,j) is the span of this symbol in the parse tree of the string $w = w_1 \dots w_n$, ie. X dominates the substring $w_{i+1} \dots w_j$. During the algorithm, a score is attached to each edge. The edges waiting to be processed are stored in the structure called agenda. The edges whose best parses have already been found are stored in a chart.

In the beginning, the edges representing the words of the parsed sentence are put into the agenda. Then, while the agenda is not empty, the following procedure (called a turn of the parser) is run: The edge with the highest priority is removed from the agenda. Then if the removed edge can be combined with some edges already in the chart then new edges are formed and inserted into the agenda. In other words, let $e = X[i,j]$ denote the removed edge. If there is an edge $Y[k,i]$ in the chart and a rule $Z \to YX$ in the set of productions, the new edge $Z[k,j]$ is inserted into the agenda. Similarly, if there is an edge $Y[j,k]$ in the chart and a rule $Z \to XY$ in the rule set, the new edge $Z[i,k]$ is formed etc. The rules of the grammar are used to combine the current edge with the symbols in the chart to create new edges and insert the newly formed edges into the agenda. The newly formed edges are put into the agenda and the process repeats until the agenda is empty.

Let $G = (V,T,R,S,P)$ be a probabilistic context-free grammar and $w = w_1 \dots w_n \in T^+$ a parsed sentence. Following [3], we will define the *Viterbi inside score* $\beta(e) = \beta_{G,w}(e)$ of an edge $e = X[i,j]$ as the logarithm of the probability of a best inside parse of e, ie. the maximum of the log-probabilities of the derivation $X \Rightarrow^* w_{i+1} \dots w_j$. We will denote by $b(e)$ the estimate of $\beta(e)$ which represent the log-probability of the best inside parse of e calculated so far. At the beginning of the algorithm $b(e) = -\infty$ and never decreases. When e is removed from agenda, $b(e) = \beta(e)$.

The *Viterbi outside score* $\alpha(e) = \alpha_{G,w}(e)$ of an edge $e = X[i,j]$ is defined as the maximum of the log-probabilities

978-83-60810-27-9/09/$25.00 © 2010 IEEE

of the derivation $S \Rightarrow^* w_1 \dots w_i X w_{j+1} \dots w_n$ (or the log-probability of the best outside parse of e). We will also estimate the value of $\alpha(e)$ by $a(e)$. We say that the estimation a is *admissible* if $a(e) \geq \alpha(e)$.

Provided that the value of $\beta + a$ never increases, we can use $\beta + a$ to prioritize edges in agenda. The β corresponds to the actual distance from the start vertex and a plays the role of the heuristic for the A* algorithm.

Klein and Manning in [3] present a series of different heuristic functions, based both on context summary estimates and grammar projection estimates.

The context summary estimates are based on the knowledge about the neighborhood of the current symbol. The value of the context summary estimate $a(X[i,j])$ is the maximum of all log-probabilities of the derivations $S \Rightarrow^* uXv$, where $uandv$ satisfies some given condition. There is a wide range of possible estimates, between the NULL estimate (when we have no information about the context), equal constantly 0, and the TRUE esimate (when we have the full information), equal the exact value of β.

The grammar projection estimates use the exact context but the reduced grammar. A probabilistic context-free grammar $G = (V, T, R, S, P)$ can be projected to some weighted context-free grammar $G' = (V', T', R', S', P')$ by a given function $\pi \colon V \cup T \to N$, where N is an arbitrary set of symbols, in the following way:

- $V' := \{\pi(A) \colon A \in V\} \subseteq N$,
- $T' := \{\pi(a) \colon A \in T\} \subseteq N$ and $T' \cap V' = \emptyset$,
- $R' := \{\hat{\pi}(r) \colon r \in R\}$, where

$$\hat{\pi} \colon R \to R',$$
$$\hat{\pi}(A \to X_1 \dots X_k) := \pi(A) \to \pi(X_1) \dots \pi(X_k),$$

- $S' := \pi(S)$,
- P' is determined as

$$P'(r') := \max_{r \colon \hat{\pi}(r) = r'} P(r).$$

The resulting grammar is not always probabilistic because the sum of the values $P'(A \to \zeta)$ for a fixed A can be greater then 1. For each rule $r \in R$ the following relation holds:

$$P'(r') \geq P(r)$$

which guarantees that the probabilities of the rules never decrease during the process of grammar projection. It implies that

$$\alpha_{G',w}(e') \geq \alpha_{G,w}(e)$$

for each edge e and its projection e', so we can use $a(e) = \alpha_{G',w}(e')$ as an estimate.

As in the case of context summary estimates, there is a wide range of possible grammar projection estimates, between the NULL estimate (which corresponds to the constant projection) and the TRUE estimate (based on the identity projection).

III. The implementation of A* parsing algorithm in the Translatica translation system

A. Translatica machine translation system

Translatica is a rule-based machine translation system that translates between languages: Polish, English, German and Russian. The German language parser of the Translatica system (in this paper I will call it *the Translatica parser* for simplicity) uses a weighted context-free grammar and a CYK-based agenda parsing algorithm. The weights used in this parser are probabilities extracted automatically from the TüBa treebank[1] using statistical methods. We decided to try to use the A* parsing algorithm in the Translatica parser to improve its speed and performance.

The implementation of the A* algorithm in the Translatica parser consisted in the adaptation of the current parser to use the A* methods.

B. SX heuristic

One of the heuristics described in [3] is the SX context summary estimate. Let $G = (V, T, R, S, P)$ be a PCFG and $w = w_1 \dots w_n \in T^+$. SX estimate $a_{G,w}(X[i,j])$ specifies the number of terminal symbols to the left from the current edge (i), to the right ($n - j$) and the label of the considered symbol (X). We chose to use the SX heuristics because [3] predicted that the application of SX heuristics in A* parsing algorithm would result in significant edge savings and require relatively little precomputation.

In every turn of the parser the sum $\beta(e) + a(e)$ is found for each edge e. The value of $\beta(e) + a(e)$ is used to prioritize the edges in agenda. The edge e with the maximal sum $\beta(e) + a(e)$ is removed from the agenda and then processed in the way described before.

The value of the heuristic is calculated only if it is necessary. In the process of memoization, the value of the heuristic for a given context summary is calculated only for the first time and then stored in the memory. This process speeds up the calculations of heuristic.

C. Attribute grammar projection

The grammar used in the Translatica parser is not exactly a PCFG but rather a kind of an attribute grammar. Each rule is equipped with a set of attribute expressions. The rule can be used only if the actual values of the attributes associated with symbols occuring in the rule satisfy the epressions associated with the rule.

It is possible to construct the context-free grammar that is equivalent to a given attribute grammar. This can be obtained by considering all possible configurations of attribute values, fixing the values of attributes and taking the adequate rules into account.

We can project the probabilistic attribute grammar to a PCFG by fixing the values of selected attributes and omitting the others.

We have implemented the grammar projection by fixing the most common attributes.

[1] http://www.sfs.uni-tuebingen.de/en/tuebadz.shtml

TABLE I
AVERAGE TRANSLATION TIMES FOR A SINGLE SET OF 100 SENTENCES.

threshold	50000		10000	
time	absolute	relative	absolute	relative
without A*	1 min 9 s	1.00	36 s	1.00
with A*, SX heur.	2 min 2 s	1.77	52 s	1.44
with A*, NULL heur.	1 min 20 s	1.16	37 s	1.03

TABLE II
AVERAGE TRANSLATION TIMES FOR A SET OF 100 SENTENCES REPEATED 5 TIMES.

threshold	50000		10000	
time	absolute	relative	absolute	relative
without A*	5 min 52 s	1.00	2 m 56 s	1.00
with A*, SX heur.	11 min 50 s	2.02	4 m 32 s	1.55
with A*, NULL heur.	6 min 55 s	1.18	3 m 6 s	1.06

D. NULL heuristic

We have also implemented the NULL heuristics, in order to compare the results of implementing various heuristics. The NULL heuristic, as constatntly equal 0, is easy to implement.

IV. RESULTS

We have conducted various tests in order to compare the performance of the current Translatica parser and the new A*-based parser.

In the Translatica parser it is possible to limit the maximum number of turns of the parser by setting a so called *threshold*. The default value of the threshold in the Translatica parser is 50000.

We have done a series of machine translations (in different conditions) of a sample set of 100 German sentences on various topics. The tests differed in the following parameters:

- the parsing algorithm:
  - the previous Translatica parser,
  - the A* parsing algorithm with SX heuristic,
  - the A* parsing algorithm with NULL heuristic;
- the value of the threshold:
  - 50000,
  - 10000;
- the way the input is given:
  - single sentences,
  - a set of all 100 sentences,
  - a set of all 100 sentences repeated few times.

Differentiation of ways of providing input data is aimed to test the influence of memoization to the performance of the parser.

Table I presents the average times of translating the test set of the 100 German sentences for different threshold values and different heuristics.

Table II presents the average times of translating the test set consisting of 5 copies of the set of 100 German sentences used in the previous test.

The implementation of the A* algorithm applied in the experiment didn't speed up the process of translation in the comparison with the previous Translatica parser. The NULL heuristic was proved to be faster than the SX heuristic and was only slightly slower than the old parser. It is particularly clear for the threshold of 10000, when the translation time using A* with NULL heuristic was comparable to translationa time without A*.

The differences in the quality of the translation were rather insignificant. The majority of the translations were identical, if not, the differences were in general in the single words. There are senteces whose translations by A* algorithm for a threshold of 10000 are identical with translations done by the old algorithm for the threshold of 50000, and better than translations done by the old algorithm for the threshold of 10000. It can mean that the number of steps needed to find the best parse using the A* algorithm is lower then using the previous algorithm. The reason why the translation time with A* is still longer than without A* can be put down to the implementation difficulties. Moreover, it is worth mentioning that the quality of translation of some sentences was better than with the old parser.

V. CONCLUSIONS

The experiment was an innovative try to implement the A* algorithm in an existing CYK-based parser. The implementation was intended to be integrated with the existing parser and to require as little changes to the existing code as possible. The experiment brought a great experience and significant knowledge about the implementational issues of the A* algorithm and about the possibilities of parsing improvement.

There are reasonable grounds that future research will bring the desired results—the faster translation with the Translatica system using the A* algorithm—especially if the following issues will be resolved.

There are various possible reasons why the implementation of A* algorithm in the Translatica machine translation system haven't brought the expected performance improvement.

One of possible reasons can be the specific of the Translatica parser. The grammar used in the Translatica parser doesn't have a distinguished start symbol. A given sentence is parsed using the bottom-up method—the parse tree is built upward while it is possible to combine its edges with edges from the agenda. The advantage of such approach is that it is possible to parse the phrases that are not the full sentences but only the fragments. On the other hand, this solution is rather unfavorable for the implementation possibilities for the A* algorithm because distingushing the start symbol is necessary for proper heuristic calculation.

The other reason is that the fully functional implementation of the A* parsing algorithm in the Translatica parser would involve the significant changes in the structure of the very parser.

VI. FUTURE WORK

We will continue the research concerning the use of modern tools and algorithms to accelerate the process of machine translation. We plan to modify some mechanisms used in the Translatica parser in order to enable the implementation of a fully functional parser based on the A* algorithm and also some other minor improvements.

References

[1] S. A. Caraballo and E. Charniak, "New figures of merit for best-first probabilistic chart parsing," *Computational Linguistics*, vol. 24, 1998, pp. 275-298.

[2] E. Charniak, S. Goldwater and M. Johnson, "Edge-based best-first chart parsing," *In Proceedings of the Sixth Workshop on Very Large Corpora*, 1998, pp. 127-133.

[3] D. Klein and C. D. Manning, "A* Parsing: Fast Exact Viterbi Parse Selection," *In Proceedings of the Human Language Technology Conference and the North American Association for Computational Linguistics (HLT-NAACL)*, 2003, pp. 119-126.

[4] C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, Prentice Hall, New Jersey, 2000.

[5] A. Ratnaparkhi, "Learning to parse natural language with maximum entropy models," *Machine Learning*, vol. 34, 1999, pp. 151-175.

[6] B. Roark, "Probabilistic top-down parsing and language modeling," *Computational Linguistics*, vol. 27, 2001, pp. 249-276.

[7] P. Skórzewski, *Efektywny parsing języka naturalnego przy użyciu gramatyk probabilistycznych* (*Effective Natural Language Parsing Using Probabilistic Grammars*), Master Thesis on Adam Mickiewicz University, Poznań, 2010.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Finding Patterns in Strings using Suffixarrays

Herman Stehouwer
Tilburg University
Tilburg Centre for Cognition and Communication
Tilburg, The Netherlands
Email: J.H.Stehouwer@uvt.nl

Menno van Zaanen
Tilburg University
Tilburg Centre for Cognition and Communication
Tilburg, The Netherlands
Email: M.M.vanZaanen@uvt.nl

***Abstract*—Finding regularities in large data sets requires implementations of systems that are efficient in both time and space requirements. Here, we describe a newly developed system that exploits the internal structure of the enhanced suffixarray to find significant patterns in a large collection of sequences. The system searches exhaustively for all significantly compressing patterns where patterns may consist of symbols and skips or wildcards. We demonstrate a possible application of the system by detecting interesting patterns in a Dutch and an English corpus.**

I. Introduction

SYSTEMS that analyze large collections of sequential data, such as when searching for regularities in collections of texts, place strict requirements on the efficiency. Trivial implementations of such systems often yield correct results, but either take too long or use too much internal memory. These trivial implementations lead to limitations on the size of the data set that can be handled practically.

In this paper we propose a novel implementation of a system that can be used to search for regularities in sequential data. We show the practical applicability by searching for patterns in large text collections in two different languages. At the time of publication the source code of the implementation is publicly available on the web at http://ilk.uvt.nl/~stehouwer/.

To search the data sets efficiently, we initially used a data structure called a suffixtree. Suffixtrees are well-known data structures with many applications within the fields of bioinformatics, natural language processing, and many others. Ukkonen introduced an efficient online construction algorithm for suffixtrees in [9].

More recently, a similar data structure called suffixarray is often used instead of suffixtrees to search sequential data. A suffixarray is an ordered list of all suffixes in a sequence. This data structure was introduced in [6].

The reason for choosing suffixarrays instead of suffixtrees is the relatively large memory requirements of the suffixtrees. Gusfield discusses suffixtrees, their construction, and complexity requirements in great detail in his book [4]. The worst-case complexity of memory utilization of a suffixtree is $\Theta(m|\Sigma|)$ with m the length of the sequence and Σ the alphabet. This results in a suffixtree that can be searched in linear time (linear in the size of the search sequence). In contrast, suffixarrays are linear in their space utilization, regardless of the alphabet $|\Sigma|$.

In practical terms, a suffixtree build on the first 500.000 sentences from the British National Corpus took up 2.2GB. The suffixarray on the other hand only used 0.25 GB of main memory to be able to access the same data. From the last 10.000 sentences of the same corpus we generated a set of 70.000 queries. The suffixtree answered these queries in about 3 minutes, where the suffixarray system took around an hour.

Whereas usable suffixtree implementations can be found online[1], implementations of suffixarrays that we found available all have drawbacks. Often, the implementation limits the maximum size of the alphabet, which makes them unsuitable for our use.

In [1] several improvements to the original suffixarray data structure are combined to form what the authors call the enhanced suffixarray. One enhancement added to the enhanced suffixarray is the implementation of an encoding of an implicit tree structure. It is this implicit suffixtree structure that we will use in the application we describe here that allows us to find interesting patterns.

In this paper we will start by outlining the suffixarray data structure as well as its enhanced version. We concentrate on the implicit suffixtree structure that is made available through the enhancements. We will then show an application that finds patterns with skips (also called wildcards) in natural language data using suffixarrays by using the implicit suffixtree structure. Based on this application we show some experimental results on Dutch and English and in our conclusion we will suggest other possible applications of this kind of pattern finding.

II. Suffixarrays

In this section we will describe the techniques behind suffixarrays and their enhanced version.

A. *Regular Suffixarrays*

As introduced in [6] suffixarrays are relatively simple data structures that contain a lexicographically ordered list of all suffixes of the input sequence. The array does not contain explicit copies of the suffixes, but stores information on the suffixes implicitly in the form of an index on the original input. This means that much less memory is needed to store the suffixes compared to suffixtrees.

To build a suffixarray, we initialize an array of indexes describing all suffixes of the input sequence. This array is then

[1] See for instance http://ilk.uvt.nl/~menno/research/software/suffixtree.

sorted. When using a regular, efficient, sort algorithm, sorting takes $n \log n$ time. However, the ordering of the prefixes (of the suffixes stored in the array) can depend on multiple consecutive positions in the original input. This means that we have to perform sequence comparison which may need symbol comparisons of at most n positions. This results in a naive construction time of $n^2 \log n$ for a sequence of length n.

During the last several years, many sorting algorithms have been developed that can construct a suffixarray more efficiently in time requirements. The fastest algorithms run in $\Theta(n)$ time. These build a suffixtree first (which can be done in linear time) and then obtaining the sorted suffixes by a simple traversal of the suffixtree. Unfortunately, these algorithms need working space of at least $15 \times n$ [7].

In [7] a reasonably fast algorithm is proposed that needs working space of only $5 \times n$. This algorithm works by partially sorting the array into buckets which have the same x position prefix and afterwards sorting each of these buckets with a blind trie. This strategy is called *deep-shallow sort*.

An interesting aspect of suffixarrays is that suffixes that share a common prefix are grouped together in the array. This means that it can be used to locate the position and number of all infixes of an input sequence. This is done by finding all suffixes that start with the given infix. Since they are grouped together, they can be found efficiently.

B. Enhanced Suffixarrays

Several extensions to the regular suffixarrays have been proposed. The ones we will use (and have implemented) are described in [1]. The extensions combined with a regular suffixarray makes up an enhanced suffixarray. This data structure provides, among others, a different way of viewing the data, which is very similar to a suffixtree.

The enhancements store some information explicitly, which means that they require some additional storage in the shape of arrays. The first of those is the longest-common-prefix (or lcp) array. This array, parallel to the regular suffixarray, denotes the size of the prefix shared with the previous element. For instance, if the first element is *aard* and the second *aardvark* then the second element would have an lcp value of 4 as it shares a prefix of length four with the previous element in the suffixarray. This lcp array can be efficiently constructed in a single pass over the regular suffixarray.

The lcp values can be used to define the intervals, so called lcp intervals. An lcp interval can be seen as defining the interval corresponding to range of suffixes (in the suffixarray) with a specific prefix. An interval $[i \dots j], 0 \leq i < j \leq n$ with n the length of the sequence, is an lcp interval of the lcp value l if the following conditions hold (lcptab denotes the lcp array):

1) lcptab$[i] < l$,
2) lcptab$[k] \geq l$ for all k with $i + 1 \leq k \leq j$,
3) lcptab$[k] = k$ for at least one k with $i + 1 \leq k \leq j$,
4) lcptab$[j + 1] < l$.

TABLE I

AN ENHANCED SUFFIXARRAY ON THE SEQUENCE S = *acaaacatat* INCLUDING ITS LCPTAB AND CHILDTAB. THE FIELDS 1, 2 AND 3 OF THE CHILDTAB STAND FOR THE UP, DOWN AND NEXTINDEX FIELDS RESPECTIVELY. THIS EXAMPLE IS TAKEN FROM [1].

			childtab			
i	suftab$[i]$	lcptab$[i]$	1.	2.	3.	S[suffix]
0	2	0		2	6	*aaacatat$*
1	3	2				*aacatat$*
2	0	1	1	3	4	*acaaacatat$*
3	4	3				*acatat$*
4	6	1	3	5		*atat$*
5	8	2				*at$*
6	1	0	2	7	8	*caaacatat$*
7	5	2				*catat$*
8	7	0	7	9	10	*tat$*
9	9	1				*t$*
10	10	0	9			*$*

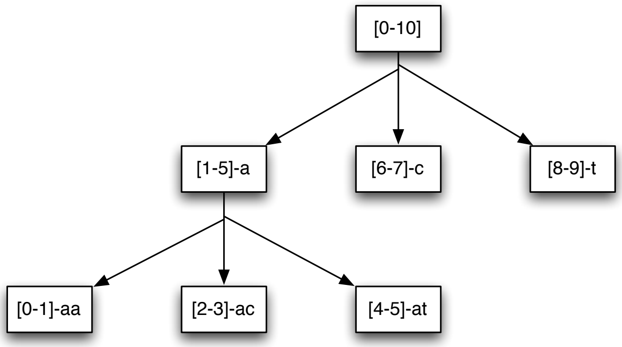


Fig. 1. An enhanced suffixarray on the sequence S = *acaaacatat* produces the lcp interval tree shown. This example is taken from [1].

The lcp intervals can have smaller lcp intervals embedded within them recursively. These recursive intervals can be seen as a tree structure and is called an lcp interval tree. This lcp interval tree is implicit. Furthermore, it has the same structure as the suffixtree if it were built based on the sequence in the suffixarray.

We would like to be able to access the implicit suffixtree structure in an efficient way. In order to do this we store the jumps through the suffixarray that we need for top-down traversal of the implicit suffixtree in an extra support array. This extra support array is called the child table in [1]. With this extra information we can determine the longest-common-prefix and its child intervals for each interval with a simple array lookup. For this to work we need to start with a valid interval. Luckily the interval $[0 \dots n]$ is always valid.

The construction of the child table can be done, just like the lcp value array, in a single pass over the suffixarray. However, building the child table depends on the presence of the lcp table, so after constructing and sorting the regular suffixarray we have to perform two more passes over the suffixarray to fill the support structures that allow access to the implicit suffixtree structure.

An example of a regular suffixarray enhanced with the lcp values and the childtab can be found in Table I. We also show the corresponding lcp interval tree in Figure 1.

III. Related Programs

In order to understand the context of the suffixarray-based patternfind program we will briefly describe related systems that are available.

The Ngram Statistics Package [2] can perform statistical tests on n-grams sampled from a window of size k. Effectively, this means that the package can look for sparse n-grams. It supports many statistical tests such as the Dice Coefficient, Fishers test, and Mutual Information. The Ngram Statistics Package is available on Pedersen's website at http://www.d.umn.edu/~tpederse/nsp.html.

Daciuk has developed several Finite-State based libraries and tools, including one for compressing FSA-based language models [3]. Such compression is related to the approach we take to find patterns with our patternfind software. This software is available at http://www.eti.pg.gda.pl/katedry/kiw/pracownicy/Jan.Daciuk/personal/fsa.html.

There are also software packages available for extracting significant n-grams and for performing statistical subsequence reduction as described in [8]. One implementation is that of Zhang, which is available at http://homepages.inf.ed.ac.uk/lzhang10/ngram.html.

PAFI is a piece of software for finding frequents patterns in large and diverse databases. In [5] some of the (graph-based) techniques that underly the system are described. The program is available from http://glaros.dtc.umn.edu/gkhome/pafi/overview together with related software.

IV. Implementation

Our implementation of the suffixarray is done in template-based C++. This leads to an extremely flexible implementation while making only limited demands on the container type. The template types used in the suffixarray must support only the following basic functionality:

1) The subtype has to support the following operators:
   a) The comparison operator $<$,
   b) The comparison operator $>$,
   c) The comparison operator $!=$,
   d) The comparison operator $==$,
   e) The assignment operator $=$,
   f) The increment operator: $++$,
   g) The output operator: $<<$.
2) The type contents must be accessible via several ways:
   a) Via the $[x]$ construct,
   b) Via the iterators x.begin() and x.end(),
   c) Via the const_iterators x.begin() and x.end().

Furthermore, when building a suffixarray, the input sequence must end with a unique element that is largest when compared to any other element in the sequence according to the comparison operators. This additional requirement allows us to eliminate several bounds checks in the heart of the sorting code.

In our implementation of the suffixarray algorithm, space utilization for a data collection of length n is

$$n \times \text{sizeof(index)} + 4 \times n * \text{sizeof(symbol)} + \text{exceptions}$$

This includes the additional arrays that are used to implement the enhancements as described in Section II-B.

In practice, building the suffixarray using our implementation is very time efficient (approximately 2–3 minutes for 1 million natural language sentences). However, for input sequences that have a high longest-common-prefix (lcp), our implementation will not be particularly efficient, due to the nature of the sorting stage. We use a deep-shallow sorting strategy with a blind trie, which was introduced by Manzini and Ferragina in [7]. Natural language data, however, which is our area of interest, is sorted very efficiently as it is by nature low in lcp.

We have implemented the enhancements that are needed for the implicit suffixtree structure as described in [1]. The hints and algorithms described in the article form the basis of the practical implementation.

V. Programs

The package contains three user programs. The usage of these programs will be described in more detail in the next three sections.

A. Patternfind

The pattern finding program looks for frequently occurring skip-grams as patterns. A skip-gram is similar to a regular n-gram with the possibility of containing non-consecutive skip positions. A skip represents a wildcard, which may match any number of arbitrary symbols. In article we will refer to such skip positions in patterns as **SKIP**. In Figure 2 a simple skip-gram is shown visually in a simple tree-structure.

1) Implementation: The pattern finding is done as an exhaustive, breadth-first search over the implicit suffixtree with pruning. During the search, the internal representation keeps track of the pattern represented so far, for instance *A *SKIP*A*, together with a set of [lcp-interval, depth] pairs.

An [lcp-interval, depth] pair represents a path through the implicit suffixtree. The lcp interval has an associated lcp value, which may be larger than the number of steps that have been taken to reach the lcp interval in the specific path. For instance, if the sequence S = *abcabc* is turned into a suffixarray, the lcp interval $[0, 1]$ has an lcp value of three as the first three characters of the two suffixes starting with *a* will be *abc*. Because we would like to be able to represent the step *a* we also associate a depth with the lcp interval to indicate the exact position in the implicit suffixtree. This example is illustrated in Table II.

The suffixarray is searched using the patterns, where for each pattern, all single steps are taken. A single step can be an explicit symbol as defined by the pattern or a skip position if the previously taken step was not a skip. Only continuations of the (sub-)pattern that are directly reachable from the set of [lcp-interval, depth] pairs are considered.

TABLE II
ILLUSTRATING THE SUFFIXARRAY, INCLUDING THE LCP TABLE, OF THE SIMPLE SEQUENCE *abcabc*.

i	suftab[i]	lcptab[i]	suffix
0	3	0	*abcabc$*
1	0	3	*abc$*
2	4	0	*bcabc$*
3	1	2	*bc$*
4	5	0	*c$*
5	2	1	*cabc$*
6	6	0	*$*

Obviously applying this approach directly results in an exponential explosion in the number of different patterns that are found in a dataset. To keep this amount manageable a threshold can be set. Based on this threshold, only interesting patterns are retained.

For each pattern discovered, a prune value is computed. This value depends on the compressibility score, which consists of the number of items in the pattern minus one, not counting the skip positions, multiplied by the number of times it occurs in the sequence. In this case, the compression rate of a uni-gram pattern is zero. To remove this limitation, we use the frequency in the sequence for patterns of length one as the compressibility score.

Based on the computed prune value, we decide whether the pattern should be pruned or not. If the score of the new pattern exceeds the prune value it is added to the pattern list for the next round.

After taking a few of such steps a pattern emerges. In Figure 2 we show a very simple implicit suffixtree and highlight the effect of the pattern *A *SKIP* A* on that example. All dark-grey nodes in the example represent [lcp-interval, depth] in the suffixarray that would occur in the cloud of the pattern in this example.

After having considered all valid steps during the pattern search phase, the remaining patterns (which have a score above the pruning threshold) are written to output. Only the patterns above a certain threshold are given to the user. The threshold can be set separately from the prune value.

The program itself is implemented in template-based C++ and in its distributed form can be applied to a sequence of text-based word tokens which are read from a file. Internally, these are mapped to a list of numbers. The suffixarray is built on the sequence of numbers. This list of numbers can be transformed back into a sequence of words using the one-to-one mapping, which is what happens during the output of the patterns. This transparent, internal mapping is done, because the comparison operators on integers are significantly faster than those on sequence objects. Additionally, a list of integers occupies less memory[2].

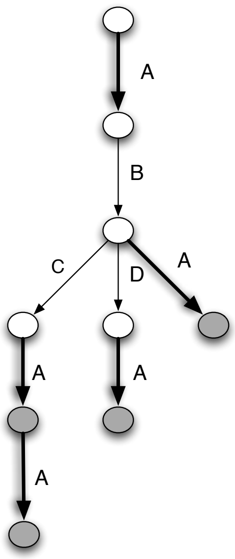


Fig. 2. An example of the pattern *A *SKIP* A* found in a simple tree. Notice how the **SKIP** can have different lengths. All dark-grey nodes represent members of the [lcp-interval, depth] set associated with the pattern *A *SKIP* A*.

TABLE III
THE OUTPUT OF THE PATTERNFIND PROGRAM WHEN RUN WITH THE −HELP OPTION.

```
patternfind -h
Usage: ./patternfind[OPTION]...
This program reads in a corpus and stores it in a suffixarray.
It then searches for all significant patterns in the corpus.
  -h, --help        Show this help and exit
  -f, --file FILE   Filename of the corpus to be read
  -p, --prune PRUNE-VALUE    The value at which to start pruning the found
          patterns.
        Determined by the number of occurrences and the size of the pattern.
  -o, --output PRINT-VALUE     The value at which to start printing the found
          patterns,  works the same as the prune value.
        If it is smaller or equal to the prune value it has no extra effect.
  -s --smallestskip SKIP    Minimum number of positions to skip for a SKIP
          part of a pattern.
  -l --largestskip SKIP    Maximum number of positions to skip for a SKIP
          part of a pattern.
```

2) Usage: The program that searches for compressing skip-grams contained in the suffixarray is called patternfind. As described above, this program identifies patterns that occur frequently and hence lead to a compression higher than a threshold. When patternfind is run with the `-help` option it gives the output as shown in Table III.

TABLE IV
THE OUTPUT OF THE PATTERNPOSITIONS PROGRAM WHEN RUN WITH THE −HELP OPTION.

```
patternpositions -h
Usage: ./patternpositions[OPTION]...
This program reads in a corpus and stores it in a suffixarray.
        It then reads in a file of patterns and for each of those patterns
        returns all occurring positions in the corpus.
  -h, --help        Show this help and exit
  -f, --file FILE   Filename of the corpus to be read
  -s --smallestskip SKIP    Minimum number of positions to skip for a SKIP
          part of a pattern. Should be the same value as used in patternfind.
  -l --largestskip SKIP    Maximum number of positions to skip for a SKIP
          part of a pattern. Should be the same value as used in patternfind.
  -p, --patterns FILE    The file containing the patterns to output the
          positions of.
         As output by patternfind. I.e. patternfind options > patternfile;
        patternpositions options -patterns patternfile
```

[2] Personally, we run a custom version of the code that takes as input a pre-mapped corpus that we have to manually map back and forth. By doing this we do not have to load the mapping in memory as well. For usability reasons we automated this process in the released tool.

The most important parameters specify an input sequence (in a file) and a prune value. The other values are optional and only modify the default behavior of the program. For example, we can look at the patterns found in the included README file by executing the following command. We will set the prune value to 10 and limit the skip to skips of the size 1–3.

```
./patternfind -f ../README -p 10 -s 1 -l 3
```

All status messages are written to standard error and all patterns are written to standard output. A pattern is presented together with a number representing its compressibility score on which pruning was done. **SKIP** denotes a skip position. At that point the pattern skips a flexible space of, in this case [1–3] positions.

B. Patternpositions

The patternpositions program searches a sequence for occurrences of patterns such as those output by the patternfind program. To specify the exact behavior of the skips, it takes the same options as the patternfind program. It takes one extra argument, namely the patterns file from which all the patterns will be read. The output of its help function can be seen in Table IV.

For each pattern in the pattern file, patternposition will print all the positions of occurrences of that pattern in the corpus. We can for instance run the program on the patterns we found in the README file with patternfind in the following way:

```
./patternpositions -f ../README -s 1 -l 3\
  -p ../README.patterns
```

Just like with the patternfind program we print all results to the standard output and all the status messages to the standard error.

C. Suffixarray

The final program included in the distribution is called suffixarray. This program has only one option, the file containing the sequence which should be used to build a suffixarray. This file should be a plain text file containing a white space separated sequence of tokens, such as words.

Running the program will result in a suffixarray being build from the specified sequence after which the program waits for input. At this point the program will answer simple n-gram queries by reporting the number of times the typed n-gram occurs in the sequence contained in the suffixarray. In these n-grams it is allowed to use simple single-positions wildcards, represented by the token *.

VI. Experimental Results

To show practical use, we performed preliminary pattern finding experiments on two natural language corpora: an English-language corpus, the British National Corpus (BNC) and a Dutch-language corpus, our own collection of texts of local newspapers known internally as the ILK-BDEDGE corpus.

The British National Corpus is a corpus of around 100 million words of both spoken and written English. We use

TABLE V
THE 40 HIGHEST RANKING PATTERNS FOUND IN ONE MILLION SENTENCES OF THE BRITISH NATIONAL CORPUS. THE SCORE OF THE PATTERN IS SHOWN AS WELL.

```
460718 the *SKIP* of
308872 of the
252720 the *SKIP* .
240072 the *SKIP* ,
235770 , *SKIP* ,
208494 , *SKIP* the
201894 . The
195756 the *SKIP* of the
195478 in the
191296 the *SKIP* the
176920 , and
173836 . "
162396 of *SKIP* ,
158076 . *SKIP* the
156170 of *SKIP* .
154302 . *SKIP* ,
146010 a *SKIP* of
138120 the *SKIP* and
127728 to *SKIP* the
122818 , the
111534 to the
108086 to *SKIP* .
107156 and *SKIP* .
104474 , *SKIP* and
98664 in *SKIP* ,
97588 in *SKIP* .
96626 of *SKIP* and
96318 and *SKIP* ,
96063 the *SKIP* of *SKIP* .
94191 the *SKIP* of *SKIP* ,
93480 " *SKIP* "
90878 . *SKIP* is
88758 a *SKIP* ,
88290 and *SKIP* the
88234 a *SKIP* .
85738 , *SKIP* a
83964 . *SKIP* was
81498 the *SKIP* to
81458 to *SKIP* ,
80346 , *SKIP* of
```

TABLE VI
10 PATTERNS FOUND IN ONE MILLION SENTENCES OF THE BRITISH NATIONAL CORPUS. THESE PATTERNS ARE IN THE MIDDLE OF THE SCORE-RANKING. THE SCORE OF THE PATTERN IS SHOWN AS WELL.

```
1872 with the *SKIP* a
1872 was *SKIP* in a
1872 to be *SKIP* , and
1872 the *SKIP* between *SKIP* and *SKIP* .
1872 that it *SKIP* not
1872 over the *SKIP* the
1872 of all the
1872 of *SKIP* in the *SKIP* the
1872 in the *SKIP* of *SKIP* "
1872 in *SKIP* will
```

TABLE VII
10 PATTERNS FOUND IN ONE MILLION SENTENCES OF THE BRITISH NATIONAL CORPUS. THESE PATTERNS ARE THE LAST OF THE SCORE-RANKING. THE SCORE OF THE PATTERN IS SHOWN AS WELL.

```
1002 . *SKIP* the second
1002 . *SKIP* the *SKIP* would
1002 . *SKIP* most of
1002 . *SKIP* help
1002 . " *SKIP* , *SKIP* he *SKIP* . "
1002 , *SKIP* to *SKIP* them
1002 , *SKIP* the *SKIP* this
1002 , *SKIP* fact that
1002 , " *SKIP* Mr
1001 . " *SKIP* , " he said ,
```

TABLE VIII
THE 40 HIGHEST RANKING PATTERNS FOUND IN ONE MILLION SENTENCES OF THE CORPUS OF REGIONAL DUTCH NEWSPAPERS. THE SCORE OF THE PATTERN IS SHOWN AS WELL.

```
286710  .       "
263612  van     de
238974  de      *SKIP*    .
237896  de      *SKIP*    van
207394  .       De
171850  de      *SKIP*    de
166406  .       *SKIP*    de
147504  in      de
132682  van     *SKIP*    .
128690  te      *SKIP*    .
120294  de      *SKIP*    van     de
119124  ,       *SKIP*    de
113170  .       *SKIP*    is
112282  het     *SKIP*    van
108446  de      *SKIP*    ,
103996  in      *SKIP*    .
102382  het     *SKIP*    .
95998   een     *SKIP*    .
94398   .       Het
92986   de      *SKIP*    in
92720   van     het
80518   .       *SKIP*    van
79314   .       *SKIP*    het
77822   en      *SKIP*    .
77746   (       *SKIP*    )
77211   van     de      *SKIP*    .
76828   het     *SKIP*    de
74458   "       ,
74276   '       *SKIP*    '
72732   op      de
72714   van     *SKIP*    ,
72386   een     *SKIP*    van
72266   de      *SKIP*    het
72176   in      het
71564   ,       *SKIP*    ,
68942   de      *SKIP*    en
67518   ,       *SKIP*    en
67102   voor    de
66948   aan     de
63842   ,       *SKIP*    .
```

TABLE IX
10 PATTERNS FOUND IN ONE MILLION SENTENCES OF THE CORPUS OF REGIONAL DUTCH NEWSPAPERS. THESE PATTERNS ARE IN THE MIDDLE OF THE SCORE-RANKING. THE SCORE OF THE PATTERN IS SHOWN AS WELL.

```
1515    op  het *SKIP*    een
1515    nog *SKIP*    van de
1515    naar    *SKIP*    .   Het
1515    is  het een
1515    is  *SKIP*    de  *SKIP*    van de  *SKIP*    .
1515    het *SKIP*    in  *SKIP*    een
1515    een *SKIP*    .   *SKIP*    :
1515    door    de  gemeente
1515    de  *SKIP*    .   *SKIP*    uit
1515    bestuur van de
```

TABLE X
10 PATTERNS FOUND IN ONE MILLION SENTENCES OF THE CORPUS OF REGIONAL DUTCH NEWSPAPERS. THESE PATTERNS ARE THE LAST OF THE SCORE-RANKING. THE SCORE OF THE PATTERN IS SHOWN AS WELL.

```
1002  .  *SKIP*  dat  *SKIP*  er
1002  .  *SKIP*  daarom
1002  .  *SKIP*  acht
1002  .  *SKIP*  Volgens  *SKIP*  is
1002  .  *SKIP*  J.  van
1002  ,  *SKIP*  of  *SKIP*  ,
1002  '  ,  '  *SKIP*  '  *SKIP*  '  *SKIP*  '
1002  '  *SKIP*  of  *SKIP*  '
1002  "  Wij  zijn
1001  ,  2  .  *SKIP*  ,  3  .  *SKIP*  ,
```

a 1 million sentence chunk for our pattern finding experiment. This chunk consist of around 21 million words.

The ILK-BDEDGE is a corpus consisting of material from the Brabants Dagblad, Eindhovens Dagblad and De Gelderlander. These are all regional, Dutch, newspapers. Again, we took a chunk of 1 million sentences from the corpus to perform experiments on. This chunk consists of around 16 million words. The average sentence length is somewhat smaller than that of the BNC.

On each of the 1 million sentences we ran the pattern finder program with a prune value of 1000. The 40 highest ranked patterns found on the BNC are shown in Table V. Likewise for the corpus of Dutch local newspapers, we show the 40 highest ranked patterns in Table VIII. For English we find 23,705 patterns and for Dutch 18,004 patterns in total that make the threshold.

The program took slightly less than 32 hours to run on the ILK-BDEDGE corpus part we used. During that time its memory usage peaked at approximately 700MB.

We see in these tables that function words such as determiners, that delimit the overall structure of the sentences, are found the most. Unsurprisingly, all these words together with punctuation marks are amongst the most frequent tokens in both languages. For instance in English we find as the most compressing pattern *the *SKIP* of* and in the Dutch list the corresponding pattern *de *SKIP* van* also scores very high.

When we look at the patterns found in the middle of the pack we see less generic and possibly more useful patterns, such as *with the *SKIP* a*. These patterns from the middle can be found in Table VI for English and Table IX for Dutch.

The lowest-scoring patterns are shown in Table VII for English and Table X for Dutch. These patterns start with punctuation and after that specialize in a specific way. Patterns with very low, compared to the size of the corpus, scores are not generally very informative. These patterns could be removed automatically by increasing the threshold.

VII. CONCLUSION

In this paper, we have described a package that contains an efficient, flexible and practical implementation of the suffixarray data-structure. Three programs are provided, allowing for efficient search in a large sequence of symbols and for finding interesting, compressing patterns.

The program that searches for compressing patterns has been applied to a collection of natural language texts. The patterns found describe the global structure of the sentences in which they occur. These patterns can be used by linguists to help them find naturally occurring constructions in language.

With respect to future work, this system could be applied in different areas. For instance, another application of the patternfind program can be found in the area of morphology. Instead of providing a collection of sentences, as shown here, a collection of words is provided to the system and patterns that occur regularly within words can be identified. This may lead to automatically found descriptions of the morphology of words.

Another possible application would be to apply the found patterns of several languages to a parallel corpus. The patterns can be used to identify often occurring patterns in the different languages, which can then be used to align translations of texts. Also, in the same line, the aligned patterns can be used to enrich a phrase-based machine-translation system.

Additionally, we would like to write several extensions to the programs, the simplest of them being a small program that identifies contexts for a given pattern in a specific position. This information is directly accessible in the suffixarray data structure, but cannot easily be identified with the current programs. Another, more complex addition we would like to write, is a program that given a pattern finds the most pertinent content that matches the skips in that pattern.

Finally, we would like to provide a version that is annotation-aware, allowing it to find patterns not only with skips that match any symbol, but taking specific annotation layer items into account. This could lead to patterns such as *the N of the N*, where the *N* specifies a noun as described by an annotation layer. This final improvement should also be extended to the program that tries to find the pertinent content that fills up skips in particular patterns.

References

[1] M.I. Abouelhoda, S. Kurtz, and E. Ohlebusch. Replacing suffix trees with enhanced suffix arrays. *Journal of Discrete Algorithms*, 2(1):53–86, 2004.

[2] S. Banerjee and T. Pedersen. The design, implementation, and use of the Ngram Statistic Package. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 370–381, Mexico City, February 2003.

[3] J. Daciuk and G. van Noord. Finite automata for compact representation of language models in nlp. In *CIAA '01: Revised Papers from the 6th International Conference on Implementation and Application of Automata*, pages 65–73, London, UK, 2002. Springer-Verlag.

[4] D. Gusfield. *Algorithms on Strings, Trees and Sequences*. University of Cambridge, Cambridge, 1997.

[5] Michihiro Kuramochi and George Karypis. An efficient algorithm for discovering frequent subgraphs. Technical report, IEEE Transactions on Knowledge and Data Engineering, 2002.

[6] U. Manber and G. Myers. Suffix arrays: a new method for on-line string searches. In *SODA '90: Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*, pages 319–327, Philadelphia, PA, USA, 1990. Society for Industrial and Applied Mathematics.

[7] G. Manzini and P. Ferragina. Engineering a lightweight suffix array construction algorithm. *Algorithmica*, 40:33–50, 2004.

[8] M. Nagao and S. Mori. A new method of n-gram statistics for large number of n and automatic extraction of words and phrases from large text data of japanese. In *In COLING-94*, pages 611–615, 1994.

[9] E. Ukkonen. On-line construction of suffix trees. *Algorithmica*, 14(3):249–260, september 1995.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Entity Summarisation with Limited Edge Budget on Knowledge Graphs

Marcin Sydow[1,2], Mariusz Pikuła[1], Ralf Schenkel[3], Adam Siemion[1]
[1]Polish-Japanese Institute of Information Technology, Warsaw, Poland
[2]Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland
[3]Max-Planck Institut fuer Informatik, Saarbruecken, Germany
{msyd, mariusz.pikula}@poljap.edu.pl, schenkel@mpi-inf.mpg.de, adam.siemion@gmail.com

***Abstract*—We formulate a novel problem of summarising entities with limited presentation budget on entity-relationship knowledge graphs and propose an efficient algorithm for solving this problem. The algorithm has been implemented together with a visualising tool. Experimental user evaluation of the algorithm was conducted on real large semantic knowledge graphs extracted from the web. The reported results of experimental user evaluation are promising and encourage to continue the work on improving the algorithm.**

I. Introduction and Motivation

KNOWLEDGE graphs are useful for representing semantic knowledge, often automatically extracted from open domains such as the web [1] in the form of entity-relationship triples. In this data model the nodes represent entities (e.g. a director or actor, in the movie domain), and directed arcs represent binary relations between the entities (e.g. “directed”,”acted in”, etc. in the movie domain). Multiple arcs between nodes are allowed (as a person can be a director and a producer of the same movie, for example) resulting in a directed multi-graph (fig. 1). There can be weights attached to nodes or arcs in the knowledge graph that reflect some additional information, for example “witness count” – reflecting the frequency of encountering the triple in the corpus, so that an arc with high witness count could be regarded as more “important”.

A standard example of this model is the RDF[1] data format with its SPARQL[2] query language, where a query can be viewed as a sub-graph pattern that is matched with the knowledge base to produce the results.[3]

Structured query languages for knowledge graphs such as SPARQL allow for semantic search and are very expressive, however they are quite complex for unexperienced users and they also assume some prior knowledge about the domain (e.g. names of relations, etc.) which limits their applications.

Simply speaking, there are currently two extremes in search paradigms: popular and simple keyword-based search interfaces that do not support semantic search, and prototype semantic search systems that enable very precise querying but demand a lot of knowledge and experience from the user.

Consequently, there is a gap between these two extremes: it would be ideal to have a tool that enables search over semantic knowledge bases and, at the same time, is very simple and does not assume any prior experience or knowledge.

In this paper we aim at filling this gap. Namely, we formulate a novel problem of answering “precis” queries on knowledge graphs, propose an efficient algorithm for solving it, and report on prototype implementation together with preliminary experimental results obtained on real data. Similar problem of “precis” queries was previously considered in the context of relational databases [4], from which we borrowed the name “precis” (meaning a short summary about a person, etc.) and in the context of query-dependent [3] and constrained [5] summarisation of XML documents.

A. Related Work

Summarization has been intensively studied for text documents, for example by Wan et al.[9], specificly for scientific papers by Hassan et al., [2], and for multiple documents by Wan [8]. Zhang et al. [10] (as a recent example for a large set of similar papers) consider the problem of creating concise summaries for very large graphs such as social networks or citation graphs; in contrast to this, our work aims at summarizing information around a single node in a graph. Ramanath et al. [6] propose methods for summarizing tree-structured XML documents within a constrained budget.

Similar problem to the one discussed here, but with a special focus on diversification was recently discussed in [7].

II. Problem Formulation

Assume a user would like to know “something about” an entity (e.g. “Woody Allen”) but does not know anything except its name to start searching.

It would be desirable that a semantic search system accepts the query ”Woody Allen” and returns some fragment of the knowledge graph that “reasonably summarises” this entity.

Unfortunately, such kind of query is not supported by the SPARQL standard.

The most equivalent SPARQL query seems to be *“Woody Allen” ?r ?x* that requires returning the whole subgraph of the knowledge base that is in the one-hop distance from the “Woody Allen” node. However, such a solution might be not the most desired one for at least two reasons:

[1] http://www.w3.org/RDF

[2] http://www.w3.org/TR/rdf-sparql-query

[3] Formally, data in RDF consists of triples: subject-predicate-object, but this is mathematically equivalent to the multi-graph model described above

978-83-60810-27-9/09/$25.00 © 2010 IEEE

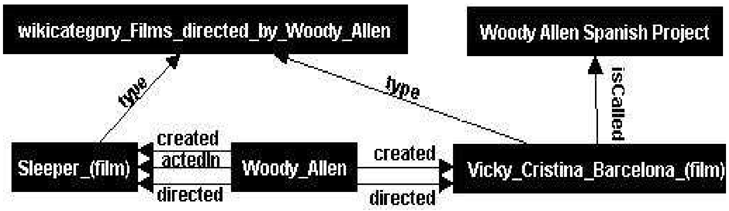

wikicategory_Films_directed_by_Woody_Allen
Woody Allen Spanish Project
type
type
isCalled
created
actedIn
directed
Sleeper_(film)
Woody_Allen
created
directed
Vicky_Cristina_Barcelona_(film)

Fig. 1: An excerpt from a semantic knowledge graph extracted from the IMDB database, concerning the movie domain.

1) the result may be too large to be comprehended by a user (e.g. in the knowledge graph extracted from the imdb database that we use, concerning movies, the "Woody Allen" node is adjacent to over 170 different arcs).
2) there may be "interesting" pieces of information concerning "Woody Allen" that are "closer" (in terms of arc weights) to the entity but are a few hops away from it in the knowledge graph (fig. 4,5)

To address these issues we introduce two elements to our model of "precis" query. First, we introduce a parameter $k \in N$ that models limited "budget" of user comprehension or display device capacity, etc. that specifies the upper bound on the number of the arcs in the presented result. Second, we introduce a novel but natural notion of distance between an arc a and node x in the knowledge graph. We assume it is the minimum sum of arc weights on a path connecting x and a, including the weight of a.

We propose the following specification of the "precis" query problem on knowledge graphs:

INPUT: B (knowledge base) – a multi-digraph with positive, real weights on arcs (considered as distance measure – currently, we use $1/\mathit{witnessCount}$ as the weight value); x (entity under interest) – a node of B; $k \in N$ (limit on arcs)

OUTPUT: subgraph D of B, containing at most k arcs of B, together with their end nodes, that are "closest" to x with respect to the arc-node distance

III. THE ALGORITHM

The problem is similar to some very well known ones such as the single source shortest paths or incremental search, however the specific conjunction of constraints such as multiple and weighted arcs, the notion of arc-node distance and limited presentation budget, taken together, make it a unique, novel graph problem, up to the author's knowledge. We propose the following algorithm to solve the problem (fig. 2).

It can be obiously viewed as a modification of the Dijkstra's single-source shortest paths algorithm adapted to the formulation of our problem. In each iteration an arc is added to RESULT thus the algorithm always stops after k iterations, at most. If we assume that there are n edges in the radius k from x in B and that comparison of *distance* value is the dominating operation, time complexity is $O(nlog(n))$ if we use a hashset implementation for RESULT and even if we use ordinary Heap for implementing PQ (the algorithm could be faster, though, if Fibonacci Heap is used instead).

```
visitTop-kClosestArcsInMultiGraph(B,x,k)

forEach a in radius k from x: a.dist := "infinity"
forEach a adjacent to x: {a.dist := a.weight; PQ.insert(a)}
while( (RESULT.size < k)
and ((currentArc = PQ.delMin()) != null) )
  forEach a in currentArc.adjacentArcs:
    if (not RESULT.contains(a)) then
      a.dist := min(a.dist, (a.weight + currentArc.dist))
       if (not PQ.contains(a)) then PQ.insert(a)
       else PQ.decreaseKey(a,a.dist)
  RESULT.add(currentArc)
return RESULT
```

Fig. 2: Algorithm for computing "precis" queries. We assume that each arc a has two real attributes: *weight* and *distance* as well as *adjacentArcs* attribute that keeps the set of arcs sharing a node with a (except a). *PQ* is a min-type priority queue for keeping the arcs being processed, with the value of weight serving as the priority and *RESULT* is a set. *PQ* and *RESULT* are initially empty. We also assume that "infinity" is a special numeric value being greater than any real number.

IV. Experimental Results

The algorithm has been implemented, integrated with a graph-visualising tool, and applied to the two real datasets concerning the domains of movies and books, respectively (figure 3). The selected results are visualised on fig. 4 and 5

A. User Evaluation Experiment

We also conducted a user evaluation experiment aiming at assessing the quality of results of the presented algorithm and collecting feedback in order to improve the algorithm.

We selected 20 active actors from the IMDB dataset, generated the summarisations for them, for two different levels of the egde budget k: low ($k = 7$) and high ($k = 12$) (20 results for each level). Next, we asked about 10 anonymous evaluators, who did not know details about the algorithm, for assessing the summaries. Technically, the summaries computed for 20 actors and for 2 different budget levels were presented to the evaluators by a web interface. The evaluators assessed them by answering the following questions:

- How useful do you find the result as a small entity summarisation with a very limited number of facts (edges) to be presented? ("good","acceptable","poor","useless").
- How many interesting/irrelevant/missing facts are in the presented summary? (three separate questions; possible answers: "almost all","some", "hardly any")

The evaluators could also give optional textual explanations of their answers. We collected about 70 assessments.

dataset	out-nodes	in-nodes	edges	relations	weights on arcs	source
IMDB-1	59013	106682	536455	73	1/witness count	www.imdb.com
LT-1	17254	45535	644055	12	1/witness count	librarything.com

Fig. 3: Datasets used for preliminary experiments. The IMDB dataset was further used for user evaluation experiment

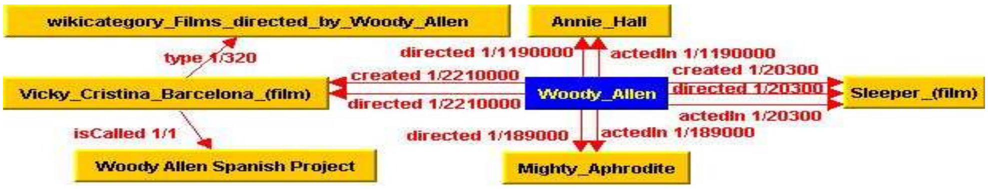


Fig. 4: An example of running the algorithm for precis query "Woody Allen" and $k = 11$ on IMDB-1 dataset. Weights represent $1/witnessCount$

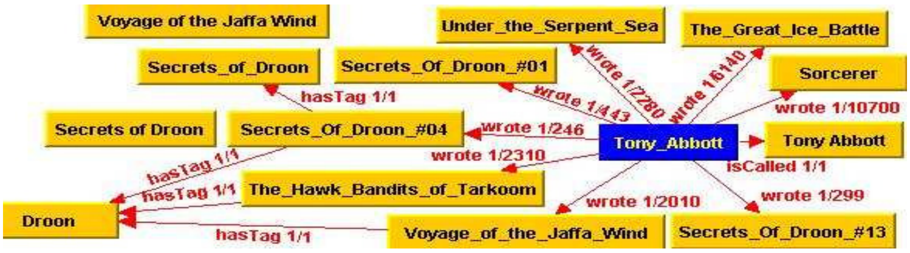


Fig. 5: An example of running the algorithm for precis query "Tony Albott" and $k = 13$ on LT-1 dataset. Weights represent $1/witnessCount$

In over 80% of the cases the summary was assessed as good or acceptable, for $k = 12$ (26% as good and only 19% as poor). In majority of cases the summary was assessed as good or acceptable (figure 6). It is noticeable that the results of the algorithm have better quality for higher value of k, while for the low value of $k = 7$, the majority of cases (67% of cases) was assessed as poor or useless (figure 6).

Concerning the assessment of the facts (edges) selected by the algorithm (figure 7), the algorithm selected "almost all" or "some" interesting facts in 83% of cases, according to evaluators. Missing facts were noticed only in 9% of cases. In 79% of cases, users did not complain about many "irrelevant" selected facts. Again, the assessments are better for the higher value of $k = 12$. See figures 8 and 9 for a detailed comparison.

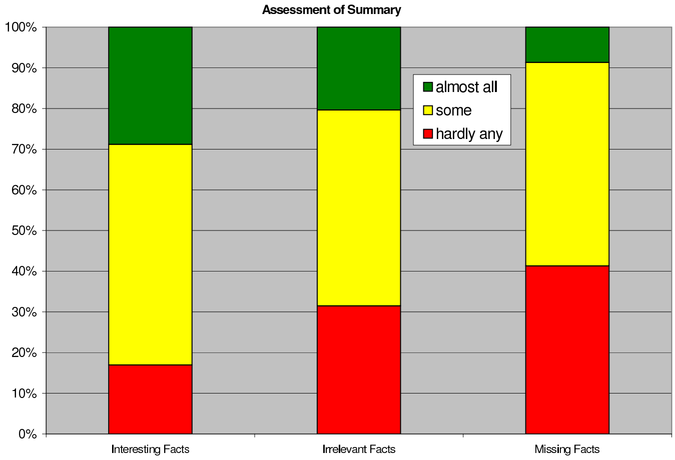


Fig. 7: Assessment of selected facts over all evaluated examples: interesting facts (left), irrelevant facts (middle), missing facts (right)

V. Conclusions and Further Work

To summarise, despite the relative simplicity of the algorithm, the results are quite positive, especially for the higher value of the limit on number of presented edges.

It seems that low assessments for the low value of k is caused by the redundancy of selected facts: "actedIn" in case of actors, or "wrote" in case of writers, for example. This is also confirmed by some textual explanations of the evaluators.

Due to this observation, to improve the summarisation algorithm in future continuation of this work we plan to pay special attention to the *diversification* of the results as preliminarily studied in [7].

Though experimentation on real datasets is still in progress, the preliminary experimental results are promising since the algorithm can reach interesting pieces of information that are a few arcs away from the entity under interest. We plan to continue experimentation, also with different settings, weights and distance computation methods and modifications of the algorithm, e.g. regarding the diversity of the returned summary.

References

[1] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. Open information extraction from the web. *Commun. ACM*, 51(12):68–74, 2008.

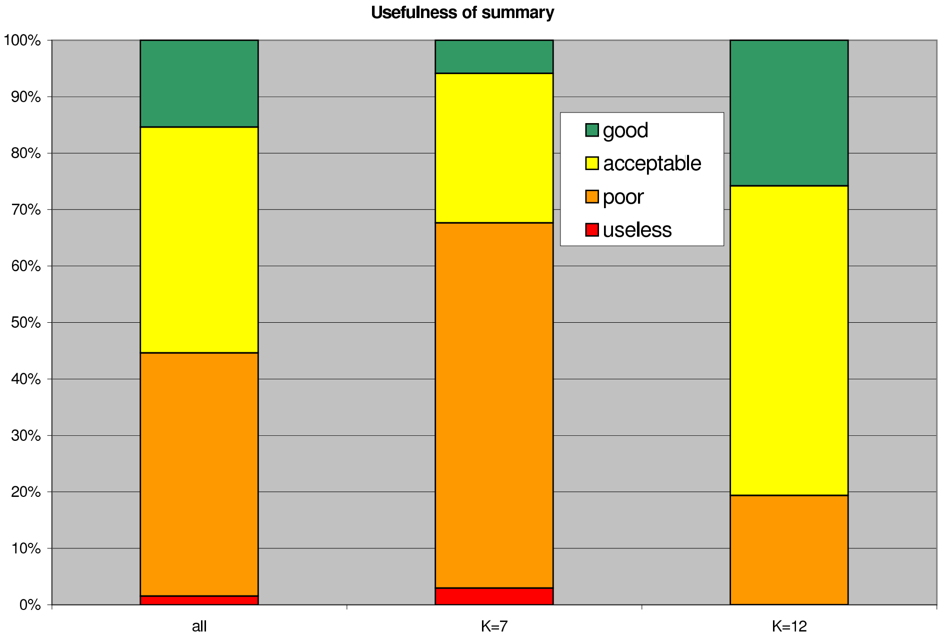


Fig. 6: Usefullness of the results

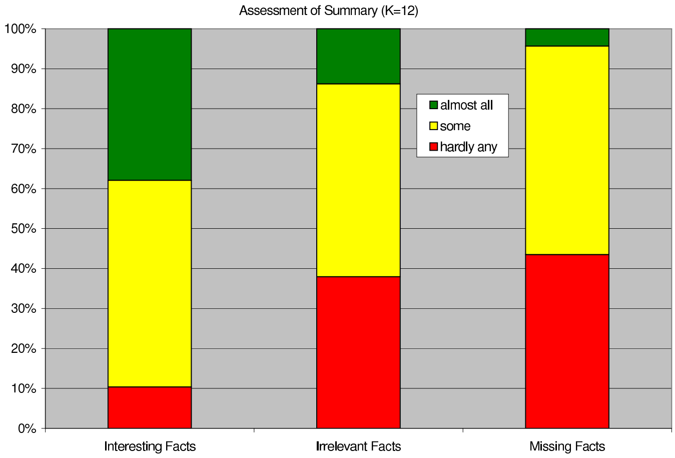


Fig. 8: Assessment of selected facts for limit on edges set to 12: interesting facts (left), irrelevant facts (middle), missing facts (right)

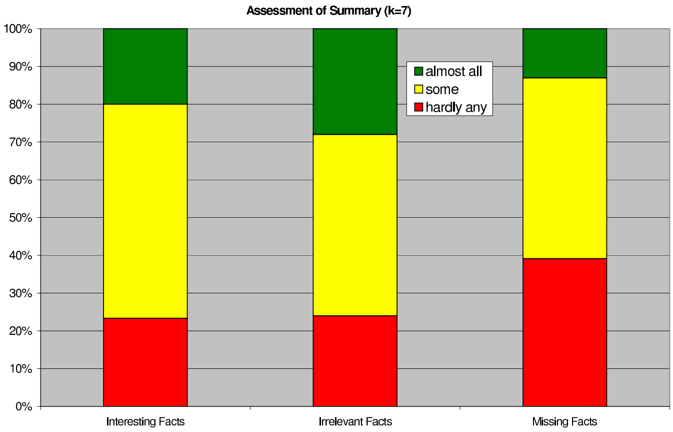


Fig. 9: Assessment of selected facts for limit on edges set to 7: interesting facts (left), irrelevant facts (middle), missing facts (right)

[2] Ahmed Hassan, Anthony Fader, Michael H. Crespin, Kevin M. Quinn, Burt L. Monroe, Michael Colaresi, and Dragomir R. Radev. Tracking the dynamic evolution of participants salience in a discussion. In *COLING*, pages 313–320, 2008.

[3] Yu Huang, Ziyang Liu, and Yi Chen. Query biased snippet generation in xml search. In Jason Tsong-Li Wang, editor, *SIGMOD Conference*, pages 315–326. ACM, 2008.

[4] Georgia Koutrika, Alkis Simitsis, and Yannis Ioannidis. Précis: The essence of a query answer. In *ICDE '06: Proceedings of the 22nd International Conference on Data Engineering*, page 69, Washington, DC, USA, 2006. IEEE Computer Society.

[5] M. Ramanath and K. S. Kumar. A rank-rewrite framework for summarizing xml documents. In *ICDE Workshops*, pages 540–547. IEEE Computer Society, 2008.

[6] Maya Ramanath, Kondreddi Sarath Kumar, and Georgiana Ifrim. Generating concise and readable summaries of xml documents. *CoRR*, abs/0910.2405, 2009.

[7] Marcin Sydow, Mariusz Pikuła, and Ralf Schenkel. DIVERSUM: Towards diversified summarisation of entities in knowledge graphs. In *Proceedings of Data Engineering Workshops (ICDEW) at IEEE 26th ICDE Conference*, pages 221–226. IEEE, 2010.

[8] Xiaojun Wan. Topic analysis for topic-focused multi-document summarization. In *CIKM*, pages 1609–1612, 2009.

[9] Xiaojun Wan and Jianguo Xiao. Exploiting neighborhood knowledge for single document summarization and keyphrase extraction. *ACM Trans. Inf. Syst.*, 28(2), 2010.

[10] Ning Zhang, Yuanyuan Tian, and Jignesh M. Patel. Discovery-driven graph summarization. In *ICDE*, pages 880–891, 2010.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 517–524

ISBN 978-83-60810-27-9
ISSN 1896-7094

Multiple Noun Expression Analysis:

An Implementation of Ontological Semantic Technology

Julia M. Taylor
RiverGlass, Inc &
Purdue University
USA
jtaylor1@purdue.edu

Victor Raskin
Purdue University &
RiverGlass, Inc
USA
vraskin@purdue.edu

Maxim S. Petrenko
Dashkova University &
RiverGlass, Inc
Russia & USA
mpetrenk@gmail.com

Christian F. Hempelmann
RiverGlass, Inc &
Purdue University
USA
kiki@riverglassinc.com

***Abstract*—The paper analyzes multiple noun expressions, or compound nouns, as part of the implementation of Ontological Semantic Technology, which uses a lexicon, an ontology, and a semantic text analyzer to access and represent the meaning of text. Because the analysis and results depend on the lexical senses of words, general principles of lexical acquisition are discussed. The success in interpretation and classification of such expressions is demonstrated on 100 randomly selected sequences of noun compounds.**

***Keywords*—multiple noun expressions, meaning interpretation, ontological semantic technology**

I. Introduction

THIS paper describes the implementation of Ontological Semantic Technology (OST), an offshoot of Ontological Semantics [1], a theoretical and computational approach to meaning in natural language. OST is an advanced, improved, and revised application of Ontological Semantics for real-life commercial systems with its latest implementation currently being developed for RiverGlass, Inc. We will illustrate the process of OST implementation by presenting the elements of its ontology and lexicon, which are activated to analyze the difficult semantics of noun + noun expressions, a specific case that the OST multiple word expression (MWE) module is responsible for handling, and comparing the output of the OST semantic analyzer with an informal taxonomy of the meaning relations between the constituent nouns in these constructions. This approach [2] is not dependent on any training corpus but rather on the semi-automated acquisition of the OST static resources (the lexicons and ontology) which can be used for any text. Based on these resources, all OST processing is fully automated. In order to evaluate whether sentences or expressions are interpreted correctly by the OST software, familiarity with OST ontology is required. While the final evaluation of the approach can only occur on the full implementation of an application, we describe partial metrics both to assess progress and to improve the resources and the software, as necessary.

MWEs are well-known as a notorious problem in NLP [3] and workshops have been dedicated to their analysis since 2003 (http://multiword.sourceforge.net/). The central tasks that MWEs pose are their identification (extraction) as being multiword and not co-occurrences of several single words, and the interpretation and representation of their meaning. Our meaning-based linguistic method focuses on the latter task, which subsumes the former by necessity. Common approaches to the MWE subclass of noun compounds proceed without wanting to use costly-to-acquire world knowledge that would distinguish *fish knife* from *steel knife* with the help of knowing that *fish* are edible and not used as the material of artifacts while *steel* is not edible, but used as material in artifacts. The only knowledge used in such approaches is contextual clues [4], ideally specific paraphrases of the compounds, typically used in supervised learning approaches, e.g., [5]. While the relation between the concepts represented by the nouns in compounds is infinite, these approaches postulate subsets of relations, e.g., [6], [7], into which they aim to classify the compounds, with smaller sets naturally resulting in better performance numbers and the items in the sets almost never being motivated by an application. These consequently very coarsely grained subsets (commonly four to twenty items, cf. [8]) are often mapped onto prepositional paraphrases, in which *fish knife* would become *knife for fish* and *steel knife* would become *knife of steel.* Our approach, on the other hand, allows any property or relation available in our ontology to hold between the concepts and aims at the correct identification of each property with its functors.

II. OST Resources

A. OST Ontology

The OST ontology attempts to capture the users' knowledge of the world in a language-independent way. Text is interpreted in terms of the knowledge in the ontology; thus, it is important that as many relationships among concepts as necessary are accurately captured.

Formally, the ontology is a lattice of logically structured concepts (for a formal representation see [9]). It is divided into EVENTS, OBJECTS, and PROPERTYS, with the first two further divided into PHYSICAL, MENTAL, and SOCIAL subcategories. EVENTS and OBJECTS are connected through PROPERTYS, while strictly adhering to inheritance rules, creating the required richness of interrelationship among concepts. Figure 1 shows the top levels of the OST ontology; Figure 2 shows a sample concept with its (noninherited) properties.

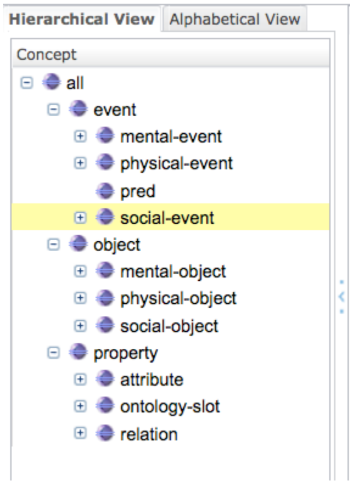


Figure 1. Top levels of OST ontology

B. *OST Lexicon*

The structure and format of lexical entries as well as lexicon acquisition strategies within ontological semantics have been discussed at length in [1: 230-245, 322-350]; acquisition techniques have been outlined in [10]; and the development of a toolkit for automated acquisition within OST has been proposed in [11]. This section will offer a concise outline of the structure and format and briefly discuss procedures of lexicon acquisition, only to the extent that they pertain to the paper.

1) *Definition, functions, and format*

In OST, all language-specific information is encoded in the lexicon. The lexicon is a machine-readable repository of words, non-compositional units (e.g., “hot dog”, “gut feeling”) and idiomatic expressions (e.g., “buy the farm”), whose meaning is defined through ontological concepts and whose morphology and syntax is defined through a set of part-of-speech tags, syntactic role tags and syntactic variables, properly ordered and co-indexed with their semantic counterparts.

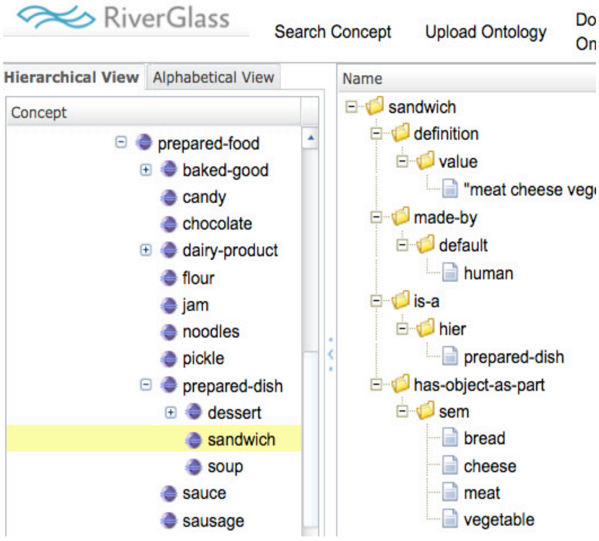


Figure 2. The concept SANDWICH

In relation to the ontology, the content and structure of which is language-independent and parsimonious, the lexicon captures all linguistic idiosyncrasies including language-specific senses, non-literal extensions, synonymy, homonymy, grammatical derivatives, optional phrasal particles, etc. As a resource immediately accessible by the Semantic Text Analyzer (STAn, see Section II C) during processing, the lexicon (1) directs the analyzer to the appropriate ontological concept, its properties, and their fillers, so that further computation can be performed based on that concept’s definition, and (2) follows restrictions stipulated in the ontology so that property fillers in lexicon senses can only narrow down (i.e., be children of) those outlined for respective ontological concepts or introduce new properties or fillers outside the branch of the parent’s filler’s ancestor.

A lexical entry template is shown below:

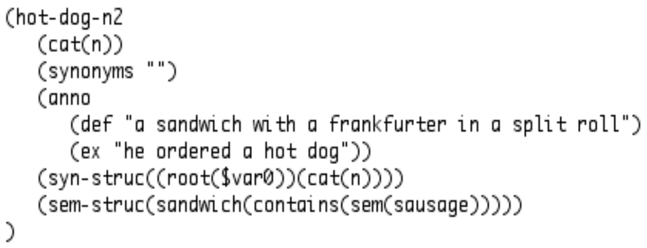
```
(head-entry
   (sense-1, 2, 3…
      (cat(n/v/adj/pro/prep))
      (synonyms "")
      (anno
         (def "")
         (comments "Acquired by <acquirer name> on <date>
            at <time>. ")
         (ex "")
      )
      (syn-struc(
         (root($var0))(cat(n/v/adj/pro/prep))
         (subject/object((root($var#))(cat(np/vp/s))))
         )
      (sem-struc
         (root-concept
            (property(value(^$var#
            (should-be-a(default/sem(concept))))))
      )
   )
)
```

The fields “synonyms”, “anno”, “comments”, and “ex” are of no value to the computer and serve the human acquirer. The fields “cat”, “syn-struc”, and “sem-struc”, meaning “category”, “syntactic structure”, and “semantic structure”, respectively, contain crucial information about the sense used by the analyzer in meaning computation.

The syntactic structure captures the position(s) a word can take in a clause: the root variable locates the word itself, and, for the case of events, syntactic roles of subject or object are indicated, along with their categorical features. For every syntactic role, a variable in the root stands for a specific word carrying this function. Every new syntactic role is assigned a new variable. If word order variation is possible, additional syntactic structures are listed.

The semantic structure anchors the meaning of the sense in ontological concepts, commonly a head concept most defining for the sense. In the case of EVENTs, every syntactic role indicated in the syntactic structure is given its case role interpretation drawn from the ontological definition of the respective EVENT. Variables for each case role are co-indexed with their counterparts in the syntactic structure. Fillers for case roles and other properties tighten more general ontological restrictions where necessary via the SHOULD-BE-A relation, the purpose of which is to further specify the ontological nature of the variable, and the DEFAULT or SEM facets, which prioritize and assign weights to the property values [9].

Figure 3 shows the entry "hot-dog-n2", meaning a sandwich with a sausage rather than the other common sense of "hot dog" which is just the sausage by itself:



```
(hot-dog-n2
   (cat(n))
   (synonyms "")
   (anno
      (def "a sandwich with a frankfurter in a split roll")
      (ex "he ordered a hot dog"))
   (syn-struc((root($var0))(cat(n))))
   (sem-struc(sandwich(contains(sem(sausage)))))
)
```

Figure 3. Lexicon sense of hot-dog

Since no direct concept for a hot dog is available in the ontology, the sense has been acquired by finding the nearest available concept SANDWICH and constraining it with the property CONTAINS filled with the concept SAUSAGE. All ontological restrictions have been observed and further specified: in the ontology, the domain and range of the property CONTAINS are filled with the concept PHYSICAL-OBJECT (of which SANDWICH and SAUSAGE are descendants).

2) *Lexicon acquisition*

A standard acquisition procedure implies creating machine-readable descriptions of lexical senses by listing appropriate concepts and their properties and restricting their fillers where needed. Given that defining the number of lexicon senses, choosing the appropriate head concept, and determining the "tightness" of restrictions for property fillers requires human competence, lexicon acquisition is largely human-driven. However, certain aspects of acquisition (entry format, inherited property fillers, domain/range compatibility check, template-based acquisition of specific word classes, etc.) lend themselves to automation, as demonstrated by various implementations of OST.

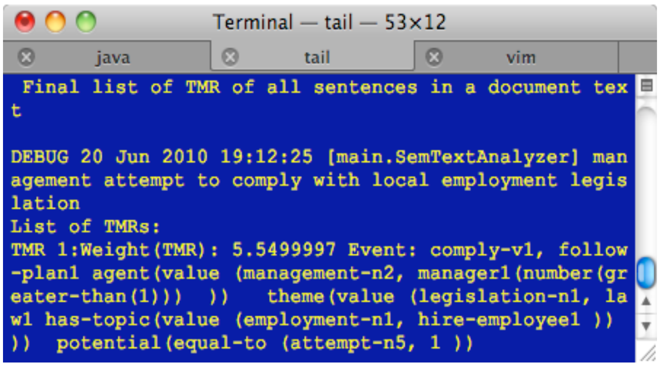
Depending on application objectives, the state of completion of the ontology, and the functionality of the Semantic Text Analyzer, three acquisition techniques have proven useful in various implementations [10]. Ontology-driven acquisition leads to a lexicon with the minimal lexicalization of concepts (i.e. one entry per concept), which is time-efficient but shallow and can only be used at early phases of ontology acquisition. Domain-driven acquisition focuses on topic-related vocabularies and onomasticons, repositories of named entities, as well as specific word classes, and is aided by the analysis of resulting text-meaning representations (TMRs) of domain-specific corpora and "gap detecting" toolkits identifying missing senses with further part-of-speech sorting. Analyzer-driven acquisition, the main method for quality-control and improvement, implies testing every newly acquired entry by the analyzer, followed by TMR analysis and subsequent modifications of the lexicon entry, ontological concepts, or functioning analyzer modules. The approaches are supplementary to each other, although, in practical applications, the domain-driven acquisition tends to dominate in order to ensure optimal lexical coverage and to minimize unattested input.

C. *Semantic Text Analyzer and InfoStore*

The Semantic Text Analyzer (STAn) is a software that interprets text according to knowledge from ontology and lexicon and assigns a TMR to each clause, with textually related clauses as linked TMRs. STAn does not have a graphical interface as its output is written directly into the InfoStore database, which collects all correctly processed TMRs. Thus, InfoStore is in unique possession of all machine-understandable interpretations of all texts processed, which it can then use to guide further interpretations by adding the counterpart of the linguistic and extralinguistic knowledge that humans use to disambiguate, complement meaning, and reasoning contextually.

For the work reported in this paper, we have not yet used InfoStore, but concentrated on STAn's interpretation results obtained only with the current limited ontology (about 2,000 concepts) and English lexicon (about 25,000 senses). The influence of the InfoStore on the overall interpretation process and the treatment of MWEs in particular as well as the resulting ontological adjustments deserve treatment in a separate paper.

Figure 4 shows STAn's interpretation of the sentence *management attempts to comply with local employment legislation* in debug mode. A more readable version of the final TMR is in Figure 5.



```
Final list of TMR of all sentences in a document tex
t

DEBUG 20 Jun 2010 19:12:25 [main.SemTextAnalyzer] man
agement attempt to comply with local employment legis
lation
List of TMRs:
TMR 1:Weight(TMR): 5.5499997 Event: comply-v1, follow
-plan1 agent(value (management-n2, manager1(number(gr
eater-than(1)))  ))   theme(value (legislation-n1, la
w1 has-topic(value (employment-n1, hire-employee1 ))
))  potential(equal-to (attempt-n5, 1 ))
```

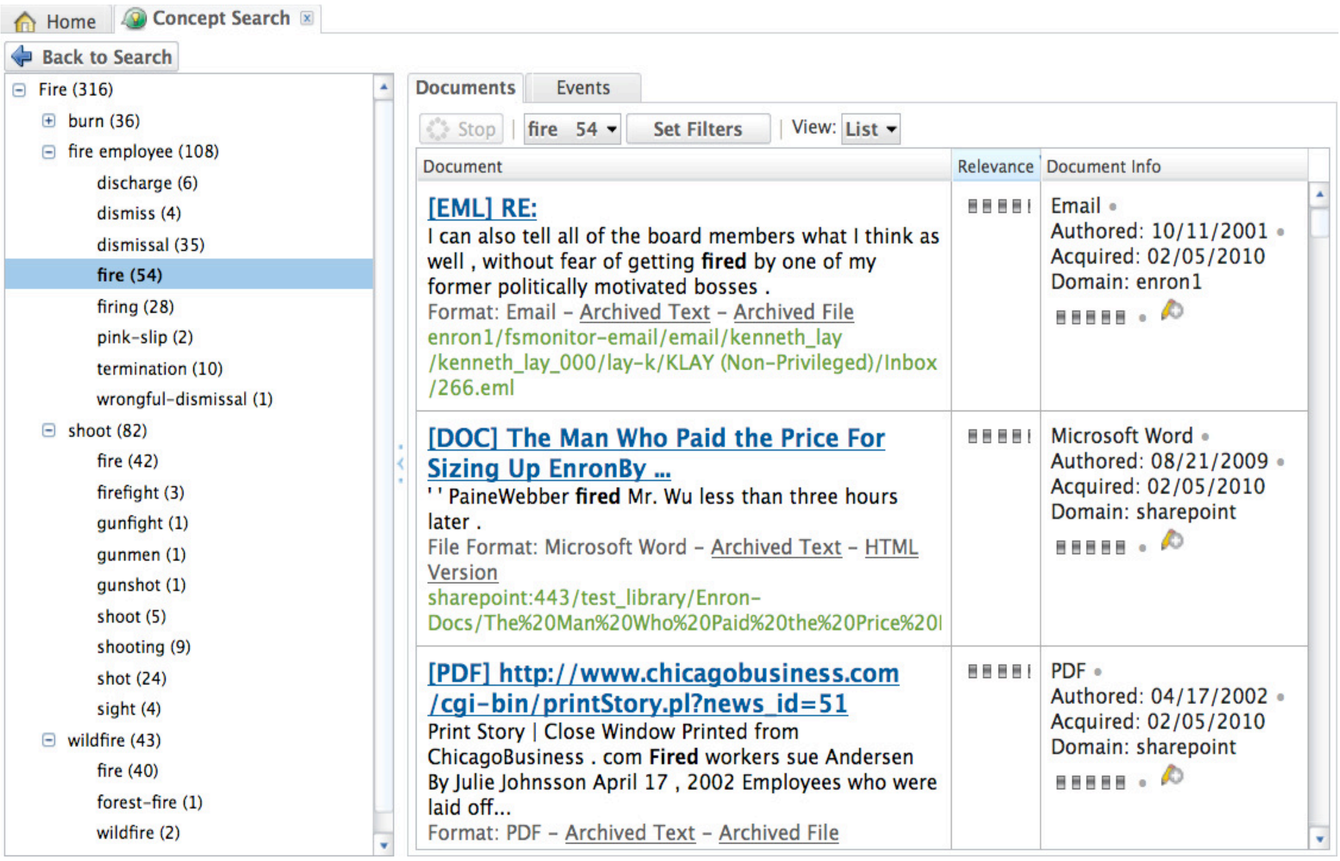
Figure 4. STAn's interpretation of *management attempts to comply with local employment legislation*

```
follow-plan1
   (agent(value(manager1
      (number(greater-than(1)))
   )))
   (theme(value (law1
      (has-topic(value (hire-employee1)))
   )))
   (potential(equal-to (1)))
```

Figure 5. A formatted TMR of *management attempts to comply with local employment legislation*

Figure 6 shows results that are derived from processing done with a simple InfoStore to illustrate the capabilities of an

OST system. It shows a set of documents retrieved from InfoStore that use the concept for firing of employees, listed separately as to which surface word triggered the concept, e.g., *fire*, as opposed to *terminate*, *discharge*, *dismiss*, etc.



Fire (316)
burn (36)
fire employee (108)
discharge (6)
dismiss (4)
dismissal (35)
fire (54)
firing (28)
pink-slip (2)
termination (10)
wrongful-dismissal (1)
shoot (82)
fire (42)
firefight (3)
gunfight (1)
gunmen (1)
gunshot (1)
shoot (5)
shooting (9)
shot (24)
sight (4)
wildfire (43)
fire (40)
forest-fire (1)
wildfire (2)

Document	Relevance	Document Info
[EML] RE: I can also tell all of the board members what I think as well , without fear of getting **fired** by one of my former politically motivated bosses . Format: Email – Archived Text – Archived File enron1/fsmonitor-email/email/kenneth_lay/kenneth_lay_000/lay-k/KLAY (Non-Privileged)/Inbox/266.eml		Email Authored: 10/11/2001 Acquired: 02/05/2010 Domain: enron1
[DOC] The Man Who Paid the Price For Sizing Up EnronBy ... ' ' PaineWebber **fired** Mr. Wu less than three hours later . File Format: Microsoft Word – Archived Text – HTML Version sharepoint:443/test_library/Enron-Docs/The%20Man%20Who%20Paid%20the%20Price%20I		Microsoft Word Authored: 08/21/2009 Acquired: 02/05/2010 Domain: sharepoint
[PDF] http://www.chicagobusiness.com/cgi-bin/printStory.pl?news_id=51 Print Story | Close Window Printed from ChicagoBusiness . com **Fired** workers sue Andersen By Julie Johnsson April 17 , 2002 Employees who were laid off... Format: PDF – Archived Text – Archived File		PDF Authored: 04/17/2002 Acquired: 02/05/2010 Domain: sharepoint

Figure 6. An example of processing with InfoStore

III. Multiple Word Expressions: N+N$^+$

In this section, we apply OST to the semantic analysis of English compound nouns. Such constructions are notoriously difficult to interpret, not only because of the English morphology which allows the adjectival usage of the first noun but mostly because any such construction can be analyzed syntactically as the transform of a clause, where the two nouns can be in any number of meaning relations. In other words, syncretism, prevalent in natural language in general, manifests itself here in the worst possible ways.

For example, the construction *IBM lecture* can mean a lecture about IBM, by IBM, at IBM, or sponsored by IBM ([1] see also [4, 12]), and the human hearers/readers have to use their knowledge of the world to understand the construction correctly—or to ask a clarification question about the intended meaning. Since no syntactic analysis can differentiate among the various possible semantic relations between the two words of a noun + noun compound, it is a good test case for the OST semantic capabilities.

It appears that [13: 589-591] contains a reasonably grain-sized listing of the possible meaning relations, which is reproduced below in an abbreviated form:

- Composition
  - N2 is made from N1
  - N2 consists of N1
- Purpose
  - N2 is for the purpose of N1
  - N2 is used for N1
- Identity
  - N2 has the same referent as N1 but classifies it in terms of different attributes
- Content
  - N2 is about N1
  - N2 deals with N1
- Source
  - N2 is from N1
- Objective type 1
  - N1 is the object of the process described in N2 or of the action performed by the agent described in N2
- Objective type 2
  - N2 is the object of the process described in N1
- Subjective type 1
  - N1 is the subject of the process described in N2
  - N2 is nominalized from an intransitive verb

- Subjective type 2
  - N2 is the subject of the process described in N1
- Time
  - N2 is found at the time given by N1
- Location Type 1
  - N2 is found or takes place at the location given by N1
- Location Type 2
  - N1 is found or takes place at the location given by N2
- Institution
  - N2 identifies an institution for N1
- Partitive
  - N2 identifies parts of N1
- Specialization
  - N1 identifies an area of specialization for the person or occupation given in N2; N2 is animate

It is immediately disclaimed in [13] that some cases may belong to two or more classes and that some cases may belong to none. What is clear is the mixed syntactic/semantic nature of the taxonomy, with the objective- and subjective-types clearly standing out as shallow and, therefore, semantically vague, thus guaranteeing the pretty crippling heterogeneity of the members of these classes.

We have selected the first 100 noun + noun expressions from the Enron corpus (http://www.cs.cmu.edu/~enron/), starting from a randomly selected text, and manually assigned each of them to the most appropriate of the class(es) above. We excluded from the selection the multi-word phrasals that OST treats as single lexicon entries as well as proper nouns. Several of the expressions were of the noun + noun + noun type, and we considered them as a single expression rather than two: the selected 100 expressions contain 6 triple-noun ones. Next, we ran STAn on the actual sentences containing these constructions and compared the OST treatment of them with the above taxonomy.

It is important to understand that OST deals with deep semantics, and therefore recognizes the fact that a shallow semantic—or even such a morphosyntactic category as part of speech—does not determine the ontological basis of a word. Thus, nouns in OST can be anchored in EVENTs (a walk), PROPERTYs (color), or OBJECTs, even though the last type is prevalent by far. The part of speech is used as a clue, often an unimportant one, along with other grammatical (morphological and syntactic) information because the emphasis in OST analysis is always on matches of semantic properties. These clues are nevertheless collected and used as needed in the analysis. As mentioned, noun + noun expressions are particularly interesting and difficult to analyze precisely because the grammatical clues fail to help in differentiating among the numerous types of connection between N1 and N2.

For any N1 + N2 construction, therefore, the Semantic Text Analyzer looks for a property of, typically, N2 such that N1 falls in the range of this property's value. The predominantly prepositive nature of the English adjectival constructions makes the reverse situation—when N2 is the value of the property of N1—rarer. The property, or semantic relation, is (automatically) selected by STAn so that:

$$\max_{p \in P}\{\max_{fct \in Facet}\{I_C(p(fct(I_D)))*\sum_i const*inhd(i)\}\},$$

where P is the property (*p*) set, Facet is the facet (*fct*) set, and I_C and I_D are interpretations of the meaning of N_1 and N_2 with C and D being concepts in $\mathcal{D}$; where $\mathcal{D}$ is the disjoint union of $\mathcal{D}$c (concepts) and $\mathcal{D}$d (literals), and given its interpretation function $\mathcal{I}$, for every atomic concept B, $\mathcal{I}$[B] $\subseteq\mathcal{D}$c; $\mathcal{I}$ [C(Rel(Facet(D)))] $\subseteq\mathcal{D}$c, $\mathcal{I}$ [C(Rel(Facet(D)))] = $\mathcal{I}$ [C] $\cap\mathcal{I}$ [Rel(D)]; and where i is C, D or p and inhd(i) is the inheritance distance measure.

In other words, after STAn selects all the properties of the concept(s) underlying either noun, such that the meaning of the other noun fits the range of those properties or the properties themselves, the main task becomes the selection of the most appropriate connecting property, and this is where the weighting metric described above becomes crucial.

A. *Analysis of 100 N+N(+N) with Biber's taxonomy*

We agree with [13] that some of the N+N expressions fall under several categories: we found 17 such cases. Most of these due to somewhat vague definitions of such rubrics as Content and Identity, but others resulting from the mixed syntactic/semantic nature of the taxonomy, so that *business community planning* could be described equally well as Content or Objective type 1. Also, an expression could be ambiguous, and the alternative meanings could belong to different rubrics and the sentence containing such an expression was not sufficient for disambiguation in isolation (and, as we mentioned above, we are not using InfoStore for this paper).

Table I shows the taxonomic distribution of the 100 expressions; those that belonged to two or more rubrics were counted as many times, thus driving the total above 100. The rubrics not represented in our sample are omitted.

TABLE I. TAXONOMIC DISTRIBUTION OF THE 100 EXPRESSIONS

Taxonomic rubric	**Number of expressions**
Composition	1
Purpose	11
Identity	20
Content	22
Objective type 1	19
Objective type 2	3
Subjective type 1	15
Subjective type 2	3
Time	2
Location type 2	3
Institution	7
Partitive	4
Specialization	6
No appropriate rubric	2

This open-ended and somewhat vague classification is hardly suitable for computation. In the next section we will demonstrate the results of the OST analysis of the same expressions, and compare these results with these in the most populous rubrics above.

B. Analysis of 100 N+N(+N) with OST

As mentioned above, English nouns can be anchored in OBJECTs, EVENTs, or PROPERTYs from the OST ontology. Thus, a more meaningful, although still crude classification is between these types, shown in Table II.

In English adjectives predominantly precede the nouns that they modify. Accordingly, the adjectival use of nouns before other nouns, possible in English because of its impoverished morphology, makes N1s in N1 N2 sequence the modifier of N2 much more frequently than the other way around. This premodification prevails in our sample, with just a sample of postmodification expressions.

TABLE II. TOP LEVEL ONTOLOGICAL DISTRIBUTION OF THE 100 EXPRESSIONS

N1 N2 (N3)	**Number of expressions**
Event Event	11
Event Object	13
Event Property	4
Object Event	19
Object Object	33
Object Property	6
Property Object	5
Property Event	1
Object Event Object	3
Property Event Object	1
Property Object Event	1
Object Object Object	2
Object Event Event	1

STAn relies on premodification-oriented rules and goes into the postmodification regime only after failing to find any possible premodified interpretation. As an example, consider *monitor performance* (taken from our data): while it is likely that a human will interpret it as monitor for performance (postmodification), the premodified interpretation is performance of the monitor, and it is the latter that was easier for STAn to access according to the rules it uses.

Honoring, as it were, the predominance of premodification in the sample and in the language, we will refer to N1+N2 constructions as EVENT-, OBJECT-, or PROPERTY- driven when N2 is an EVENT, OBJECT, or PROPERTY, respectively. The interpretation of the EVENT-driven expressions is clearly delimited by the properties of the events, such as AGENT, INSTRUMENT, THEME, LOCATION, etc, and just as many objects in a regular sentence fit comfortably enough as the fillers of these properties, they do so in the N+N expressions. Unfortunately, this group is minority in the sample (33 instances). The EVENT + EVENT combinations present several additional alternatives for the analysis, and the interpretation is typically more difficult, often requiring InfoStore input.

The PROPERTY-driven compounds are easy to compute as the portion most difficult to find, the property, is explicitly stated in N2. The only work that is required is then to check whether N1 should be in the domain or range of the given property.

The OBJECT-driven compounds, which constituted the majority of the sample (57 instances), are more difficult to interpret correctly than the previous two groups. Problematic among them are OBJECT + OBJECT expressions that require some connection, where this connection might go far beyond a single property, but rather require a “story” involving the two objects. Ellipses (see below) are rampant in this category: thus, *group plan* is actually an insurance plan for the group.

Using the taxonomy in [13], Table III indicates significant OST subclasses in the most populous taxonomic rubrics where the nature and difficulty of computational treatment differ a greatly from one subclass to another.

TABLE III. TOP LEVEL ONTOLOGICAL DISTRIBUTION OF THE 100 EXPRESSIONS

Biber \ OST	Object=N2	Event=N2	Property=N2
Identity	15	3	2
Content	15	6	1
Objective 1	8	11	0
Subjective 1	1	12	2

It is interesting to see how many different connecting properties OST establishes on the OBJECT-driven subclasses, the two most populous ones being the Identity and Content rubrics. The properties listed here were found by STAn and recognized by human expert judges as correct:

- Identity/OBJECT-driven:
  - HAS-SOCIAL-ROLE
  - SALIENCY
  - RANK
  - EMPLOYED-BY
- Content/OBJECT-driven:
  - HAS-TOPIC
  - REPRESENTED-BY

Thus, the OST properties provide a finer grain size. They also can move an expression from one rubric to another, overriding the weak intuition of a human taxonomist or confirming one of the possible choices and excluding the other.

Let us repeat the procedure for the next two most populous EVENT-driven subclasses:

- Objective type 1/EVENT-driven:
  - INSTRUMENT

- EXPERIENCER
- THEME
- THEME-OF
- HAS-TOPIC

- Subjective type 1/EVENT-driven:
  - AGENT
  - LOCATION
  - THEME
  - EXPERIENCER
  - BENEFICIARY
  - PART-OF-EVENT

The 6 subtypes of the last type are particularly noteworthy because they include almost all of the common event properties in OST, INSTRUMENT notoriously but understandably missing.

C. *Evaluation*

1) *Types of evaluation*

There are multiple factors contributing to the process of correct interpretation in OST. As far as noun compounds are concerned, these factors include:

- First and foremost, the correct disambiguation of each noun's meaning.
- The identification of the possible connection(s) between the nouns in the expressions, according to the ontological world model.
- The discovery of the best interpretation of the expressions.
- The assignment of the highest rank (weight) to the TMR with the best interpretation.

2) *The numbers*

For the final version of the paper, we used STAn 4.0 for the evaluation. In the 100 expressions that we used, there were 205 words, most of them with several senses in the OST lexicon. 196 senses of them (or 95.6%) were selected correctly. In the few cases when the words had only one sense each in the lexicon, OST still checked (and the human expert verified) whether the sense fit. So, technically, in 95.6% of the nouns the correct sense was reliably selected.

Next, we evaluated whether STAn selected a connection between the nouns which was possible according to its world model and plausible for human experts, according to their world knowledge. The triple-noun expressions count as one successful selection only when the entire interpretation is possible. We encountered 84 possible interpretations (or 84%) in our sample.

Obviously, only some of these possible interpretations could be considered the best of all possibilities in the given context. However, STAn also has a chance to rank its preference towards sentence interpretation. In other words, the degree of success in interpreting a noun + noun expression may be subordinated to the correct interpretation of the rest of the sentence. We are interested here in whether the best interpretation, according to the human experts, was found; and the number for that is 69 (69%); again, the triple-noun expressions counted as one unit.

Yet a lower number corresponds to the those correct interpretations that were ranked the highest by STAn:

- 46 of them had the highest rank
- 8 of them had the second-highest rank
- 6 of them had the third-highest rank
- 9 more were lower than the third-highest rank

Therefore, given a choice between TMRs, the selection by STAn and the human experts coincided 66.(6)%, or exactly 2/3 of all cases in the sample.

3) *Sources of failure*

There are three major reasons for failure, all of which will be removed at the later stages of OST implementation. First, we ran this sample on a limited lexicon, supported by a limited ontology. Both of these are steadily expanded—see [2] for the cost and time estimates for OST acquisition. The Unattested Input Module was not activated in this experiment, thus lowering STAn's robustness.

Second, we marked 9 conspicuous cases of ellipsis, accounting for 60% of all failures to identify the possible connection, and a closer analysis would probably reveal several more cases. As we mentioned before, STAn couldn't handle *group plan* correctly because it lacked the information that it was actually a group insurance plan. Semantic ellipsis remains an underrecognized and underexplored phenomenon, leading researchers to giving up occasionally on interpretations of such expressions—see, for instance, [14] on the meaning of *fast motorway*. OST can interpret the meanings of such expressions by looking at the domain of SPEED—"fast" is in the range of this property—and the conceptually contingent CAR, which is what is fast on the motorway.

For the reconstruction of ellipsis, OST has to use InfoStore, and the choice not to use information in it for this experiment is the third major reason for the failures, especially when the discovery of the best property and TMR ranking are concerned.

These and other possible problems are going to be corrected in the course of further OST implementation.

IV. Summary

We demonstrated, on a limited sample of 100 N+N$^+$ compounds, the principles of analysis and computational interpretation for compositional multi-word expressions in Ontological Semantic Technology. We compared this OST-based methodology with the traditional taxonomy for the meanings of noun compounds and achieved a finer grain size with the former. While the numbers characterize the experiment as reasonably successful, we also indicated the clear directions of improvement in the course of further OST implementation.

ACKNOWLEDGMENT

The authors are grateful to RiverGlass, Inc., for the permission to use their proprietary resources for examples and illustrations.

REFERENCES

[1] S. Nirenburg and V. Raskin, *Ontological Semantics*. Cambridge, MA: MIT Press, 2004.

[2] V. Raskin, J. M. Taylor, and C. F. Hempelman, "Guessing and knowing: Two approaches to semantics in natural language processing," in *Papers from the Annu. Int. Conf. "Dialogue"*, Moscow, 2010, vol. 9, no. 16, pp. 642-650.

[3] I. A. Sag, T. Baldwin, F. Bond, A. Copestake, and D. Flickinger, "Multiword expressions: A pain in the neck for NLP," in *Proc. 3rd CICLING*, 2002, pp. 1–15.

[4] T. Finin, *The semantic interpretation of compound nominals*. Ph.D. Dissertation, University of Illinois at Urbana-Champaign, 1980.

[5] R. Girju, D. Moldovan, M. Tatu, and D. Antohe, "On the semantics of noun compounds," *Computer Speech and Language*, vol. 19, no. 4, pp. 479–496, 2005.

[6] M. Lauer. *Designing Statistical Language Learners: Experiments on Noun Compounds*. Ph.D. Thesis, Macquarie University, Australia, 1995.

[7] D. Ó Séaghdha, and A Copestake, "Co-occurrence contexts for noun compound interpretation." in *Proc. ACL-2007 Workshop on A Broader Perspective on Multiword Expressions*, Prague, Czech Republic, 2007, pp. 57–64.

[8] M. Lapata, "The disambiguation of nominalizations." *Computational Linguistics*, vol. 28, no. 3, 2002, pp. 357–388.

[9] J. M. Taylor, V. Raskin, "Fuzzy ontology for natural language," 29th *Int. Conf. of the North American Fuzzy Information Processing Society,* Toronto, Canada, July 2010.

[10] M. Petrenko, "Lexicon management in ontological semantics," in *Papers from the Annu. Int. Conf. "Dialogue"*, Moscow, 2010, vol. 9, no. 16, pp. 636-641.

[11] J. M. Taylor, C. F. Hempelmann, and V. Raskin, "On an automatic acquisition toolbox for ontologies and lexicons," In *Proc. ICAI'10*, Las Vegas, USA, July 2010.

[12] P. Isabelle, Another look at nominal compounds," in *Proc. COLING-84*, Stanford, CA, USA, 1984.

[13] D. Biber, S. Johansson, G. Leech, and S. Conrad, *Longman Grammar of Spoken and Written English*. Indianapolis, IN: Pearson ESL, 1999.

[14] N. Asher and A. Lascarides, "Metaphor in discourse," in *Symposium: Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity, and Generativity. Working Notes. AAAI Spring Symposium Series.* J. B. Klavans, B. Boguraev, L. Levin, and J. Pustejovsky, Eds. Stanford, CA: Stanford University, 1995

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 525–530

ISBN 978-83-60810-27-9
ISSN 1896-7094

A web-based translation service at the UOC based on Apertium

Luis Villarejo, Mireia Farrús
Office of Learning Technologies
Universitat Oberta de Catalunya
Av.Tibidabo, 47. 08035. Barcelona (Spain)
Email: {lvillarejo,mfarrusc}@uoc.edu

Sergio Ortiz, Gema Ramírez
Prompsit Language Engineering, S.L.
Avinguda Sant Francesc, 74, 1L.
03195. L'Altet (Spain)
Email: {sergio,gema}@prompsit.com

***Abstract*—In this paper, we describe the adaptation process of Apertium, a free/open-source rule-based machine translation platform which is operating in a number of different real-life contexts, to the linguistic needs of the Universitat Oberta de Catalunya (Open University of Catalonia, UOC), a private e-learning university based in Barcelona where linguistic and cultural diversity is a crucial factor. This paper describes the main features of the Apertium platform and the practical developments required to fully adapt it to UOC's linguistic needs. The settting up of a translation service at UOC based on Apertium shows the growing interest of large institutions with translation needs for open-source solutions in which their investment is oriented toward adding value to the available features to offer the best possible adapted service to their user community.**

I. Introduction

Machine Translation (MT) is one of the classic tasks of Natural Language Processing (NLP) and still an ongoing problem. Since the first attempts, dating from the 1950s [1], the presence of MT in a multitude of assimilation (understanding) and dissemination (publishing) scenarios has increased, as has interest in producing and accessing multilingual content. Companies, public and private institutions, and individual users have looked for solutions to cover their needs and this increasing demand has led, in the last years, to a growing interest of big companies (such as Google[1]) or big public-funded projects (such as EuroMatrix[2]) for the field of MT. The number of MT initiatives has risen greatly in recent years, mainly in statistical MT, as a result of the availability of vast multilingual parallel texts, but also in rule-based MT, example-based MT or hybrid systems. Many of these efforts have been released in the last decade as free/open-source systems (FOSS)[3] making MT available to the whole user community and not just to restricted groups.

This is the case of the Apertium[4] FOSS MT platform presented in this paper. Apertium is a framework in which rule-based machine translation systems can be created. It was first released in 2005 with only two available language pairs while today there are more than 20 stable language pairs available. Apertium has, since then, been chosen by a range of users to meet a number of MT needs. Examples can be seen in a wide variety of scenarios:

- individual users interested in using or developing the Apertium platform,
- less-resourced language communities, interested in increasing language visibility,
- research groups interested in carrying out R&D projects related to MT or NLP,
- companies interested in using MT or in improving the platform for internationalization or to offer commercial services,
- companies from the translation or localization industry interested in increasing productivity,
- public administrations interested in promoting languages and language technologies or,
- academic institutions working in multilingual environments.

From the scenarios above, we can see that Apertium is used in many different real-life contexts[5]. Apertium is, for example, used to publish online bilingual versions of *La Voz de Galicia* newspapers in Spanish and Galician, to generate, via translation, book reviews in many languages on the online Casadellibro.com bookshop or, recently, to produce bilingual versions of the University of Alicante (UA) website in Spanish and Catalan. Beyond Spain, Apertium is being used in companies such as Autodesk, where it is used to produce rough translations from Spanish into Brazilian Portuguese [2] as part of its localization workflow.

One of the most successful industrial applications of Apertium is the online MT web service at the Universitat Oberta de Catalunya (Open University of Catalonia, UOC). A long-term project developed within the UOC in collaboration with Prompsit Language Engineering, as the Apertium service provider, was set up in July 2008. The aim of this project was to improve and adapt Apertium to the UOC's needs for a translation service on its virtual campus. All the adaptations and improvements made by the UOC are free and open-source, and available on the Apertium platform. The UOC has become a regular developer on the Apertium platform, making

[1] http://google.translate.com
[2] http://www.euromatrixplus.net/
[3] See FOSS MT systems at http://computing.dcu.ie/∼mforcada/fosmt.html
[4] http://www.apertium.org
[5] http://www.translationautomation.com/technology/open-mt-ready-for-business.html

contributions and benefiting from the improvements made by the whole Apertium community.

This paper describes Apertium and its application in the context of the UOC. Section II describes the Apertium platform and its main innovative features. Section III sets out the language needs at the UOC, and how Apertium has been integrated in response to these needs. Section IV presents the evaluation and user feedback obtained after integrating Apertium, and finally, section V details the main conclusions and work for the future.

II. Apertium

This section includes a description of the tool: the general architecture, a more specific technical description, and its historical evolution.

A. General Architecture

Apertium is a rule-based MT platform providing the engine, tools and data for a large number of languages under the GNU General Public License[6], and it is being developed by a community of users worldwide. The platform has been described in depth in papers such as [3]. Here is a brief overview of the platform, its history and the main innovative features.

B. Technical Description

The Apertium MT engine is a shallow-transfer system consisting of pipelined independent modules that intercommunicate using text streams (see the architectural diagram in Figure 1). Modules can be used in isolation and other modules can be added to the pipeline. Data and engine are fully decoupled to make the engine language-independent. During the translation, finite-state lexical processing, statistical disambiguation and shallow structural transfer based on finite-state pattern matching takes place. Linguistic data feeding the engine are coded in XML-based files which are compiled into binary format (finite-state letter-transducers [6]) to speed up the translation process (10,000 words can be processed per second on a basic desktop PC). An extensive documentation is also available[7].

The technology behind it is largely based on that of systems already developed by the Transducens group at the University of Alicante (UA), such as the Spanish–Catalan MT system interNOSTRUM[8] [4], and the Spanish–Portuguese translator Traductor Universia[9] [5].

Apertium-based systems consist of three different packages:

- `apertium`: MT engine modules for format management (deformatters and reformatters), part-of-speech tagging (tagger) and transfer tasks (structural transfer).
- `lttoolbox`: toolbox for processing and compiling letter transducers into which data are transformed (morphological analyzers and generators, lexical transfer and post-generators).
- `apertium-l1-l2`: language package containing XML-based data for translating between two languages, l1 and l2, (monolingual and bilingual dictionaries, post-generation dictionaries, data for part-of-speech tagging).

Moreover, the system has the following modules that are connected in a serial way to produce translations of texts in a diversity of formats:

- **Modules for format processing**: they are in charge of separating (without removing) the original format information from text to be translated and restoring the format at the end of the translation process. The **deformatter** and the **reformatter** are the modules that perform this processing.
- **Modules for lexical processing**: they use the information contained in the monolingual and bilingual dictionaries. These are:
  - the **morphological analyzer**, which provides all the possible lexical forms (consisting of lemmas and morphological information) for each word (surface form) in the original text;
  - the **lexical transfer** of the transfer module, which performs the word-by-word (or multiple-word-by-multiple-word) translation of each lexical form delivered by the morphological analyzer and, if needed, disambiguated by the lexical disambiguator;
  - the **morphological generator**, which generates the correct surface form for each lexical form of a word coming from the transfer module;
  - and the **post-generator** that performs some orthographical tasks such as contractions.
- **Lexical disambiguator**: it provides, based on probability estimates, one single lexical interpretation (and the most probable but not always the correct) of an ambiguous word for which the morphological analyzer delivers more than one lexical form.
- **Structural transfer module**: it performs one or three pass transfer operations (depending on the language pair) to apply structural changes between source and target language such as gender, number or case agreement, reorderings, changes in verb tenses (including clitics) or verbal structures, changes in prepositions, generation or deletion of partitives, articles, prepositions or subject pronouns for non-pro-drop languages, etc.

C. Historical Evolution

Since 2005, with the first release of the engine (which was aimed at dealing with closely related languages), tools and the Spanish↔Catalan and Spanish↔Galician pairs, the Apertium platform has been extended greatly:

[6] http://www.gnu.org/copyleft/gpl.html

[7] The Apertium Wiki provides documentation of a wide variety of development and usage scenarios (http://wiki.apertium.org/).

[8] http://www.internostrum.com

[9] http://traductor.universia.net

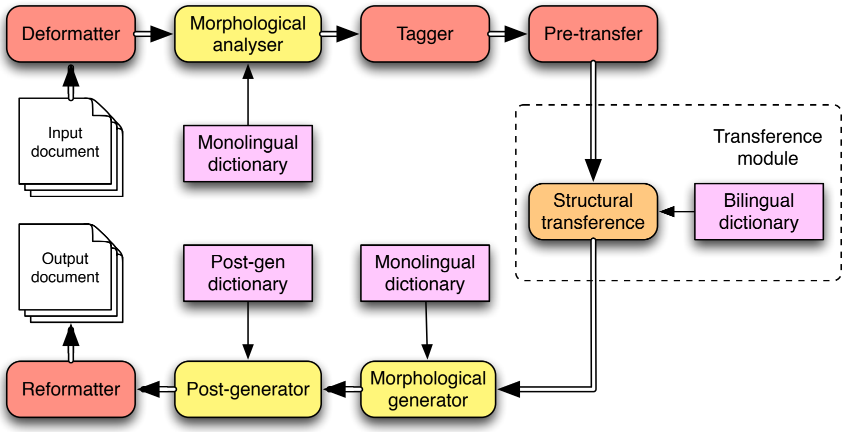


Fig. 1. Modules of the Apertium machine translation system

- more than 22 language pairs have been released[10] and many others have been started or are in development,
- the engine has been improved to deal with less related languages (thanks to the three pass transfer) and to be Unicode compliant
- file format support has been extended to all Office formats, Quark-Xpress, special XML-based formats, etc.,
- support for translation memories has been enabled (still experimental),
- language variants, polysemic or specific domain management has also been enabled,
- applications and tools have been developed for Apertium, such as Tinylex[11], a version of the bilingual dictionaries for mobile devices; `apertium-subtitles`, a tool for translating subtitles; user interfaces or add-ons for Firefox, and
- research on social MT development based on Apertium is currently being developed as part of a project entitled Tradubi [7].

All these efforts result in the continuous improvement of the Apertium platform and are carried out by a worldwide community of developers and committers acting individually or as groups in companies, organizations, research groups, etc.

We found that Apertium is especially suitable for its integration inside universities like UOC for various advantages:

- **Open source**: Apertium is licensed under the GPL (GNU General Public License). This implies that the source code is provided with the application, and this allows UOC to adapt both the MT engine and the linguistic data to its specific needs.
- **Free software**: GPL requires all derivative software to be also licensed under GPL; this promotes the availability of all new source code developed for Apertium by the user community. Therefore, anyone using the system automatically benefits from new developments made by third parties, both on the engine and the data.
- **Predictability**: Given that Apertium a rule-based MT system, the obtained results when translating documents are highly predictable. We believe that this is an advantage over other non-rule-based MT technologies for several reasons. Firstly, many of the systematic mistakes made by the MT system can be corrected in a systematical way. Secondly, human post-editors, once accustomed to the system behavior, are able to reduce the amount of work checking the original when post-editing a document. This reduction, which can even be automatized, makes their work simpler and more productive.

Some of the improvements described above, as well as linguistic improvements in existing language pairs, were made by Prompsit as part of the development project designed by the UOC as described in the following section.

III. Practical developments for adaptation to the UOC

The UOC is a private Catalan online university whose mission is to provide people with lifelong learning and education. The UOC community is made up of more than 54,000 students, over 2,000 teaching counsellors and faculty working alongside an administrative staff of around 500 people. More than 1,475,000 documents have been downloaded from its Virtual Campus[12], including articles, studies and teaching materials. Given these figures and the fact that the university mainly works with three languages: Catalan, Spanish and English, there is the vital need to make intensive use of language technologies. Thus, the Office of Learning Technologies[13] and the Language Service[14] at the UOC have been developing several language tools [8] in order to exploit and reuse the

[10] http://wiki.apertium.org
[11] http:www.tinylex.org
[12] Data from academic year 2007-2008
[13] http://learningtechnologies.uoc.edu/
[14] http://www.uoc.edu/serveilinguistic/

language data generated by the University. In this context, several machine translation tools were evaluated. Apertium, thanks to its modular architecture, fully customizable nature and coverage of the languages used at the UOC, was adopted and improved on to meet the University's needs. Once Apertium was selected, it was integrated with the aforementioned language tools in order to create a document flow [9] for semi-automation of language processes. Below is a description of the translation service functionalities available and the user interface developed as part of the integration process.

A. Features Installed and Improvements Developed

Apertium has been installed on the UOC's Virtual Campus and is available both for faculty and administrative staff. The translation service has been set up for Catalan, which is the language used most frequently at the University, and the supported language pairs are Catalan↔Spanish, Catalan↔English and Catalan↔French. The functionalities implemented include:

- text translation
- document translation
- translation as you browse
- advanced HTML treatment (with structure validation)
- compressed files
- TMX utilization creation

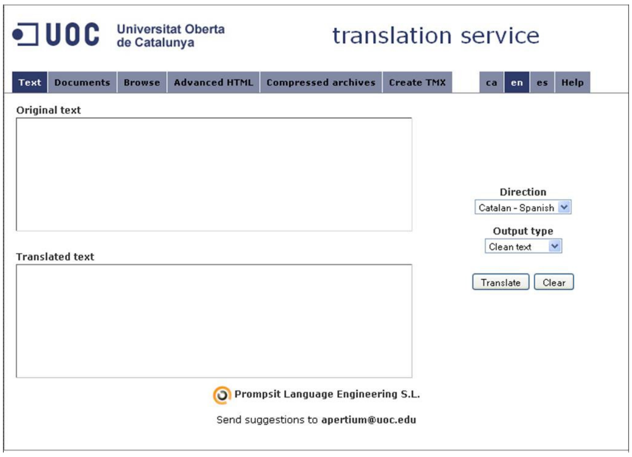
The integration in the Virtual Campus involved a series of specific actions in order to meet the UOC's needs. An upgrade was performed on the dictionaries using specific vocabulary extracted from the UOC's website. Regarding formats, specific support for Microsoft Excel and PowerPoint was developed. The marking of unknown and ambiguous words was adapted to ease post-editing.

Moreover, support for compressed files was developed so that users could send rar, zip, 7z, tar.bz or tar.gz files and receive an equivalent file with its contents translated.

In terms of the translation workflow, the use of translation memories prior to the machine translation phase was introduced in order to reuse the information generated within the university. This is especially useful in an institution where some departments generate similar documents, in which only small parts of them are modified, and many of the sentences translated could be reused in future translations. Moreover, since professors need to generate documentation in very specific domains, translation memories can help ensure more accurate and personalized translations.

B. User Interface

A special user interface was developed to integrate all the functionalities adopted or developed to create the translation service. This interface is available in Catalan, Spanish and English. Its different functionalities are structured by means of browser-like tabs as shown in Figure 2. Help is provided by means of HTML pages, and a downloadable PDF file, with detailed information on each functionality.



Direction: Catalan - Spanish

Output type: Clean text

Fig. 2. Apertium user interface

TABLE I
ERRORS OBSERVED WHEN TESTING THE FORMAT QUALITY OF ELEVEN DOCUMENTS.

Part of document	DOC	PPT	XLS
titles and subtitles	1	0	0
paragraph structure	0	1	0
bold and italics	0	1	1
figures	0	1	0
tables	0	1	1
headings and footnotes	3	0	0
lists	0	0	0
apostrophes	3	0	1
content (missing parts)	0	0	0
strange characters	2	0	0

IV. EVALUATIONS

Before releasing the first version of the translation system in December 2009, a series of tests were carried out in order to check its effectiveness in handling formats and the user interface's usability. To do so, we followed some of the ideas introduced in [10] and [11]. This section briefly outlines the design and the results obtained in both tests, performed by nine people from the university staff who volunteered to evaluate system quality for different formats and the usability of the interface.

After the release, the developers compiled user feedback from the suggestions and questions submitted to an address given for this purpose. The last part of this section summarizes the comments, frequently asked questions and suggestions made by users in order to improve interface usability and format handling.

A. Format quality test

The aim of the format quality test was to evaluate whether the use of different file formats led to problems in the translations or not, focusing on DOC, PPT and XLS formats. Table I shows the errors encountered by the nine evaluators in different parts of documents for each file format: titles and subtitles, figures, missing parts or strange characters in the translation, etc. A total of eleven documents were tested by the evaluators.

It can be seen from the table that most of the errors encountered were related to Microsoft Word documents and headings, footnotes and the appearance of strange characters.

Nonetheless, when enquiring about system performance when translating compressed files (zip), no problems were encountered: the format and folder structure were maintained in the output file, and all the files included in the compressed folder were completely translated.

This test allowed us to identify and solve minor problems when dealing with formats. This was true in particular when translating Microsoft Office documents, due to their closed internal structure.

B. *Usability test*

The goal of the usability test was to evaluate the general level of user satisfaction regarding a number of aspects. We mainly wanted to detect possible inconveniences in the information layout and the user's ability to perform the intended tasks. Table II shows the results obtained where *NA* stands for *not answered*, *0* means *disagree entirely*, *1* means *disagree*, *2* means *agree*, and *3* means *agree entirely*.

TABLE II
DEGREE OF SATISFACTION IN THE INTERFACE USABILITY TEST

Concept evaluated	**NA**	**0**	**1**	**2**	**3**
easy-to-use interface	2	0	0	3	4
information easy to find	2	0	1	2	4
information clearly organized	2	0	1	2	4
submitted tasks completed	3	0	0	2	4
understandable help	3	0	1	3	2
help easy to find	3	1	0	2	3
easy to access interface	2	0	0	3	4

As can be seen from the table, generally speaking, the interface's usability was rated highly.

C. *User feedback*

Once the translation system was opened to the UOC community, users had the opportunity to send their questions and suggestions. More than 400 users visited the web every month, and among the most frequent errors made were forgetting to select the format type via a drop-down menu. This menu left room for errors when selecting the appropriate format and this situation could have been avoided from the start by incorporating the automatic format detector that it is used when translating compressed files. Another of the most frequent errors was to try to translate a type of document that differed from the default extension, or cutting and pasting XML files into Word documents instead of translating them in XML format. This error stems from use of the previous machine translator used by the UOC community, where XML documents were translated in this way.

Among the suggestions received, the most frequent were to include the translation of PDF documents or automatic selection of the file format, whereby the user does not need to be aware of the type of file they wish to translate.

The feedback we obtained led to a series of actions. First of all, we compiled the most frequently asked questions in a document, providing answers to these questions. This document was made available through the user interface so it could be easily accessed by users.

D. *Linguistic quality*

An evaluation of the linguistic quality of the three available language pairs was carried out after the integration of the Apertium system at the UOC. These three pairs are at different levels of development:

- Spanish↔Catalan is the most actively developed pair inside the Apertium platform, having, for example, more than 41,000 bilingual correspondences in its dictionaries.
- English↔Catalan has also been part of various development projects. It has around 34,000 bilingual correspondences.
- French↔Catalan is the least improved pair, still considered a prototype, with around 12,000 bilingual correspondences.

The evaluation was done using the tool *apertium-eval-translation*, which compares the marked Apertium output of a text (unknown words are marked with a star before the word) and a post-edited version of the same text to calculate some evaluation variables based on edit-distance techniques. During our evaluation, two variables were assessed: *Coverage*, i.e. the percentage of words for which Apertium returned at least one translation, and *Word Error Rate* (WER), i.e. the percentage of words being post-edited to convert the MT output into the post-edited file, as shown in Table III. These results are the ones obtained after improving for Spanish↔Catalan as part of the development project carried out by the UOC. Catalan↔English is also being improved as part of two projects led by the Linguamon-UOC Chair in Multingualism. Catalan↔French is almost at the same level of development as it was before starting the project.

TABLE III
SAMPLE APERTIUM MT SYSTEM OUTPUT QUALITY.

Language pair	**Coverage**	**WER**
Spanish-Catalan	97.1%	4.3%
Catalan-English	94.1%	29.3%
Catalan-French	92.4%	21.9%

V. CONCLUSIONS AND FUTURE WORK

We have presented the details of the integration of a web-based translation service based on Apertium at the UOC. This integration was conducted together with a series of evaluations that lead us to make minor changes in the overall project in order to better meet UOC users' needs. The translation service has been running since December 2009 and the number of unique visitors in the first five months adds up to more than 600. Given that figure and the positive feedback obtained from the users via e-mail and the different evaluations made, we can say that the translation service has had a general good acceptance. Once we covered an internal testing period, and as a conseqence of the general good acceptance of the system, we decided to provide the translation service for the general public and it is now publicly available at http://apertium.uoc.edu.

In terms of future work, and in order to obtain more effective translations, we plan to include semantic domains that will allow for the disambiguation of polysemic and homonymous words. The semantic domains will be classified according to the subjects linked to studies at the UOC. For instance, the word *table* that can be translated as *tabla* (table of results) or *mesa* (dining table), depending on whether the word is related to mathematics or general vocabulary, would be clearly disambiguated through the use of semantic domains.

In addition to that, and inside the Google Summer of Code program, we are currently developing an online post-editing tool which will provide the user with a smooth integration of spell checker, grammar checker and dictionaries together with the Apertium platform.

Another direction for future work is improving the accessibility of the interface. In order to achieve this goal experts in accessibility are currently running a series of tests on the web service to be able to identify potential problems from this point of view.

In more general terms, user feedback will guide future work which will take into consideration improvements to the Apertium platform regarding new functionalities or increasing linguistic quality.

ACKNOWLEDGMENT

The authors would like to thank the Linguistic Service technician's fruitful suggestions which helped us to improve the overall system not only in terms of the linguistic information but also in terms of the user experience.

REFERENCES

[1] W. J. Hutchins and H. L. Somers, *An Introduction to Machine Translation*. Academic Press, London, UK, 1992.

[2] F. Masselot and P. Ribiczey and G. Ramírez-Sánchez, *Using the Apertium Spanish-Brazilian Portuguese machine translation system for localization*. Proceedings of the EAMT Conference, Sain-Raphaël, France, 2010.

[3] M. L. Forcada and F. M. Tyers and G. Ramírez-Sánchez, *The free/open-source machine translation platform Apertium: Five years on*. Proceedings of the First International Workshop on Free/Open-Source Rule-Based Machine Translation FreeRBMT'09, Alacant, Spain, 2009.

[4] R. Canals-Marote and A. Esteve-Guillén and A. Garrido-Alenda and M. Guardiola-Savall and A. Iturraspe-Bellver and S. Montserrat-Buendia and S. Ortiz-Rojas and H. Pastor-Pina and P. M. Pérez-Antón and M. L. Forcada, *The Spanish-Catalan machine translation system interNOSTRUM*. Proceedings of MT Summit VIII: Machine Translation in the Information Age, Santiago de Compostela, Spain, 2001.

[5] A. Garrido-Alenda and P. Gilabert-Zarco and J. A. Pérez-Ortiz and A. Pertusa-Ibáñez and G. Ramírez-Sánchez and F. Sánchez-Martínez and M. A. Scalco and M. L. Forcada, *Shallow parsing for Portuguese-Spanish machine translation*. In Branco, A., A. Mendes, and R. Ribeiro, eds., *Language technology for Portuguese: shallow processing tools and resources*, pages 135–144. Edições Colibri, Lisboa, 2004.

[6] A. Garrido-Alenda and M. L. Forcada and R. C. Carrasco, *Incremental construction and maintenance of morphological analysers based on augmented letter transducers*. Proceedings of the TMI, Keihanna/Kyoto, Japan, 2002.

[7] V. M. Sánchez-Cartagena and J. A. Pérez-Ortiz, *Tradubi: open-source social translation for the Apertium machine translation platform*. The Prague Bulletin of Mathematical Linguistics, 2010.

[8] L. Villarejo and J. Moré and M. Vázquez., *Proyecto RESTAD - Herramientas de código libre para la traducción y postedición de documentos*. In Proceedings of the FLOSS (Free/Libre/Open Source Systems) International Conference, 2007.

[9] L. Villarejo and D. Cullen and A. Corral, *La integració de les tecnologies de la llengua en el flux de treball del Servei Lingüístic de la UOC*. Llengua i ús, revista tècnica de política lingüística 46, 2009.

[10] G. Letnikova, *Developing a Standardized List of Questions for the Usability Testing of an Academic Library Web Site*. Journal of Web Librarianship, vol. 2(2–3), pages 381–415, 2008.

[11] P. Gore and H. G. Sandra., *Planning Your Way to a More Usable Web Site*. Online, vol. 27(3), pages 20–27, 2003.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Tools and Methodologies for Annotating Syntax and Named Entities in the National Corpus of Polish

Jakub Waszczuk[*†], Katarzyna Głowińska[*], Agata Savary[◇*], Adam Przepiórkowski[*†]
[*]Institute of Computer Science, Polish Academy of Sciences, ul. Ordona 21, 01-237 Warsaw, Poland
[◇]Université François Rabelais Tours, Laboratoire d'Informatique, 3 pl. Jean-Jaurès, 41000 France
[†]University of Warsaw, ul. Banacha 2, 02-097 Warsaw, Poland
jw235843@students.mimuw.edu.pl, k.glowinska@gmail.com, agata.savary@univ-tours.fr, adamp@ipipan.waw.pl

***Abstract*—The on-going project aiming at the creation of the National Corpus of Polish assumes several levels of linguistic annotation. We present the technical environment and methodological background developed for the three upper annotation levels: the level of syntactic words and groups, and the level of named entities. We show how knowledge-based platforms Spejd and Sprout are used for the automatic pre-annotation of the corpus, and we discuss some particular problems faced during the elaboration of the syntactic grammar, which contains over 800 rules and is one of the largest chunking grammars for Polish. We also show how the tree editor TrEd has been customized for manual post-editing of annotations, and for further revision of discrepancies. Our XML format converters and customized archiving repository ensure the automatic data flow and efficient corpus file management. We believe that this environment or substantial parts of it can be reused in or adapted for other corpus annotation tasks.**

I. Introduction

The National Corpus of Polish (Pol. *Narodowy Korpus Języka Polskiego*; NKJP; http://nkjp.pl/) is a 3-year project (2007-2010), involving a consortium of four partners coordinated by the Institute of Computer Science, Polish Academy of Sciences ([1], [2]). The aim of the project is to create a 1-billion (10^9) word corpus of Polish annotated at various levels, with a 300-million balanced subcorpus and a number of annotation tools. The following linguistic annotation layers are distinguished: segmentation (word-level and sentence-level), morphosyntax, word sense disambiguation (limited to around 100 lexemes), syntactic words, syntactic groups and named entities.

A 1-million word balanced subcorpus undergoes manual annotation at all these layers and it serves as a training corpus for various annotation tools (cf., e.g., [3]). The current paper gives an overview of methodologies and tools used for the semi-manual annotation of the 1-million corpus at the last three – broadly syntactic – layers.

The layer of syntactic words (SWs) builds on top of the morphosyntactic layer. Fine-grained word-level segments are grouped into more traditional words, including reflexive verbs (consisting of two segments: the verb and the reflexive marker), analytical tense and mood forms, etc. [4] Syntactic groups (SGs) are constructed on top of SWs, and include nominal groups, prepositional groups, clause-level groups, etc., but no attempt is made to solve attachment ambiguities. Finally, named entities (NEs), i.e. proper names of persons, geographical objects and organisation, as well as temporal expressions, refer again to the layer of morphosyntactically annotated segments.

II. Related Work

Some of the first treebanks were constructed fully manually, by drawing trees for particular sentences; this is the case, for example, for the Penn Treebank (PTB) of English [5], the German Negra/Tiger Treebank [6] and the Prague Dependency Treebank [7]. Some treebanks were created by converting existing treebanks to the new linguistic theory; for example, parts of roughly Chomskyan PTB were converted to Head-driven Phrase Structure Grammar, Lexical Functional Grammar and Constraint Categorial Grammar. However, the usual way of developing new treebanks consists in the automatic parsing of texts and the manual selection of the right parse. For example, [8] reports that the ERG grammar [9] covers around 80% of the Wall Street Journal part of PTB, with sentences not adequately covered by the grammar serving as the basis for further grammar development.

The outcome of the effort reported here will not constitute a typical treebank, as the annotation in NKJP stops at the partial (or shallow) syntactic markup (cf., e.g., [10], [11], as well as [4]), where structural ambiguity is not an issue.[1] Hence, the approach mentioned above, focussing on disambiguation, is not directly applicable to the task at hand, but the general semi-manual iterative methodology is similar: parse sentences using a manually constructed grammar, ask annotators to correct the results of parsing by hand, and use error and emission reports for the improvement of the grammar, before applying it to the next batch of sentences.

There are many multi-layer corpora developed by now, typically containig morphosyntactic, (deep) syntactic and some semantic and/or discourse representation. For example, the Prague Dependency Treebank mentioned above has these three layers (called morphological, analytical and tectogrammatical), currently further extended with coreference [13] and high-level

This research was funded in 2007-2010 by a research and development grant R17-003-03 from the Polish Ministry of Science and Higher Education.

[1] In fact, there is a separate project carried out at the same institute, aiming at the construction of a full constituency treebank on the basis of the same 1-million word subcorpus; cf. [12].

inter-clausal structure [14]. The current project adopts a more fine-grained and conservative approach, with three layers between morphosyntax and deep syntax proper: possibly multi-segment SWs, NEs and possibly partial SGs. We claim that this gradual procedure makes it possible to better control the quality of the linguistic annotation.

Also at the level of NEs, the annotation strategies adopted here are rather fine-grained, namely, not only the longest-match occurrences of NEs are annotated, but also all recursively embedded ones, and, moreover, overlappingly coordinated NEs are appropriately marked (cf. [15], [16]).

Let us finally note that, while only partial syntactic structures are annotated here, syntactic groups contain the kind of information not usually found in treebanks, namely, they mark both syntactic and semantic heads. For example, in case of prepositional groups, the preposition serves as the syntactic head[2], but the semantic head is the most meaningful word within the argument of this preposition. Arguments for the usefulness of this kind of annotation, and further details, may be found, e.g., in [17], [4].

III. Annotation Data Flow

The three syntactic annotation layers in the NKJP are organised into two parallel data flows: one for syntactic words and syntactic groups (henceforth, *syntactic annotation* in the narrower sense), and the other for named entities.

The main differences between the data flows show up during the pre-processing step – different tools, with different input specifications, are used for automatic pre-annotation. In case of the syntactic annotation (Fig. 1) a shallow parsing system called Spejd is used to extract SWs and SGs from the underlying morphosyntax level (cf. section IV-A). For the similar (from the data flow point of view) task of NE recognition another platform, Sprout, is used (see section IV-B).

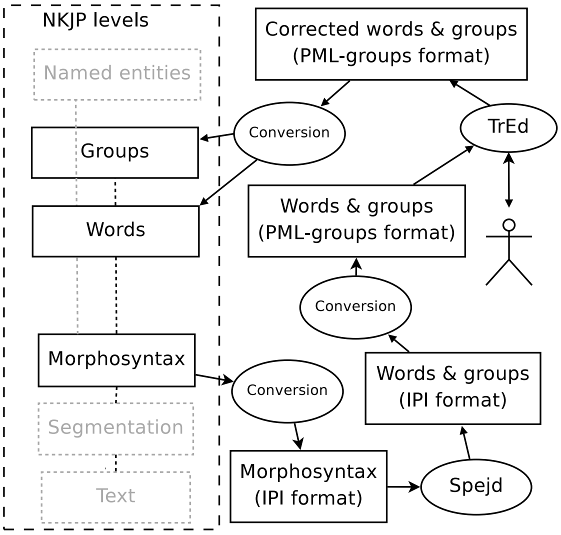

NKJP levels
Named entities
Groups
Words
Morphosyntax
Segmentation
Text
Corrected words & groups (PML-groups format)
Conversion
TrEd
Words & groups (PML-groups format)
Conversion
Words & groups (IPI format)
Conversion
Morphosyntax (IPI format)
Spejd

Fig. 1. Data flow in the syntactic annotation task of the NKJP corpus

Spejd takes structured text with segmentation and morphosyntax information as input. It requires a specific input format (called *IPI format* in Fig. 1) that can be automatically obtained from the NKJP morphosyntax level. Conversely, Sprout requires pure text as input, which complicates the whole process of data conversion (see Fig. 2). A raw text taken from the corpus repository is processed by lexical resources and grammar rules. The NEs identified in the process, together with their embedded structures, are marked in an XML Sprout-specific output. Since Sprout outputs the cardinal numbers of the beginning and ending characters of each recognized sequence, the converter consults the segmentation level of the text in order to translate text ranges into token identifiers. Moreover, for each token, its morphological tag and lemma are recopied from the morphosyntactic annotation of the text.

After the pre-processing step, files have to be prepared for manual annotation. In both data flows annotators use a tree editor TrEd (see section V) to examine and correct results of automatic annotation. Again, files have to be translated to TrEd-readable formats (PML-groups for syntactic annotation and PML-NE for NEs), which were defined using the Prague Markup Language (PML, http://ufal.mff.cuni.cz/jazz/PML/doc/). Due to the sampling methodology adopted in NKJP, files contained in the 1-million gold standard subcorpus are of a very variable length (from several to several thousand sentences). They do not correspond to complete texts taken from the 1-billion word corpus, but to randomly chosen paragraphs thereof. For the sake of ergonomy, it is important to present the annotator with text portions of a uniform length, thus easily manageable. Therefore, the converter divides each text which is too large into files of a limited number of sentences corresponding to roughly 1 hour of human annotation effort. Text splitting is designed so as to keep together all sentences appearing in one paragraph. Conversely, too small files (of one or several sentences), are organised into file lists and annotated as bigger units.

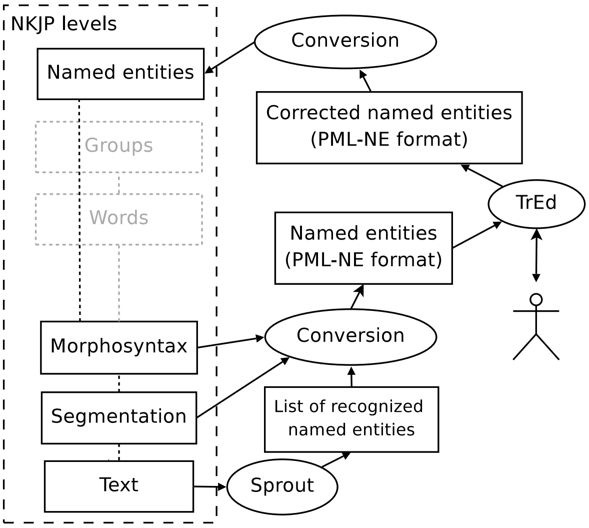

NKJP levels
Named entities
Groups
Words
Morphosyntax
Segmentation
Text
Conversion
Corrected named entities (PML-NE format)
TrEd
Named entities (PML-NE format)
Conversion
List of recognized named entities
Sprout

Fig. 2. Data flow in the NEs annotation task of the NKJP corpus

[2] because it governs the case inflection of its arguments

Finally, PML files are transferred to *corpus files management system* (see section V-C) which is responsible for distributing files between annotators and for storing results of consecutive annotation steps.

Two annotators work on each corpus fragment. An adjudicator reviews any cases of disagreement and chooses the correct annotation. Each annotator and adjudicator works offline with TrEd installed locally, connecting to the subversion repository only to send results of his work, or to download new files. Two TrEd extensions, NKJP_groups and NKJP_names, have been developed to support annotation of PML-groups and PML-NE files. An annotator can download (or upgrade) extensions from within the TrEd application, with no need to run separate installation process. Despite no particular computing background of the annotators, they successfully install and operate the whole annotating platform.

The last stage consists in converting the PML formats of the validated annotations into the final NKJP formats. Here, the subfiles have to be merged into files corresponding to the initial texts and embedded XML elements (NEs and SGs) get transformed into pointers (for stand-off annotation).

IV. Automatic Annotation

A. Shallow Parsing with Spejd

Syntactic annotation in the National Corpus of Polish consists in joining words together into constituents: first at the level of SWs, then at the level of, possibly embedded, SGs. At the former, fine-grained word-level tokens are replaced by coarse-grained SWs (e.g., analytical tense and mood forms, analytical degree forms, reflexive verbs, discontinuous conjunctions, etc.). The tagset at this level differs somewhat from the tagset of word-level segments in order to allow for broader grammatical classes and more traditional grammatical categories, such as tense, mood and reflexivity. The complete tagset for SWs is presented in [18].

At the SG level, each identified group is annotated with pointers to its syntactic head (SynHead) and semantic head (SemHead). Only those groups that can be recognized with very high accuracy are marked, so that the shallow grammar resulting from the manual correction process can be reliably applied to the whole 1-billion word corpus. For example, a nominal phrase that consists of a noun and a prepositional phrase, e.g., *dom z ogrodem* 'a house with a garden', is always treated as two SGs (*dom* and *z ogrodem*), without an attempt to solve PP-attachment ambiguities. We make an exception for compound prepositions that consist of two prepositions and an interposing noun (e.g., *w odniesieniu do* 'with reference to'), as well as for elective constructions (e.g., *jeden z najlepszych* 'one of the best').

The following SGs are distinguished in NKJP:

- nominal group (NG): *malarz kwiatów* 'a painter of flowers', *nic specjalnego* 'nothing special',
- numeral group (NumG): *stu z nas* 'a hundred of us',
- adjectival group (AdjG): *zbyt długi* 'too long',
- prepositional-nominal group (PrepNG): *za murami miasta* 'beyond city walls',
- prepositional-adjectival group (PrepAdjG): *[wyglądasz] na zmęczonego* '[you look] tired',
- prepositional-numeral group (PrepNumG): *[wakacje] dla dwojga* '[a holiday] for two',
- adverbial group (AdvG): *ładnie*[3] 'nicely',
- discourse group (DisG): *no cóż* 'oh well', *itd.* 'etc.',
- subordinate clause (CG) (with subordinate conjunction): *[powiedział], że nie przyjdzie* '[he said] he wouldn't come',
- interrogative clause (KG): *[nie rozumiem], dlaczego to zrobił* '[I don't understand] why he's done it'.

The manually constructed grammar, for both SWs and SGs, is encoded in the shallow parsing system Spejd (http://nlp.ipipan.waw.pl/Spejd/) [19], a novel open source tool for simultaneous morphological disambiguation (this functionality is not used in this project) and partial parsing with unification.

Spejd rules form a cascade, with the output of one rule constituting the input of the next rule. Therefore rule ordering is crucial. For example, since nominal groups are embedded in prepositional-nominal groups, the rules for the former precede those for the latter.

Spejd rules are created in a conservative fashion, so as to avoid excessive matching, and in order to detect errors on the underlying morphosyntactic level. Firstly, as a parser finds a match for a lemma it is usually checked for grammatical class. For example, in the rule for *nie tylko ..., lecz także* (see below), the word *nie* 'not' must be marked as `conj` and not `qub`. Secondly, rules are made maximally specific in that some SGs are divided into several subtypes, e.g., there are 11 types of nominal groups: `NGa` (`Noun`[4]+`Adj`), `NGs` (`Noun`+`Noun` with the same value of case), `NGg` (`Noun`+$\texttt{Noun}_{\text{gen}}$), `NGk` (`Noun`+and+`Noun`), `NGn` (`Noun`+`Num`), `NGb` (`Noun`+`Brev`[5]), `NGe` (`Noun` as a head of the elective construction), `NGx` (`PPron3`+`Adj`, e.g. *something special*), as well as special groups `NGadres, NGgodz, NGdata`, for describing addresses, hours and dates. So, instead of the plain `NG`, an alternative of subtypes is given in the rule, e.g., `NGa|NGs|NGk|NGg|NGn|NGb`.

A Spejd rule may consist of five elements: `Rule` (rule identifier), `Left` (left context), `Match` (specification), `Right` (right context), `Eval` (conditions and operations). Context specification is optional.

```
Rule    "frazeo: nie tylko [lecz także]"
Match:  [base~"nie" && pos~"conj"]
        [base~"tylko" && pos~"conj"];
Right:  []+ ns [base~","]
        [base~"lecz" && pos~"conj"]
        [base~"także"]?;
Eval:   word(Conj1:discr, "nie tylko");
```

The above rule[6] identifies the first part of a discontinuous

[3] Recall that a SG may contain one or several SWs.

[4] In fact it is a nominal group, not a single noun.

[5] `Brev` stands for an abbreviation.

[6] In this rule `ns` stands for "no space" between tokens, the operator ~ means equal, `&&` denotes logical conjunction, `Conj1` is a grammatical class tag for the first part of the discountinuous conjunction and `discr` is a value of the `continuity` attribute.

conjunction *nie tylko . . . , lecz także* 'not only . . . but also' (note that there must be second part of this conjunction in the right context).

Two types of syntactic operations are available: word, that joins tokens into SWs, and group, that joins SWs into SGs.

The word operation has two mandatory arguments: 1) information about a token in accordance with the tagset (i.e., grammatical class and grammatical category values; pieces of information are separated by colons), 2) the base form of the resulting SW. These two arguments may be preceded by an optional argument: reference to the token which provides some morphological information for the whole SW. In this case the second argument determines how this information should be modified. In Spejd, the token referred to in this way must be unique – it is impossible to inherit information from different components. For example, an analytical future tense (e.g., *będę szedł* 'I will walk') is a combination of future auxiliary (będzie) and past participle (praet). All the information is taken from the bedzie form, except for the gender, which should be taken from the praet form. A solution to this problem is a multiplication of rules. An example of a rule for future tense forms in the feminine (f) is presented below. Here, the gender, instead of being inherited from the third component, is explicitely fixed to be feminine. Similar rules have thus to be created for all other possible genders.

```
Rule    "analytical future tense:
        bedzie + się + praet (f)"
Match:  [pos~"bedzie"] [base~"się"]
        [pos~"praet" && aspect~"imperf"
        && gender~"f"];
Eval:   word(1,Verbfin:fut:ind:refl:f,3.base);
```

The group operation (as in example below corresponding to e.g. *po tych trzech zdaniach* 'after these 3 sentences'), has three arguments: 1) the type of the SG, 2) the reference to the SynHead of the phrase (*po*), 3) the reference to its SemHead (*trzech*).

```
Rule    "PrepNumG: Prep + Adj + Num + Noun"
Match:  [pos~"Prep"] [pos~"Adj|Pact|Ppas"]
        [pos~"Num|Numcol"]
        ([pos~"Noun"] | [type="NG"]);
Eval:   unify(case number gender,2,3,4);
        unify(case,1,3);
        group(PrepNumG,1,3);
```

The problems encountered in pointing at SynHeads and SemHeads were:

- absent heads
  The SemHead of an interrogative clause is a finite verb. In the sentence *Nie wiem, kiedy i ile.* 'I don't know when and how many.' there is no verb in the subordinate clause. In this case the SemHead is made equal to SynHead, here *kiedy*.
- coordination
  In a coordinated group (e.g., *rząd i parlament* 'government and parliament'), the first element is marked as both the semantic and the syntactic head. If the conjunction were the syntactic head of the group, any information about the part of speech, case, etc., of the conjuncts would be lost, which would render such a conjunction group practically invisible to further syntactic rules.

See Tab. I for breakdown of Spejd rules into various types.

TABLE I
TAXONOMY AND QUANTITIES OF SPEJD RULES

Syntactic words			Syntactic groups	TOTAL
multiword entities	abbreviations	others		
339	360	122	242	1063

Spejd rules are applied to a corpus when its underlying morphosyntactic level has already been disambiguated manually. We fully benefit from this fact in our rules. The information about context is used to a lesser degree. Rules are based mainly on morphological information of the matched items themselves. As a result, our grammar performs very well on a good quality disambiguated corpus. However if applied to a non- or poorly disambiguated corpus it would require more matching context data in rules.

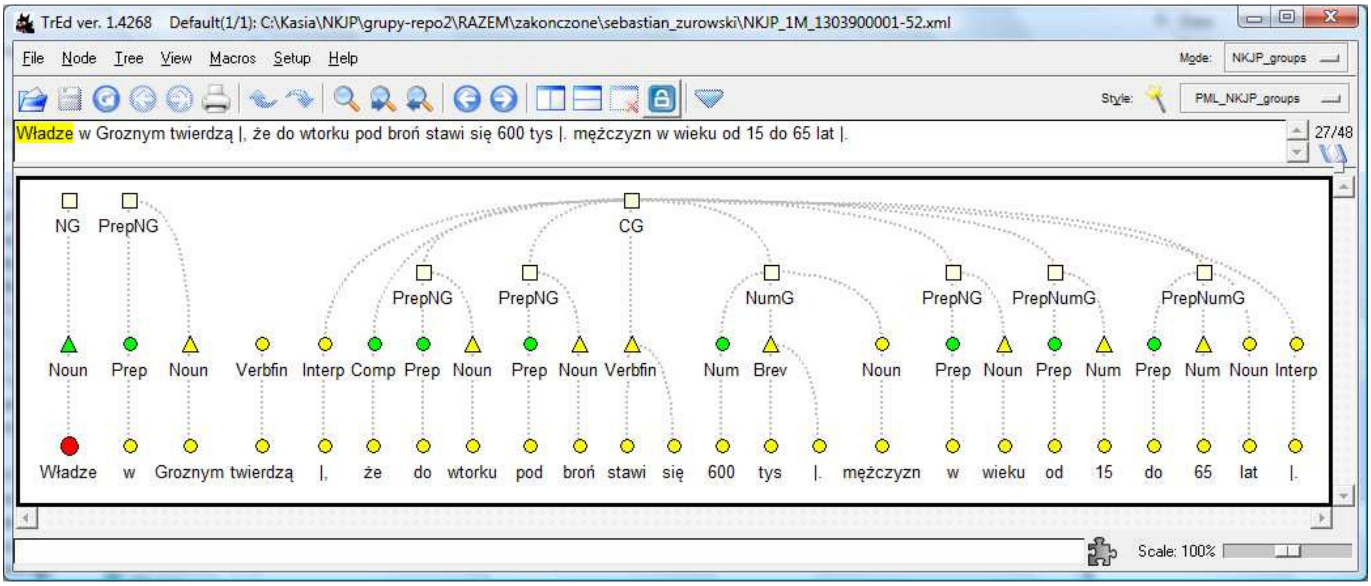
B. Named Entity Recognition with Sprout

As discussed in [16], the automatic pre-annotation of named entities in NKJP is done by the general-purpose knowledge-based NLP platform Sprout [20]. This tool offers several convenient features such as: (i) a rather rich grammar formalism with finite-state operators, unification and cascading, (ii) a very fast gazetteer lookup, (iii) an XML-based output, called Sproutput, in the form of typed feature structures whose type hierarchy can be defined by the user. Existing Polish named entity grammar and resources for Sprout [21] have been extended and adapted for the annotation task in NKJP. They include a gazetteer of about 300,000 inflected forms (55,000 lemmas), and 120 grammar rules for 6 types and 8 subtypes: (i) personal names (persName) with subtypes forname, surname, and additional name (addName), (ii) names of organisations (orgName), (iii) names or geographical objects such as rivers, mountains, etc. (geogName), (iv) names of geo-political units (placeName) with subtypes district, settlement, region, country, and bloc, (v) date expressions, (vi) time expressions. Initial results show the overall precision of 88% and recall of 61%.

V. MANUAL POST-EDITING

Manual post-editing of annotations is the most labor-intensive subtask and requires efficient and user-friendly tools. We have evaluated several annotation platforms such as Synpathy[7], MMAX[8], and GATE [22], before selecting the tree editor TrEd[9] [23] for the following reasons: (i) admitting pre-annotated input and multi-level annotation, (ii) customizable open XML-based abstract data format (PML), (iii) easy manipulation of tree representations, (v) ergonomic customizable

[7] http://www.lat-mpi.eu/tools/synpathy
[8] http://mmax2.sourceforge.net
[9] http://ufal.mff.cuni.cz/˜pajas/tred/


TrEd ver. 1.4268 Default(1/1): C:\Kasia\NKJP\grupy-repo2\RAZEM\zakonczone\sebastian_zurowski\NKJP_1M_1303900001-52.xml
File Node Tree View Macros Setup Help
Mode: NKJP_groups
Style: PML_NKJP_groups
27/48
NG PrepNG CG
PrepNG PrepNG NumG PrepNG PrepNumG PrepNumG
Noun Prep Noun Verbfin Interp Comp Prep Noun Prep Noun Verbfin Num Brev Noun Prep Noun Prep Num Prep Num Noun Interp
Władze w Groznym twierdzą |, że do wtorku pod broń stawi się 600 tys |. mężczyzn w wieku od 15 do 65 lat |.
Scale: 100%

Władze w Groznym twierdzą |, że do wtorku pod broń stawi się 600 tys |. mężczyzn w wieku od 15 do 65 lat |.

Fig. 3. Syntactic annotation with the use of the TrEd editor for the sentence ‘The authorities in Grozny claim that 600 thousand men of 15 to 65 years of age will turn up to arms till Tuesday.’

graphical user’s interface, (vi) parallel edition of concurrent annotations, (vii) rich documentation, (viii) technical reliablility. TrEd is also widely used by the international community – it scored as the second most used annotation tool on the LREC 2010 map of language resources and tools.

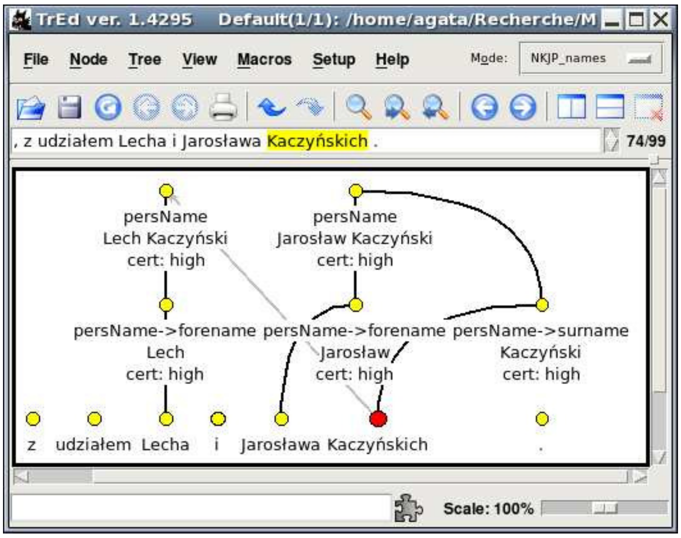
A. Annotator’s Workbench

1) Workbench for Syntactic Words and Groups: As shown in Fig. 3, in the central part of a TrEd’s window the annotation tree of the sentence is shown. Nodes are situated on three horizontal levels, which represent (from bottom to top) segments, SWs and SGs. The annotator can add or remove nodes, and draw edges between them. Each node has a set of type-specific attributes editable in a separate window on double-clicking the node. Toolbar icons are useful for navigation between sentences (or files), as well as undo and redo actions.

For each annotation level special *TrEd extention* has been prepared. NKJP_groups extention for syntactic annotation supplies a set of macros and keyboard shortcuts for: adding a SW or a SG, adding a regular or a secondary edge[10], pointing at the SynHead or SemHead, grouping multiple nodes at a time, etc. It also provides a PML schema defining PML-groups format and a stylesheet with encoded rules of syntactic tree visualisation. In Fig. 3 the SynHead of each constituent is marked in green and the SemHead is marked with a triangle. Thus, *Władze* is both a SynHead and a SemHead while *do* is a SynHead and *wtorku* a SemHead head only.

When closing a file, a final checkup is done via TrEd in that missing SynHeads and SemHeads of each group are reported.

2) Workbench for Named Entities: The annotator’s workbench for named entities is presented in detail in [15]. We recall here its main facilities with respect to the relatively rare but linguistically difficult phenomenon of overlapping in coordination. Fig. 4 shows a sentence in which two coordinated names have been annotated according to the guidelines. Both of them are personal names with an embedded forename and a common embedded surname *Kaczyńskich*, which appears only once. Since TrEd does not allow one tree node to have two father nodes, the highlighted node representing the surname is assigned to one father by a regular edge, and to another one by a secondary edge (in grey). As in all trees, most attributes of a node are visible below it, and their rapid modification is possible by mouse clicks. These attributes include: the type and subtype (*persName→forename*), the lemma (*Lech Kaczyński*), the derivation type and base, if



, z udziałem Lecha i Jarosława Kaczyńskich .

Fig. 4. Annotating coordinated names in the phrase ‘with the participation of Lech and Jarosław Kaczyński’

[10] A secondary edge is used in case of overlapping segments.

any (irrelevant here), and the certainty degree of the annotation (*cert:high*).

B. Workbench for Revision of Annotations

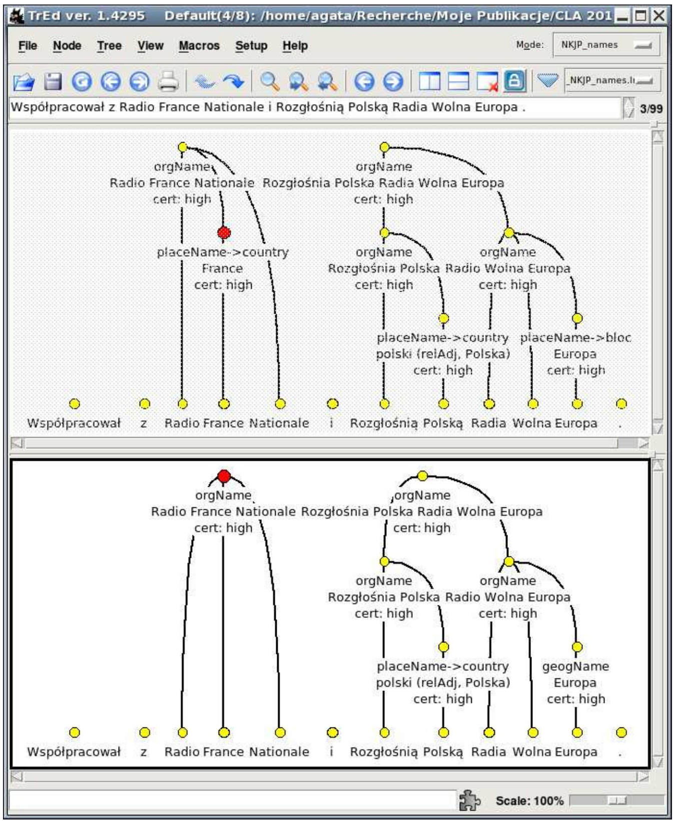
As mentioned above, each text of the gold standard subcorpus is to be annotated at each level by two annotators. Disagreement cases are further reviewed and resolved by an adjudicator (called *super-annotator*), who usually is a person with rich previous experience in annotation at the same level. In order to maximize the objectivity of judgement, the general principle is that: (i) the two annotators of the same text know nothing about each other's results, except what they may learn via the discussion list, (ii) a super-annotator cannot review any portion of the corpus that he or she has previously annotated.

In order for the annotator's work to be most effective, a set of macros and keyboard shortcuts were developed to automatically find discrepancies in two annotations of the same text. Thus, the super-annotator does not review annotations on which the two annotators agree. Another macro exists for an automatic transfer of an annotation between two files. Fig. 5 shows a TrEd screenshot with two NE annotations of the same sentence, containing recursively embedded organisation and location names. The lower window, corresponding to the annotator *a2*, was chosen as the final version of the annotation. However, the upper window, corresponding to the annotator *a1*, contains a node for the country name *France* that hasn't been annotated as a NE by *a2*. The nodes corresponding to this discrepancy are highlighted in red in both windows. By a single keyboard shortcut we can transfer the missing node to the lower window, over node *France* and under node *Radio France Nationale*, so that the remaining nodes remain intact. The automatic detection and transfer of discrepancies act not only on missing or dislocated nodes, but also on a node's attributes. In Fig. 5 the next difference to be highlighted will be the node over *Europa* that has been assigned different types (here *a2* chose the correct type, thus the annotation by *a1* will not be transferred). The same types of macros exist for the revision of disrepancies in annotated SWs and SGs.

C. Managment of Corpus Files

Corpus files management system consists of two main components. The first one is the svn[11] repository, where all files earmarked for annotation are stored. The second element is a textual database (versioned XML file), which contains all information regarding the current state of annotation.

Every annotator has access to his own, private directory in the repository. There he keeps currently annotated files, which he can modify and send back to the `/zakonczone` directory in the subversion repository – target directory for completed files. As a rule, every file will be examined by two different annotators. The annotator does not have the necessary permissions to run all svn operations – he can edit files in the private directory, but cannot add, move or delete files in the repository. The additional functionality – downloading files for annotation and sending off the completed files – is realised by a special `message.txt` file, placed in the private directory. This file works as an interface between the annotator and the subversion server. For example, in order to download five files to his private directory, the annotator has to add the `checkout = 5` line to the `message.txt` file, and run *svn commit* [12] and *svn update* [13] on the directory. The rest of the work – finding appropriate files and moving them throughout repository – is performed on the server side by means of a post-commit subversion hook[14]. Another command, *checkin* (with `checkin = FILE_NAME` syntax), can be used to send annotated files to the `/zakonczone` directory.



TrEd ver. 1.4295 Default(4/8): /home/agata/Recherche/Moje Publikacje/CLA 201

Współpracował z Radio France Nationale i Rozgłośnią Polską Radia Wolna Europa .

Fig. 5. Comparing two NE annotations in TrEd for the same sentence 'He collaborated with Radio France Nationale and the Polish Station of the Free Europe Radio.'

For super-annotation, similar commands, *s_checkout* and *s_checkin*, exist. The *s_checkout* command will download a chosen number of files to compare – every file in two copies, validated by two different annotators. It is guaranteed that the super-annotator will not get files which he has previously seen. The super-annotator corrects one of the downloaded files, using the NKJP_diff TrEd extension to compare it with its second copy (see section V-B), and finally calls *s_checkin* to send corrected version to the `s_zakonczone` directory.

While the `message.txt` file can be modified by the anno-

[11] Version control system, keeps track of changes made in maintained files.

[12] Sends locally modified files to central subversion repository.

[13] Brings changes from repository into local directory.

[14] Process run on server after every `commit` operation.

tator directly, a client-side GUI application – with *[s_]checkout* and *[s_]checkin* functionality – has been developed for annotators' convenience. It fills out the `message.txt` file automatically, thus saving the annotator's effort of editing additional commands manually.

The database – a versioned `db.xml` file – keeps track of every important repository operation. The information about every new file placed in `/nowe` directory is stored automatically in the database. When files are downloaded or sent by an annotator (*[s_]checkout* and *[s_]checkin* operations), his name and the operation date are also saved in the `db.xml` file. There are two main reasons for saving this kind of information in a separate database file. First, it allows to quickly find the information about the current state of the annotation, which is important, e.g., for the implementation of the server-side part of the *[s_]checkout*/*[s_]checkin* operations. Second, it simplifies searching the repository – most of the important information can be obtained from the database, without looking into the repository itself. Extending repository with database brings about the need for additional integrity-preservation mechanism. Atomicity of generic svn operations is guaranteed, but in case of *[s_]checkout*/*[s_]checkin* operations the whole process (commit and post-commit) has to be carried out in one transaction. As a solution to this problem, a FIFO[15] with one element has been set up on the server. Every special operation has to borrow an element from the FIFO in advance (and return it when operation is completed), so two post-commit processes will never modify the repository simultaneously.

To simplify querying the database another tool has been developed. It takes, as command-line arguments, a number of various searching parameters – annotator's name and file name (as regular expressions), file status (checked in or checked out), checkin date range, etc. Another option can be used to extract number of sentences and words from particular files (in this case the tool has to consult the repository, because files statistics are not stored in the database). Additionaly, the tool can be used to find files left for annotation (that is, files which haven't been downloaded by two annotators yet).

D. Project Managment

Multi-level corpus annotation such as in NKJP is a complex and labor-intensive task. To ensure the coherence of annotations, detailed annotation guides have been edited for both tasks, as well as a Frequently Asked Questions list for the NE task. Additionally, NKJP-proper user's guides have been prepared for TrEd and for the svn client tool SVNTortoise[16] used by the annotators. All these documents are regularly updated and diffused via the repository.

Currently, the team working on the two annotation levels consists of three project managers, one programmer, 15 annotators for the syntactic level, and 6 for the named entities. The project managers and the programmer meet on a monthly basis, while communication with the annotators is mainly maintained via discussion lists. The annotation of SWs and SGs seems particularly challenging, as witnessed by the rather rich activity on the corresponding discussion list. During a sample week about 70 messages have been sent to the list containing: (i) mentions of new multi-word entities to be accounted for, (ii) proposals of new grammar rules, (iii) errors on the underlying morphosyntactic level, (iv) various problems with the scope, type and heads of SWs and SGs such as time expressions, fractions, internet and postal addresses, and unexpected syntactic constructions (e.g., *od kiedy* 'since when' is a group of the *preposition-adverb* type, which is not allowed by traditional grammars). The discussion list for NEs mainly receives questions whether a given sequence should or should not be annotated (e.g., names of animals), and doubts about the type and subtype of a NE (e.g., *Palestyna, Kosowo, Arab*).

VI. Links between Two Annotation Levels

Clearly, SGs are tightly connected to NEs. However, as discussed by [24] and [25], a NE does not necessarily precisely coincide with a nominal group. The following types of mutual relations between NEs and SGs were identified in our corpus:

- a NE coincides with a nominal group, e.g., *Stany Zjednoczone* 'United States',
- a NE is a subsequence of a nominal group, e.g., *[ksiądz biskup [Leszek Głódź]$_{persName}$]$_{NG}$* 'priest bishop Leszek Głódź',
- a NE embraces a sequence of SWs and SGs, e.g., *[[Komisja Badań]$_{NG}$ [na Rzecz Rozwoju Gospodarki]$_{PrepNG}$]$_{orgName}$* 'Research Commission for Economical Development'.

A partial overlapping of a NE and a SG seems infeasible, and we plan to detect such problems during the final corpus consistency check.

With the above typology it is clear, on one hand, that a pipelined processing of the annotations on both levels would not be a satisfactory solution. On the other hand, a completely joint processing of both levels seems rather complex, and inconsistent with pre-existing multi-format resources for Polish. Thus, we think that a parallel processing of both annotation levels is a good solution, even if some knowledge must be encoded twice (e.g., some Spejd rules have to cover most frequent types of NEs, described also in more details by the Sprout grammar). We believe that a common project managment of both tasks, enhanced with shared communication means, as described above, helps in assuring the consistence of the annotations.

VII. Present Outcome and Future Work

The annotation on the SW, SG and NE levels is currently at its zenith. Until mid-August, out of about 85,000 sentences of the gold standard subcorpus over 41,000 have been double-annotated for SWs and SGs, and 73,000 for NEs. Thus, 15% through 52% of the corpus remains to be annotated, including particularly demanding extracts of spoken dialogue data. The revision of annotations is done for 14% of the corpus for SWs and SGs, and is just starting for NEs. Moreover,

[15] Named pipe, inter-process communication method.

[16] `http://tortoisesvn.tigris.org`

the whole corpus needs to be revised by super-annotators. Further, the 1 billion-word main corpus will be annotated first with a morphosyntactic tagger trained on the gold standard subcorpus, then with enhanced Spejd and Sprout grammars, finally additional machine learning tools will complete the automatic NE annotation. We are also currently developing the final conversion tool that will allow to obtain the TEI P5-conformant format (cf. [26], [27]) for both annotation levels out of the TrEd PML format.

In future, we expect the NKJP corpus to be used in advanced linguistic corpus studies of Polish, and as a training or evaluation corpus for various linguistic processors such as taggers, parsers, and information retrieval engines.

VIII. Conclusions

We have presented the methodology and the technical environment developed for the annotation tasks in the National Corpus of Polish on three levels: syntactic words and syntactic groups (annotated jointly), as well as named entities. Two rule-based annotation platforms, Spejd and Sprout, are used to automatically pre-annotate the corpus. The Spejd grammar developed within this study and containing currently circa 1063 rules is among the largest chunking grammars for Polish.

An interoperable tree editor TrEd allows for manual correction of annotations by human experts, as well as for the revision of dual annotations by a super-annotator. NKJP-proper extensions, macros, keyboard shortcuts and stylesheets for TrEd enhance the annotator's and super-annotator's workbench. Several XML-to-XML converting tools enable the necessary processing chains between Spejd, Sprout, TrEd and the NKJP TEI P5 conformant encoding standard.

A central versioning repository with custom facilities is responsible for the corpus file managment. We think that its architecture is an interesting alternative to web graphical interfaces used in other annotation projects: (i) it limits the server's charge, (ii) the annotators do not have to rely on constant high-capacity Internet connections – they only connect to the server for down- and uploading the files to be annotated. Here, again, automatic procedures have been developed, such as repository access statistics, automatic creation of file lists, coherence verification, etc. They reduce the annotators' efforts with respect to the manipulated files, facilitate the project management and ensure the security of the corpus.

While the project is still on-going, we think that its solid technical and organisational environment gives it a good chance of success. We also believe that this environment, or substantial parts of it, can be reused or adapted for other annotation tasks.

References

[1] A. Przepiórkowski, R. L. Górski, B. Lewandowska-Tomaszczyk, and M. Łaziński, "Towards the National Corpus of Polish," in *Proceedings of LREC 2008, Marrakech*, 2008.

[2] A. Przepiórkowski, R. L. Górski, M. Łaziński, and P. Pęzik, "Recent Developments in the National Corpus of Polish," in *Proceedings of LREC 2010, Valletta, Malta*, 2010.

[3] S. Acedański, "A Morphosyntactic Brill Tagger with Lexical Rules for Inflectional Languages," in *Proceedings of the 7th International Conference on Natural Language Processing, IceTAL 2010, Reykjavík, Iceland*, ser. LNAI. Berlin: Springer-Verlag, 2010.

[4] A. Przepiórkowski, *Powierzchniowe przetwarzanie języka polskiego*. Warsaw: Akademicka Oficyna Wydawnicza EXIT, 2008.

[5] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of English: The Penn Treebank," *Computational Linguistics*, vol. 19, pp. 313–330, 1993.

[6] S. Brants, S. Dipper, S. Hansen, W. Lezius, and G. Smith, "The TIGER treebank," in *Proceedings of the Workshop on Treebanks and Linguistic Theories*, Sozopol, 2002.

[7] A. Böhmová, J. Hajič, E. Hajičová, and B. Hladká, "The Prague Dependency Treebank: Three-level annotation scenario," in *Treebanks: Building and Using Parsed Corpora*, ser. Text, Speech and Language Technology, A. Abeillé, Ed. Dordrecht: Kluwer, 2003, vol. 20, pp. 103–127.

[8] V. Kordoni and Y. Zhang, "Annotating Wall Street Journal Texts Using a Hand-Crafted Deep Linguistic Grammar," in *Proceedings of the Third Linguistic Annotation Workshop (LAW III) at ACL-IJCNLP 2009, Singapore*, 2009, pp. 170–173.

[9] D. Flickinger, "On building a more efficient grammar by exploiting types," in *Collaborative Language Engineering*, S. Oepen, D. Flickinger, J. Tsujii, and H. Uszkoreit, Eds. Stanford, CA: CSLI Publications, 2002, pp. 1–17.

[10] S. Abney, "Parsing by chunks," in *Principle-Based Parsing*, R. Berwick, S. Abney, and C. Tenny, Eds. Kluwer, 1991, pp. 257–278.

[11] ——, "Partial parsing via finite-state cascades," *Natural Language Engineering*, vol. 2, no. 4, pp. 337–344, 1996.

[12] M. Świdziński and M. Woliński, "Towards a bank of constituent parse trees for Polish," in *Text, Speech and Dialogue: 13th International Conference, TSD 2010, Brno, Czech Republic*, ser. Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag, 2010.

[13] A. Nedoluzhko, J. Mirovský, and P. Pajas, "The Coding Scheme for Annotating Extended Nominal Coreference and Bridging Anaphora in the Prague Dependency Treebank," in *Proceedings of the Third Linguistic Annotation Workshop (LAW III) at ACL-IJCNLP 2009*, Singapore, 2009.

[14] M. Lopatková, N. Klyueva, and P. Homola, "Annotation of Sentence Structure; Capturing the Relationship among Clauses in Czech Sentences," in *Proceedings of the Third Linguistic Annotation Workshop (LAW III) at ACL-IJCNLP 2009*, Singapore, 2009, pp. 74–81.

[15] A. Savary, J. Waszczuk, and A. Przepiórkowski, "Towards the Annotation of Named Entities in the Polish National Corpus," in *Proceedings of LREC 2010*, Valletta, Malta, 2010.

[16] A. Savary and J. Piskorski, "Lexicons and Grammars for Named Entity Annotation in the National Corpus of Polish," in *Proceeding of IIS'10, Siedlce, Poland*, 2010.

[17] A. Przepiórkowski, "On Heads and Coordination in a Partial Treebank," in *Proceedings of the Second Workshop on Treebanks and Linguistic Theories (TLT 2006)*, Prague, 2006.

[18] K. Głowińska and A. Przepiórkowski, "The Design of Syntactic Annotation Levels in the National Corpus of Polish," in *Proceedings of LREC 2010*, Valletta, Malta, 2010.

[19] A. Buczyński and A. Przepiórkowski, "♠ Demo: An Open Source Tool for Shallow Parsing and Morphosyntactic Disambiguation." in *Proceedings of LREC 2008, Marrakech*, 2008.

[20] M. Becker, W. Drożdżyński, H.-U. Krieger, J. Piskorski, U. Schäfer, and F. Xu, "SProUT - Shallow Processing with Typed Feature Structures and Unification," in *Proceedings of ICON 2002, Mumbay, India*, 2002.

[21] J. Piskorski, "Named-Entity Recognition for Polish with SProUT," in *LNCS Vol 3490: Proceedings of IMTCI 2004, Warsaw, Poland*, 2005.

[22] G. Wilcock, *Introduction to Linguistic Annotation and Text Analytics*. Morgan & Claypool, 2009.

[23] P. Pajas and J. Štěpánek, "Recent Advances in a Feature-Rich Framework for Treebank Annotation," in *Proceedings of COLING'08, Manchester*, 2008.

[24] P. Osenova and S. Kolkovska, "Combining the named-entity recognition task and NP chunking strategy for robust pre-processing," in *Proceedings of the Workshop on Linguistic Theories and Treebanks*, Sozopol, Bulgaria, 2002.

[25] J. R. Finkel and C. D. Manning, "Nested Named Entity Recognition," in *Proceedings of EMNLP-2009*, Singapore, 2009.

[26] A. Przepiórkowski and P. Bański, “Which XML standards for multilevel corpus annotation?” in *Proceedings of the 4th Language & Technology Conference*, Poznań, Poland, 2009.

[27] A. Przepiórkowski, “TEI P5 as an XML Standard for Treebank Encoding,” in *Proceedings of the Eighth International Workshop on Treebanks and Linguistic Theories (TLT 8)*, Milan, Italy, 2009, pp. 149–160.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 541–546

ISBN 978-83-60810-27-9
ISSN 1896-7094

TREF – TRanslation Enhancement Framework for Japanese-English

Bartholomäus Wloka, Werner Winiwarter

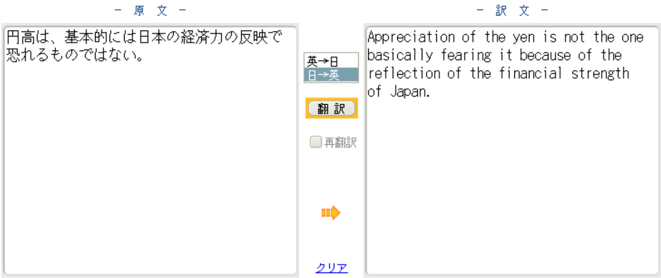
Abstract—**We present a method for improving existing statistical machine translation methods using an knowledge-base compiled from a bilingual corpus as well as sequence alignment and pattern matching techniques from the area of machine learning and bioinformatics. An alignment algorithm identifies similar sentences, which are then used to construct a better word order for the translation. Our preliminary test results indicate a significant improvement of the translation quality.**

Index Terms—**Machine Translation, Syntactical Analysis, Sequence Alignment.**

I. Introduction

MACHINE translation has been an active research area throughout the last 40 years. During this period, many promising concepts were proposed; however, there is still much room for improvement [1]. Especially when translating languages with radically different surface characteristics, as it is the case for Japanese-English, current machine translation techniques tend to produce unsatisfying results. The problems of automated translation between these languages become readily apparent when looking at current Web-based translations, e.g. from `www.excite.co.jp/world/english`, which is shown in Fig. 1. While the translations of short phrases are of reasonable quality, translation systems struggle with long sentences. This is due to the growing complexity of sentences with increasing length and the vast differences in word and subclause order between these langauges. Additionally, the characteristics of the Japanese language pose a great challenge for translation into other languages in general [2], [3]. Those characteristics are:

- two syllabaries and a system of several thousand kanji, i.e. originally Chinese characters with several pronunciations and readings,
- lack of spaces to delimit word boundaries,
- a very high ambiguity in the grammar, as there exist no articles to indicate gender or definiteness,
- the tendency to omit information which can be inferred implicitly,
- sociolinguistic factors, e.g. avoiding direct and decisive expressions for reasons of politeness,
- an extensive system of formality with several levels of politeness forms, honorific expressions, and humble verb forms depending on the social status, relationship and other factors of the people involved.



円高は、基本的には日本の経済力の反映で恐れるものではない。

Appreciation of the yen is not the one basically fearing it because of the reflection of the financial strength of Japan.

Fig. 1. Example of current Web-based machine translation

To overcome those intricacies, we have directed our attention to a new and interdisciplinary approach. We have designed and implemented a method for finding structurally similar sentences with the help of an algorithm usually employed in the field of bioinformatics [4], [5]. The underlying assumption of our approach is that there is a significant overlap between the **structure** of a sentence and its **meaning**. In this paper, we show that it is possible to enhance statistical machine translation results using this assumption. The *TRanslation Enhancement Framework* (TREF) [6] utilizes aligned and clustered sentence pair data to enhance the output of the statistical machine translation system *Moses* [7].

Though trained for the Japanese-English language pair, the system is modular and flexible. An adjustment or extension to other languages is a matter of changing mere implementation details and adding the language-specific resources, such as lexica, parser, corpora, etc. It is important to mention, however, that our translation framework is specifically designed and well-suited for languages with radically different surface characteristics, e.g. European-Asian language pairs.

The rest of this paper is organized as follows: In Sect. II the research relevant to our work is narrated, before we discuss TREF in Sect. III. Section IV presents our evaluation method and the results, followed by a conclusion and future work in Sect. V.

II. Related Work

The ultimate goal of machine translation, i.e. abolishing language barriers, is presented by [8] in an entertaining narration. This ambitious pursuit of a system which will relieve the lingua franca and enable boundless communication between cultures is not quite yet in the realm of the possible. Nonetheless, research efforts towards this goal have been undertaken. In this section, we outline the research relevant to our work.

A. Corpora

A vital resource for machine translation are bilingual corpora. Unfortunately, these are very rare, especially for the

978-83-60810-27-9/09/$25.00 © 2010 IEEE

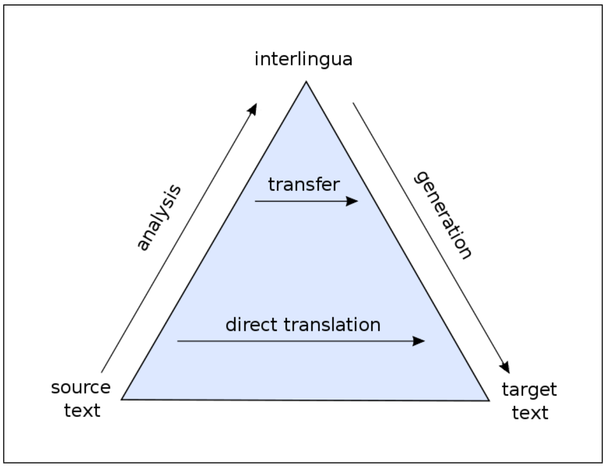


Fig. 2. Translation pyramid

Japanese-English language pair. The currently predominant ones are the *Tanaka corpus* [9], the *Jenaad corpus* [10], and the *Verbmobil treebank* [11]. The Verbmobil treebank contains dialogs from telephone conversations in English, German, Japanese, and other languages, collected during the speech recognition research project of Verbmobil. The Japanese part contains around 160,000 words of text and is written in *Romaji*, i.e. the transcription of Japanese script into Roman literals. The Tanaka corpus consists of roughly 180,000 sentences and has a very broad domain. It has been collected over several years from various sources and compiled by Yasushito Tanaka in 2001. The Jenaad corpus is a collection of close to 150,000 sentence pairs. Extracted from news articles, it offers a certain consistency in terms of sentence types, while still offering a wide range of vocabulary and a variety of grammatical constructs. Because of these qualities, we have chosen the Jenaad corpus for our work. In addition, it is written in Japanese script, thereby avoiding potential ambiguities of the Romaji transcription.

B. Machine Translation

The research in machine translation has ever since included many different approaches. An overview of different techniques can be obtained from [1]. Their visual classification is exemplified by the Vauquois' triangle in Fig. 2 [12]. The historically first method, located at the very top of the triangle, is the *interlingua* approach. It aims towards a language-independent representation, which mediates between two or more languages. In contrast, *statistical machine translation* is at the bottom of the triangle, where no intermediate information is considered in the process, and there is a direct mapping from source to target text, depending on previously trained statistical data. A good overview of this technique can be obtained from [13].

Other approaches, which are also described in more detail in [14], are found somewhere between those two extremes, and the advantage of each depends on the demands of the given language pair. The challenges of translating Japanese to English gave birth to the new idea of *corpus-based* machine translation [15]. Apart from its success in translating between these languages, it further provides the opportunity for enhancing language learning environments by presenting the intermediate steps, i.e. the linguistic analysis of the translation process, to the learner. This was successfully accomplished by [16], [17]. The corpus-based method was quickly adopted by the machine translation community and merged with other techniques, as for example in [18]. Together with the idea of [19], that a mapping of grammatical functions and semantic roles is crucial for the Japanese-English pair, we have decided to mold these ideas into a new approach.

We have chosen a statistical machine translation method for a baseline translation in TREF, since it performs well in terms of translation of individual words and short phrases. It does not adhere to finding transition rules for syntax ordering and therefore leaves a good first candidate for the post-editing done by TREF.

Amongst different tools, we have chosen *Moses*, since it is particularly effective when trained with a sufficiently large bilingual corpus. Moses scores well for structurally similar languages; however, for language pairs like Japanese-English, the word order is disarranged, which significantly lowers the quality of the translation, up to the point where the meaning of the sentence is irrecognizable. Moses does not consider any grammatical rules, so the output is syntactically wrong most of the time. The post-editing and rearranging of the Moses output aims at addressing this problem. Our method finds the correct word order for the translation result and produces a grammatically correct sentence, which conveys the meaning of its English counterpart.

C. Natural Language Processing

To analyze the tokens of our bilingual corpus, we have used the *MontyTagger* from the *MontyLingua* project [20] for English, and *ChaSen* [21] for Japanese. Besides a part-of-speech tagging capability, MontyLingua offers an end-to-end natural language processing toolkit. ChaSen is a high-quality part-of-speech tagger tool for Japanese. Recently, *CaboCha* [22], a Japanese dependency parser, which offers an even wider spectrum of NLP capabilities, has been developed, and we plan to integrate it into TREF in the near future.

D. Sequence Alignment

The Needleman-Wunsch algorithm for computing similarities in protein building blocks, i.e. amino-acid chains, was published in 1970 [23]. Quickly, many derivatives and extensions of this method followed. The basic idea behind this concept was to depict amino-acid chains as strings of alphabetic characters, align them to offer the best match between two strings, and compute a similarity measure [24]. This method was further improved by [25], using a distance measure in conjunction with dynamic programming. Many other research efforts found different distance measures to identify the similarity of sequences. The approach of [26] is generic enough to be extended to the area of machine

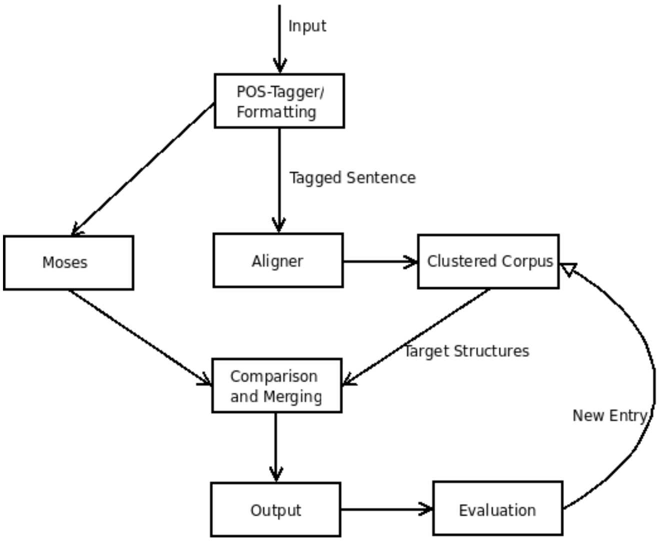


Fig. 3. Overview of dataflow

石炭の利用拡大は大気汚染をさらに悪化させる sekitan no ryou kakudai wa taiki osen wo sarani okka saseru		
石炭/セキタン/2/0/0	の/ノ/71/0/0	利用/リョウ/17/0/0
拡大/カクダイ/17/0/0	は/ハ/65/0/0	大気/タイキ/2/0/0
汚染/オセン/17/0/0	を/ヲ/61/0/0	さらに/サラニ/56/0/0
悪化/アッカ/17/0/0	さ/サ/47/3/5	せる/セル/49/6/1

Fig. 4. Tagged Japanese sentence

expanded use of coal worsens air pollution						
expanded VBN	use NN	of IN	coal NN	worsens VBZ	air NN	pollution NN

Fig. 5. Tagged English sentence

translation, therefore we use it in our research effort by treating sentences from a bilingual corpus analogous to the sequence alignment of amino acid chains.

III. TREF

The overview of the architecture of TREF is shown in Fig. 3. The *PoS-Tagger/Formatting* module tokenizes the input sentence and assigns PoS tags in a format which is described below. The sentences in their tokenized format are then aligned with the clustered corpus to find the target structure, which is sent to the *Comparison and Merging* module. This module takes this input as well as the translation from *Moses* and enhances its translational quality applying a template approach. The resulting translation can then be evaluated and added to the corpus. Each step is described in detail in the following subsections.

A. Part-of-Speech Tagging

The input sentence is sent to either one of the *part-of-speech* (PoS) tagger modules MontyTagger [27] or ChaSen [21]. The result of this process can be seen in Fig. 4 and Fig. 5 for Japanese and English respectively. The Japanese sentence is written in Roman transcription for the reader's convenience. The tags produced by ChaSen consist of a sentence token, its *katakana* representation (one of the Japanese syllabaries, which indicates the pronunciation of a kanji), and a numerical representation of the morphological data. The English tags contain the word itself and the PoS tag as an acronym. After each sentence token is assigned a PoS tag, the sentence and its tags are compared with the sentences already stored in a clustered corpus, which is a customized and enriched version of the *Jenaad* Corpus [10]. We have modified it by removing as much noise as possible, assigned PoS tags to each sentence token, and stored them in an SQL database. We have kept the data with all available PoS tags and additionally created a reduced and optimized tag set, which provides a quick access for efficient processing. Other representations and tag sets can be added easily to satisfy different needs in future work.

B. Aligning and Clustering

In order to identify similar sentences, we have used a slightly modified alignment algorithm from bioinformatics. Instead of aligning protein chains, we align chains of words, i.e. sentences. We have applied relational sequence alignment [4], [5] to obtain clusters of structurally similar sentences. The alignment is done according to the Nienhuys-Cheng distance function.

An example of a distance between the tokens of each sentence is shown in Fig. 6. If the token and its PoS tag differ, the distance is 1. In the case of a structural match, the distance is 0.5, and 0 for a perfect match. The subsequent distance calculation of an entire sentence is depicted in Fig. 7. Gaps, which are identified and symbolized with *(g)* in the example, are assigned variable *gap penalties*. In order to achieve better

d(nn(house),nn(house))	= 0
d(nn(house),nn(office))	= 0.5
d(nn(house),dt(the))	= 1

Fig. 6. Distance calculation example

S1	He	went	to	the	store	to	buy	(g)	milk
T1	PRP	VBD	TO	DT	NN	TO	VB	(g)	NN
S2	She	hurried	to	the	university	to	attend	a	lecture
T2	PRP	VBD	TO	DT	NN	TO	VB	DT	NN
D	0.5	0.5	0	0	0.5	0	0.5	1	0.5

$SentenceDistance$	$\frac{1}{2 \times 9} \times (0.5+0.5+0+0+0.5+0+0.5+1+0.5) = 0.19444$

Fig. 7. Sequence alignment distance calculation example

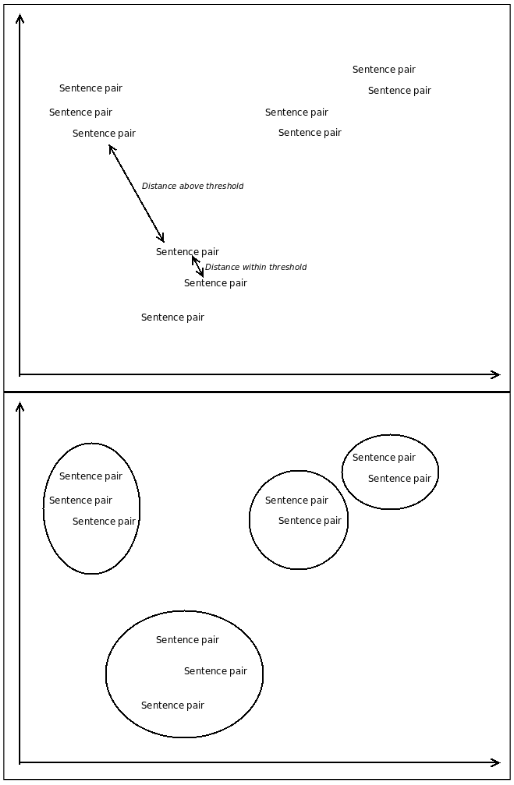


Fig. 8. Clusters in Euclidean space

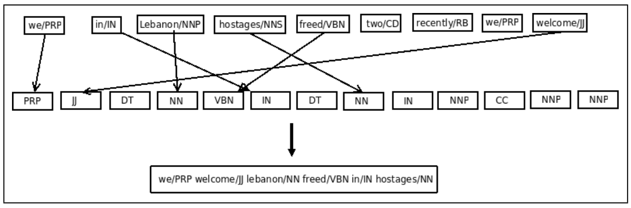


Fig. 9. Matching

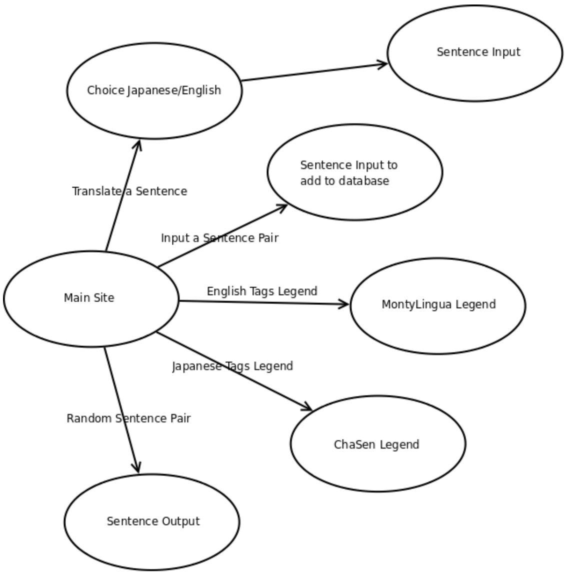


Fig. 10. Structure of the Web framework

matching results, we differentiate between *gap opening* and *gap extension*, which allows us to separate subordinate clauses from otherwise non-matching word sequences.

The similarity measure parameters can be adjusted to fine-tune the result, depending on the text type and text domain. By allowing lower similarity values, a higher number of candidates can be produced, whereas a higher similarity value reduces the number of candidates. This flexibility can be utilized for a language learning application to present an arbitrary amount of similar translations to the student. The output is then evaluated by the user and added to the corpus. Once the distances are computed, clusters can be defined setting a threshold value. This concept is shown in Fig. 8 in a Cartesian coordinate system. Each sentence which has a distance lower than a certain threshold value is assigned to a cluster and is therefore considered *structurally similar* to sentences in this cluster.

C. Comparison and Merging

The comparison of the query sentence with the clusters yields several similar structures. At the same time, the query sentence is processed with Moses to obtain a preliminary translation. This translation is then used to fill the template of the structures which have been found in the previous step. Thereby, a certain number of translation candidates is produced. The filling of the structure templates from the aligning step is shown in Fig. 9. In this example, we use the sentence: “We welcome the progress achieved in the dialog between North and South Korea.” The translation by Moses is: “we in Lebanon hostages freed two recently we welcome”. TREF transforms this by filling the structure template into “we welcome Lebanon freed in hostages”. As can be seen, some tokens are lost in the process of filling the template, which leaves room for future work and potential for further improvement of the translational quality.

D. Web Interface

The clustered corpus of PoS tagged sentence tokens in several representations, as well as morphological information, is stored in a MySQL database and is accessible through a Django Web framework (`http://www.djangoproject.com`). In Django, all interactive content as well as settings, modules, and database setup are written in Python, which made it a good candidate for our system due to its powerful string and text manipulation capabilities. Further, Django provides stable Web development and administrative utilities. In particular, the communication to the database and efficient Web design tools including HTML code inheritance made it an ideal developing environment. The structure of the framework is depicted in Fig. 10. From the main site, the user can navigate to the translation module, the sentence pair input, the random sentence output, as well as legends for the PoS tags for English and Japanese. The translation module offers an interface, which upon input of a sentence sends it to the server and – after the above described translation process – displays the result. The sentence input module takes a sentence pair input, which is flagged as a new addition and is checked manually before being added to the database. The random sentence output is a first step towards the language learning functionality and outputs a sentence from the database including its translation, its tags, and morphological information. We have created a page for the explanation of PoS tags. The translation of the original Japanese ChaSen tags into English is, to the best of our knowledge, the only English ChaSen PoS-tag legend available.

The framework is available on the Web server maintained by the authors under the URL: (`https://wloka.dac.univie.ac.at/project/`).

E. Showcase

Figure 11 shows an example of the workflow from the input of a sentence to an output of several translation candidates. The input ”My name is Yamada.” is tagged and compared with the clustered data. The PoS tags for the sentence in this case are: `My/POP` (personal pronoun), `name/NN` (noun), `is/VBZ` (verb), `Yamada/NNP` (proper noun). The alignment detects sentences in the database, which are similar in terms of words and PoS-tags (see Fig. 6). The translations of the identified structures are also checked for similarities within other clusters. This step, which we call *structure-to-meaning-mapping* identifies other structures of potential translation candidates. These structures are sent to the *matching and translation step*, where the structures and the output from Moses are merged to yield the final output, i.e. the translation candidates.

IV. Evaluation

To create a testing scenario, we have extracted 1000 out of the total 150,000 sentences from the Jenaad corpus. The remaining 149,000 sentences were used as training data for Moses and for clustering. Due to the long processing time for each sentence, we have decided to analyze fewer sentences in detail instead of using standard scoring tools, such as [28] or [29], which would be more significant for larger amounts of output. Morover the validity of automated scoring tools of this kind has been criticized by [14], [30]. Hence our evaluation was done by an expert who judged each translation on four categories: word order, word translations, semantics, and fluency. The categories were equally weighted with a top score of 25 each (see Fig. 12). A total of 40 sample sentences were evaluated, and a statistical significance of the result was verified with a Wilcoxon signed-rank test [31]. The result was a better score for the sentences processed with TREF with a score of W=139 over a sample size of N=34 and a P(1-tail) value of 0.119.

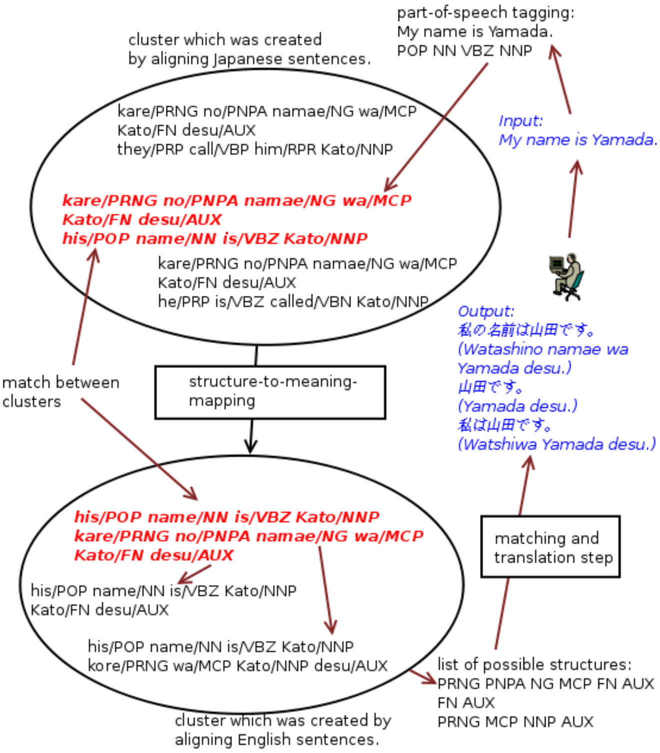

part-of-speech tagging:
My name is Yamada.
POP NN VBZ NNP
cluster which was created
by aligning Japanese sentences.
kare/PRNG no/PNPA namae/NG wa/MCP
Kato/FN desu/AUX
they/PRP call/VBP him/RPR Kato/NNP
kare/PRNG no/PNPA namae/NG wa/MCP
Kato/FN desu/AUX
his/POP name/NN is/VBZ Kato/NNP
kare/PRNG no/PNPA namae/NG wa/MCP
Kato/FN desu/AUX
he/PRP is/VBZ called/VBN Kato/NNP
Input:
My name is Yamada.
Output:
私の名前は山田です。
(Watashino namae wa
Yamada desu.)
山田です。
(Yamada desu.)
私は山田です。
(Watshiwa Yamada desu.)
match between
clusters
structure-to-meaning-
mapping
his/POP name/NN is/VBZ Kato/NNP
kare/PRNG no/PNPA namae/NG wa/MCP
Kato/FN desu/AUX
his/POP name/NN is/VBZ Kato/NNP
Kato/FN desu/AUX
his/POP name/NN is/VBZ Kato/NNP
kore/PRNG wa/MCP Kato/NNP desu/AUX
matching and
translation step
list of possible structures:
PRNG PNPA NG MCP FN AUX
FN AUX
PRNG MCP NNP AUX
cluster which was created by
aligning English sentences.

Fig. 11. Translation via Clustering

Input sentence:	我々は、レバノンにおける復興努力を支持する。
Correct Translation:	We support the efforts of reconstruction in Lebanon.
Moses Translation:	we support in lebanon reconstruction efforts .
Enhanced by TREF:	we support lebanon in reconstruction .

	Word Order	Word Translations	Semantics	Fluency
Moses:	5	15	15	5
TREF:	15	15	20	15
Total Score Moses: 40 Total Score TREF: 65				

Fig. 12. Example evaluation

V. Conclusion

In this paper, we have described a design for enhancing state-of-the-art machine translation using sequence alignment from the area of bioinformatics, combined with PoS tagging and clustering of a bilingual corpus. Our results have proven that similarities in sentence structure can be used to create templates for translation candidates, in particular for the Japanese-English language pair. We have described our implementation of the system and its Web framework. We have trained the system with the Jenaad Corpus and tested the system for Japanese-English. The evaluation of the system yielded promising results. At the time of writing, TREF is already integrated in another research project focusing on ubiquitous translation and language learning with the help of mobile devices.

For future work, we plan to optimize the parameters in the aligning process to fine-tune the word reordering as well as adding grammatical parsing steps after the template filling to improve the syntactical correctness of the sentence. An additional dictionary lookup will be integrated to amend word translations, which could not be processed by the statistical translation step.

We want to extend the language learning aspect of the system to offer a Web-based learning platform and improve the efficiency of the entire system with pre-computing and indexing methods. We plan to incorporate a Japanese dependency parser. The currently active research efforts on the Japanese WordNet [32] and CaboCha [22] are promising candidates for an additional extension for TREF as a language learning platform offering extensive semantic and syntactic information as well as visual representations of vocabulary.

REFERENCES

[1] Y. Wilks, *Machine Translation: Its Scope and Limits*. Springer-Verlag, 2008.

[2] Y. McClain, *Handbook of Modern Japanese Grammar*. The Hokuseido Press, 1981.

[3] S. Makino and M. Tsutsui, *A Dictionary of Basic Japanese Grammar*. The Japan Times, 1986.

[4] K. Kersting, L. D. Raedt, B. Gutman, A. Karwath, and N. Landwehr, *Probabilistic Inductive Logic Programming*. Springer Berlin/Heidelberg, 2008, ch. Relational Sequence Learning.

[5] A. Karwath and K. Kersting, "Relational sequence alignments and logos," pp. 290–304, 2007.

[6] B. Wloka, "Enhancing Japanese-English machine translation – a hybrid approach," Master's thesis, University of Freiburg, 2009.

[7] H. Hoang *et al.*, "Moses: Open source toolkit for statistical machine translation," 2007, pp. 177–180.

[8] N. Ostler, Ed., *The Jungle Is Neutral – Newcomer Languages Face New Media*, Foundation for Endangered Languages 172 Bailbrook Lane Bath BA1 7AA England. University Politecnica de Catalunya Barcelona Spain, 2009.

[9] Y. Tanaka, "Compilation of a multilingual parallel corpus," in *Proceedings of the PACLING 2001*, 2001, pp. 265–268.

[10] M. Utiyama and H. Isahara, "Reliable measures for aligning Japanese-English news articles and sentences," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 72–79.

[11] M. Finke, P. Geutner, H. Hild, T. Kemp, K. Ries, and M. Westphal, "The Karlsruhe-Verbmobil speech recognition engine," *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 1, p. 83, 1997.

[12] W. J. Hutchins and H. L. Somers, *An Introduction to Machine Translation*. Academic Press, 1992.

[13] P. F. Brown, J. Cocke, S. A. D. Pietra, V. J. D. Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin, "A statistical approach to machine translation," *Comput. Linguist.*, vol. 16, no. 2, pp. 79–85, 1990.

[14] C. Boitet, H. Blanchon, M. Seligman, and V. Bellynck, "Evolution of MT with the web," in *Proceedings of the Conference "Machine Translation 25 Years On"*, Cranfield, England, 2009.

[15] M. Nagao, "A framework of a mechanical translation between Japanese and English by analogy principle," in *Proceedings of the international NATO symposium on Artificial and human intelligence*. New York, NY, USA: Elsevier North-Holland, Inc., 1984, pp. 173–180.

[16] W. Winiwarter, "WILLIE – a Web Interface for a Language Learning and Instruction Environment," in *Proceedings of the 6th International Conference on Web-based Learning*. Edinburgh, United Kingdom: Springer-Verlag, 2008.

[17] ——, "WETCAT – Web-Enabled Translation using Corpus-based Acquisition of Transfer rules," in *Proceedings of the Third IEEE International Conference on Innovations in Information Technology*, Dubai, United Arab Emirates, 2006.

[18] M. Carl, A. Way, and W. Daelemans, "Recent advances in example-based machine translation," *Comput. Linguist.*, vol. 30, no. 4, pp. 516–520, 2004.

[19] T. Mitamura and N. Eric, "Hierarchical lexical structure and interpretive mapping in machine translation," in *Proceedings of the 14th Conference on Computational Linguistics*. Morristown NJ USA: Association for Computational Linguistics, 1992, pp. 1254–1258.

[20] H. Liu, "An end-to-end natural language processor with common sense," MIT Media Lab, Tech. Rep., 2004.

[21] Y. Matsumoto, A. Kitauchi, T. Y. Hirano, H. Matsuda, K. Takaoka, and M. Asahara, *Japanese Morphological Analysis System ChaSen version 2.2.1*, 2000.

[22] T. Kudo and Y. Matsumoto, "Japanese dependency analysis using cascaded chunking," in *CoNLL 2002: Proceedings of the 6th Conference on Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops)*, 2002, pp. 63–69.

[23] S. Needleman and C. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, no. 2, pp. 443–453, 1970.

[24] T. Smith and M. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, pp. 195–197, 1981.

[25] A. Karwath and K. Kersting, "Relational sequence alignments," in *Proceedings of the 4th International Workshop on Mining and Learning with Graphs (MLG'06)*, 2006.

[26] A. Karwath, K. Kersting, and N. Landwehr, "Boosting relational sequence alignments," in *Proceedings of the 8th IEEE International Conference on Data Mining*, 2008.

[27] H. Liu and P. Singh, "Conceptnet — a practical commonsense reasoning tool-kit," *BT Technology Journal*, vol. 22, no. 4, pp. 211–226, 2004.

[28] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2002, pp. 311–318.

[29] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrency statistics," in *Proceedings of the ARPA Workshop of Human Language Technology*, 2002.

[30] C. Callison-Burch and M. Osborne, "Re-evaluating the role of BLEU in machine translation research," in *Proceedings of the Conference EACL*, 2006, pp. 249–256.

[31] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, December 1945.

[32] F. Bond *et al.*, "Enhancing the Japanese WordNet," in *Proceedings of the 7th Workshop on Asian Language Resources, in conjunction with ACL-IJCNLP*, 2009.

Bartholomaeus Wloka, MSc is a doctoral student at the Department of Scientific Computing, University of Vienna, Austria. He received his BSc degree in 2005 at the University of South Alabama, USA and his MSc degree in 2009 at the University of Freiburg, Germany. His main research interests are human language technology, machine translation and computer-assisted language learning, in particular combined with mobile learning.

Prof. Dr. Werner Winiwarter is the Vice Head of the Department of Scientific Computing, University of Vienna, Austria. He received his MS degree in 1990, his MA degree in 1992, and his PhD degree in 1995, all from the University of Vienna, Austria. The main research interest of Prof. Winiwarter is human language technology, in particular machine translation and computer-assisted language learning. In addition, he also works on data mining and machine learning, Semantic Web, information retrieval, electronic business, and education systems.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Matura Evaluation Experiment Based on Human Evaluation of Machine Translation

Aleksandra Wojak and Filip Graliński
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
Poznań, Poland
Email: aleksandra.wojak@wp.eu, filipg@amu.edu.pl

***Abstract*—A Web-based system for human evaluation of machine translation is presented in this paper. The system is based on comprehension tests similar to the ones used in Polish *matura* (secondary school-leaving) examinations. The results of preliminary experiments for Polish-English and English-Polish machine translation evaluation are presented and discussed.**

I. Introduction

THE SUCCESS of Statistical Machine Translation, well illustrated by the popularity of Google translation tools, has a positive impact on the development of the whole Machine Translation discipline. This phenomenon brings about the need for comparing the quality of MT tools and systems that keep appearing, both in the academic field and on the commercial market.

In [1] Papineni et al. introduced the BLEU metrics that counts well-translated n-grams (sequences of n words). Improvements of the metrics have been proposed by Doddington [2] (NIST) and Lavie and Agarwal [3] — METEOR. The common feature of those approaches is that evaluation is executed fully automatically, by comparing the text translated by an MT system to the reference translation, prepared by a human.

One of the advantages of automatic evaluation is that it can be used for training. For example, Tenerowicz [4] uses the METEOR metrics in a genetic algorithm that trains the probabilistic grammar used in a parser of the Translatica MT system.

The drawback is its weak correlation with human evaluation. Turian et al. [5] claim that the most popular MT evaluation metrics, BLEU and NIST, fail to correlate well with human judgements of translation quality. In the experiment of Tenerowicz [4], a significant number of translations, improved by the METEOR measure, were estimated as worse translations by linguists.

One of the reasons behind it is the following common feature of automatic evaluation tools: they assign points for parts of sentences, even if the whole sentences are not comprehensible. Points are not assigned for translation adequacy if it is not mirrored by appropriate word strings (although METEOR tries to overcome this drawback by scoring synonyms).

On the other hand, in human simple evaluation (ranking quality of translations or choosing the best one from a given set, when a source sentence is known) evaluators' knowledge of sentence meanings affects the measurement results.

We propose an idea of human evaluation with evaluators being not aware of the source sentence. Evaluators are supposed to give answers to a prepared set of questions, knowing only the target text translated automatically. The evaluation resembles the comprehension test for the Polish *matura* (i.e. secondary school-leaving examination) foreign language exam.

The *Matura* Evaluation obviously measures the comprehensibility of the translated text as well as its adequacy. The latter is achieved by preparing the test questions based solely on the source test.

Please note that our approach does not require evaluators to be native speakers or experts on the target language.

The idea to use comprehension tests for machine translation evaluation is not new [6] [7]. What is new is the use of Web-based application for such purposes.

II. Experiment summary

Matura Evaluation is an experiment for human-based evaluation of machine translation. Its main idea was to compare intelligibility and adequacy of different translations of the same source text, with the correctness of answers being the measurement criteria.

The experiment was performed using two directions of translation between Polish and English. Several source texts were translated by each translation system under test. Source texts came with about 10 questions. Each experiment participant was presented with a random translation of a random text along with a relevant set of questions. The participants were expected to answer these questions using the information provided in the translated text.

It is assumed that if the source text is translated correctly by an MT system, i.e. all the information from the source text can be found also in the translation in an intelligible form, then the participant should easily find the correct answer. On the other hand, if the sentence meaning is changed in

978-83-60810-27-9/09/$25.00 © 2010 IEEE

the translation, the experiment participant obtains the wrong information and, hence, chooses the wrong answer. It is also likely that the relevant information was translated correctly, yet it cannot be inferred from the whole sentence/paragraph where the answer is to be found; or that the translation is difficult to understand. In such a situation the participant is supposed to mark "Translation impossible to understand" as an answer.

III. Texts and Translations Used in the Experiment

Translations were done in two directions between English and Polish. Nine source texts were used in the experiment: five in Polish and four in English. Each of them was translated by each of the three MT systems tested: Google Translate (*http://translate.google.com/*), Kompas (http://www.kompas.info.pl/) and Translatica (http://www.translatica.pl/).

Texts used in the experiment differed in topic and level of difficulty: some of them were supposed to be more specialised (e.g. summary of the *System of Education Act*), other more general (e.g. an article from Wikipedia on *Alice's Adventures in Wonderland*), yet another were parts of literary works (*Little Prince* by A. de Saint-Exupery and *The Deluge* by H. Sienkiewicz). The aim of this diversity was to compare the results for translations of different types of texts. It is well known that it is much more difficult for an MT system (and for human translator as well) to translate literary works than other types of texts, because they contain a large number of metaphors, which cannot be translated literally.

We decided to use real texts, not artificially crafted for the purposes of machine translation evaluation. The following texts were used in the experiment:

1) Polish source texts:
   - article from Wikipedia about the book *Alice's Adventures in Wonderland* (1581 words)
   - part of the first chapter of *The Deluge* by H. Sienkiewicz (2128 words)
   - an article *Aesthetics of the Pythagoreans* (1575 words)
   - summary of the *System of Education Act* (1659 words)
   - an article *Vanishing Venice* (2748 words)
2) English source texts:
   - English translation of the first chapter of *Little Prince* by A. de Saint-Exupery (1753 words)
   - an article about *Greater Poland Uprising* (826 words)
   - an article about the history of St. Patrick's Day celebrations (767 words)
   - an article by Paul Graham *What You Wish, You'd Known* (5083 words)

IV. Questions

Questions were based on the source texts, but written in the target language of the translations. About ten questions based on the source text were prepared in the target language. There were three/four variant answers prepared for each question but only one of them was correct.

Questions were supposed to check if some precise information from the source text had been preserved during translation. Therefore a very specific information was usually expected as an answer to each question.

Various types of questions were prepared for the experiment. Some of them were supposed to check if a word with multiple meanings was translated correctly.

For example, in the text about *St. Patrick's Day* there was a question:
Why did the experiment fail in Savannah?
which the answer to could be found in the following paragraph:
"[...] in 1961, Savannah mayor Tom Woolley had plans for a green river. Due to rough waters on March 17, the experiment failed[...]".
The relevant answer was the correct translation of the word *rough*. There were three answer variants:

- *surowy*
- *szorstki*
- *wzburzony*

All of the answers are different (and, in general, correct) translations of the word *rough* into Polish. However, only the third translation fits the context. It turned out that only one MT system tested (Translatica) translated this word using the correct meaning.

Other questions checked if the meaning of the sentence, possibly with more than one negation word, was not changed (some MT systems have problems with complex negative sentences). It sometimes happens that two negation words, related to two different words in the source text, appear one after another in the translation, thus changing the meaning of the sentence or making it impossible to understand. A sentence from the *System of Education Act* is an example of negation-related problems in translation. The sentence started with the clause *If the child didn't go to nursery school*, which was translated correctly by Kompas and Translatica. However, Google Translate did not manage to translate this sentence correctly. In its translation, the output sentence started with *If the child went to kindergarten*, which totally changed the meaning of the sentence.

Another type of questions was supposed to check the adequacy of translation of compound sentences, especially relative clauses – if the logical relation between parts of the sentences remains the same after translating the source text. However, these relations were usually preserved in translations.

Preparing the questions for such an experiment is quite a challenge, because they should check various aspects of translations. Moreover, it is impossible to check if every sentence is translated correctly. Due to the time limit imposed on the participants, there had to be a limited number of questions to each text. In this experiment we decided that ten questions for each text would be enough to check the general understanding and some chosen specific information.

V. Participants

The *Matura* Evaluation experiment was carried out through the Internet. It was prepared in the form of web application created in Silverlight, so persons taking part in the experiment could access it through the website. Participants were provided with random translations of randomly selected source texts and the corresponding questions. They were supposed to select answers based on the information from the given translation.

The majority of participants taking part in the experiment were students (mainly from the Faculty of Mathematics and Computer Science, but not only). All of them were educated enough to be able to find the correct answer in the text if it was translated clearly and correctly enough. Of course every person has different reading comprehension skills and different deduction abilities, so this experiment should be conducted on a large number of participants for credible and meaningful results. Sometimes it also could happen that the participant knew the answer to the question even without reading the text. It was due to the fact that texts used here were not written for the purpose of this experiment. On the contrary, they consisted of well known fragments of literature works (*The Deluge*, *Little Prince*), articles describing problems which could be known to the participants (Greater Poland Uprising, aesthetic of Pythagoreans etc.). The aim of this experiment was to check the quality of translations, not the knowledge of people taking part in it. Therefore participants were asked to choose answers according to the given text, not their previous knowledge or guesses.

All the participants were Polish native speakers. As the experiment tested the translations between Polish and English in both directions, every participant was supposed to define their English skills prior to its beginning. Texts in English were given only to participants who described their English skills as *good* or *medium*. All the other participants were provided with texts in Polish. Of course the ideal situation would be to give English translations to English native speakers to be sure that if they choose the wrong answer, it is because the text is translated wrongly, and not that the participant's reading comprehension skills in English are too poor. However, we wanted to test how an automatically translated text is perceived by source language native speakers (commercial Polish-English MT systems are usually reviewed in the press or on the Internet by Polish native speakers rather than English native speakers). Therefore, the following solution was used: an additional answer variant was added to each question - *My English skills are not good enough to provide the correct answer to this question*. It was done in order to prevent a participant from guessing the correct answer or choosing the option *translation impossible to understand*, while the translation could be actually quite good but in English too advanced for the participant to understand.

VI. Results of the Experiment

The experiment results are presented in Table I. In the top row of the table the average results (i.e. the percentage of correct answers) obtained by each of the tested MT systems are displayed. As we can observe, all the systems received quite high rates: from 65.45% up to 74.81%. From these figures we can deduce that the translations produced were generally quite understandable, because in average every participant was able to answer correctly about six – seven questions out of ten. This result is quite optimistic, because it implies that an average translation was in about 70% understandable and adequate in reference to its source text.

All the average results presented in Table I are counted using weighted arithmetic mean. Weights depend on a number of times a specific translation was used in the experiment (as mentioned before, translations and texts were chosen randomly for each experiment). Table II indicates how many times each translation of each text was used in the experiment.

When we compare the results obtained by each of the MT systems in both directions of the translations tested in this experiment, we can notice quite a difference. Generally translations from Polish into English received higher marks than translations in the opposite direction. This is even more interesting if we keep in mind that all the experiment participants were Polish native speakers and, hence, able to better understand texts written in their mother tongue, Polish, even after translation. However, the assumption turned out to be false. Polish translations were not only more difficult to understand then the English translations, but texts translated into Polish more often contained wrong information. This could be because English is more difficult to parse than Polish and Polish is more difficult to synthesise than English (because of complex morphosyntactic agreements) and therefore it is more difficult for an MT system to generate a correct and understandable sentence in Polish than in English.

VII. Result Analysis

The interesting fact is that for some texts translation results differ significantly between MT systems. The most essential difference can be observed between the translations into Polish, e.g. Polish translation of *Greater Poland Uprising* translated by Google Translate obtained 73.33% (the best score for translation of this text), while the same text translated by Translatica received an average mark of 33.33%. Quite the opposite results were obtained by these MT systems as far as translation of the article about *St. Patrick's Day* is concerned: Google Translate received the lowest mark for this translation: 37.50%, while Translatica 84.00%. These both results are quite objective, because the translations mentioned above were used in almost the same number of experiments: translation of *Greater Poland Uprising* by Google Translate: 5 times, by Translatica: 6; translation of *St. Patrick's Day* by Google Translate: 4 times, the same amount by Translatica. Translation from Polish into English did not differ so significantly. The largest differences in marks occurred in translation of the most specialised text – *System of Education Act*. Again Translatica translated it in the best way, obtaining 89.47% score, while Google Translate only 66.67%. However, these results cannot be compared in a very credible way, because translation by

Translatica was used 6 times in the experiment, while the translation by Google Translate only 3.

If we want to go deeper in our analysis, we can compare the number of wrong answers which were given to each question after reading translations generated by each MT system. There were two types of answers considered as "wrong": an answer which was not the correct one and an answer saying that *translation is impossible to understand*. The number of such answers was counted for each text and for each translation separately. The most interesting were situations in which one question was answered almost always correctly when using one translation, and the wrong answer was provided based on another translation. Usually it implied that the translation of the paragraph/sentence with information needed to answer the question was much worse in the second case.

An example question with sentences from different translations to illustrate such a situation comes from the article about *Alice's Adventures in Wonderland*. Question no. 8 was not answered correctly by anyone using translation by Google Translate, and it was answered correctly by 3 out of 4 participants using the translation created by Kompas:

Question 8: Who was the author of the first [Polish] translation closest in meaning to the original text?

- *Chuck Connors (1965) – The first translation in line with the original (Google Translate)*
- *Maciej Słomczyński (1965) – the first translation corresponding to the original (Kompas)*

Correct answer to this question is "Maciej Słomczyński". Of course no one reading translation by Google Translate would be able to answer this question correctly because of changed name (*Maciej Słomczyński* was translated into *Chuck Connors*).

Another interesting example comes from the second chapter of *Little Prince* and a question *Over how many parts of the world did the author fly?* All the participants reading this text translated by Google Translate (5 persons) gave the wrong answer, and all the participants using translation made by Translatica (6 persons) answered correctly. The assumption that something was wrong with the translation of Google turned out to be correct. The original sentence *I have flown a little over all parts of the world* was translated by Google into *Mam lotu mało w stosunku do wszystkich części świata*, what gives the wrong understanding that the author has flown *not much*.

VIII. Conclusions

The experiment called *Matura Evaluation* turns out to be a good method of human-based evaluation of machine translation. Results obtained in this experiment show the correspondence with the quality of translations. However, such an experiment has to fulfil some requirements for its results to be credible. First of all, a large number of participants must take part in the experiment. Moreover, all tested translations should be used by similar number of participants. There are also some requirements regarding the experiment preparation: the texts should be correctly selected, different in style and difficulty to enable the comparison of translations of different types of texts.The questions should be clear, prepared based only on source texts for the results of experiment to be objective. All answers should be easy to find in the source text (and, in consequence, in the correct translations), because this experiment does not check the participants' reading comprehension skills, but the quality of translation.

Acknowledgment

The paper is based on research funded by the Polish Ministry of Science and Higher Education (Grant No 003/R/T00/2008/05).

References

[1] K. Papineni, S. Roukos, T. Ward, and W. jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[2] G. Doddington, "Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurence Statistics," in *Proceedings of the Human Language Technology (Notebook)*, San Diego, CA, 2002, pp. 128–132.

[3] A. Lavie and A. Agarwal, "METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments," in *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 228–231. [Online]. Available: http://www.aclweb.org/anthology/W/W07/W07-0734

[4] Z. Tenerowicz, "Zastosowanie obliczeń ewolucyjnych w przetwarzaniu j ezyka naturalnego (Using evolutionary computation in Natural Lanuage Processing)," Master's thesis, Adam Mickiewicz University, Poznań, 2010.

[5] J. Turian, L. Shen, and I. D. Melamed, "Evaluation of machine translation and its evaluation," in *In Proceedings of MT Summit IX*, 2003, pp. 386–393.

[6] M. Tomita, M. Shirai, J. Tsutsumi, M. Matsumura, and Y. Yoshikawa, "Evaluation of MT Systems by TOEFL," in *Proceedings of the Theoretical and Methodological Implications of Machine Translation*, 1993.

[7] M. Fuji, "Evaluation Experiment for Reading Comprehension of Machine Translation Outputs," in *Proceedings of Machine Translation Summit VII*, 1999, pp. 285–289.

TABLE I
EXPERIMENT RESULTS

	Google Tr.	Kompas	Translatica	Weighted avg
Weighted average result	**65.34%**	**71.98%**	**74.16%**	**70.96%**
PL → EN translation	78.17%	80.60%	89.25%	83.64%
EN → PL translation	58.22%	62.21%	59.83%	59.96%
Polish → English translations				
Alice in Wonderland	80.56%	70.83%	88.89%	81.41%
Vanishing Venice	80%	90%	93.33%	90.00%
Pythagorean aesthetics	100%	94.44%	90.7%	92.80%
System of Education Act ...	66.67%	80.43%	89.47%	78.11%
The Deluge – chapter I	80%	78.95%	83.33%	80.64%
English → Polish translations				
Little Prince	56.25%	64.41%	51.67%	57.51%
St. Patrick's Day	37.50%	65%	84%	63.64%
What You Wish, ...	62.50%	62.50%	75%	66.67%
Greater Poland Uprising	73.33%	55.56%	33.33%	53.84%

TABLE II
NUMBER OF EXPERIMENTS PERFORMED

	Google Translate	Kompas	Translatica
Alice in Wonderland	3	4	6
Vanishing Venice	1	2	3
Aesthetics of the Pythagoreans	1	2	5
System of Education Act	3	5	2
The Deluge – chapter I	2	4	3
Total Polish → English	10	17	19
Little Prince	5	6	6
St. Patrick's Day	4	2	5
What You Wish, You'd Known	4	4	4
Greater Poland Uprising	5	3	5
Total English → Polish	18	15	20
Total	28	32	39

ISBN 978-83-60810-27-9
ISSN 1896-7094

German subordinate clause word order in dialogue-based CALL

Magdalena Wolska, Sabrina Wilske
Computational Linguistics,
Saarland University,
Saarbrücken, Germany
{magda,sw}@coli.uni-saarland.de

***Abstract*—We present a dialogue system for exercising the German subordinate clause word order. The pedagogical methodology we adopt is based on focused tasks: the targeted linguistic structure is embedded in a naturalistic scenario, "Making appointments", in which the structure can be plausibly elicited. We report on the system we built and an experimental methodology which we use in order to investigate whether the computer-based conversational focused task we designed promotes acquisition of the form. Our goal is two-fold: First, learners should improve their overall communicative skills in the task scenario and, second, they should improve their mastery of the structure. In this paper, we present a methodology for evaluating learners' progress on the latter.**

I. Motivation

VERBAL communication in a foreign language, actual interaction, is for the language learner the ultimate site of language acquisition. Dialogue is a source of naturally occurring comprehensible input as well as useful negative feedback, reformulations or clarification questions, which may arise from communication problems and which draws attention to correct forms. What is crucial is that dialogue is an opportunity for learners to *produce language* as well as to *modify* their language in response to feedback. All these aspects of conversational interaction have been shown to promote learning [1], [2], [3], [4].

The communicative approch to language teaching advocates the use of goal-oriented realistic communicative activities, *tasks*, in the foreign language classroom, in order to encourage learners to use their developing language [5]. Important definitional characteristics of tasks are: focus on meaning, well defined communicative outcome, and free use of linguistic forms. If a specific grammatical structure is the target of instruction, the latter property of tasks turns out problematic: because learners are free to use any forms they want, it cannot be guaranteed that they will use the forms of interest. To remedy this, *focused tasks*, encouraging the use and processing of specific linguistic features, have been proposed [5]. Focused tasks combine focus on forms with the communicative approach to instruction by, among others, exploiting scenarios which are likely to elicit the structures of interest in an unobtrusive way, promoting thereby their incidental acquisition.

In this paper we report on a system we built and an experimental method designed to find out whether *computer-based* conversational focused tasks also promote acquisition of forms. The structure we targeted was the German word order in subordinate clauses. The goal of the computer-based communicative task was two-fold: On the one hand, learners should improve their overall communicative skills in the task scenario and, on the other hand, they should expand their mastery of the target structure. We have conducted a preliminary small-scale experiment with the system we built in order to assess the feasibility of an in-classroom evaluation. The learning gains results we have observed so far have, however, *not* been statistically significant. In this paper we concentrate on the system itself and the evaluation methodology in general.

The idea of computer-based dialogue activites for foreign langauge learning is not new: computer assisted language learning (CALL) has been an active research field for many years. With the progress in language technology, the number of intelligent CALL systems which allow learners to use natural dialogue has been growing. The study we present is at the intersection of two fields that are intuitively close: second language acquisition (SLA) and Natural Language Processing-enhanced CALL.

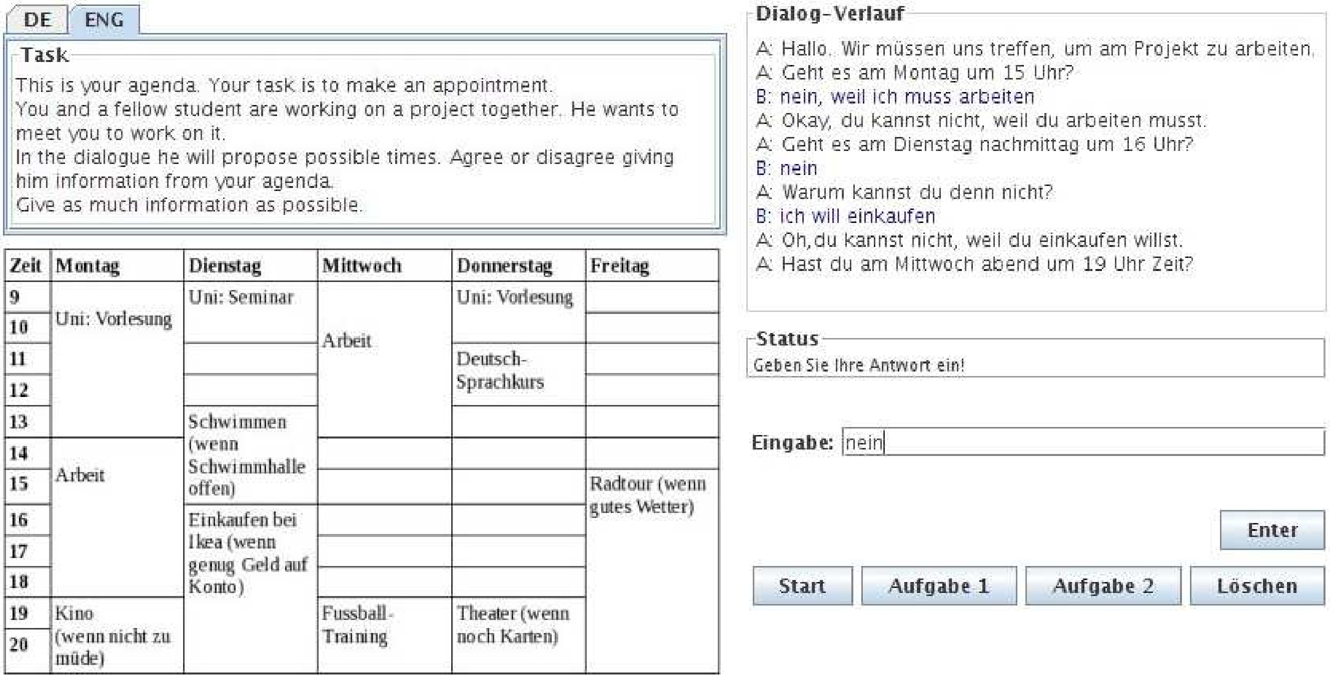
A CALL system can be evaluated in terms of learning gains it generates [6], [7], in terms of its usability (How did learners enjoy playing with the system? [8], [9], [10]) or its performance from an engineering perspective (Did it fail? [11]). We built a CALL system which implements two established SLA methodologies (focused tasks and conversational interaction) and attempt to evaluate whether iteracting with the system produces learning gain.

Outline The paper is organized as follows: In Section II we introduce the linguistic form of interest and the task scenario. In Section III we present two language learning activities we designed. The architecture of the system is outlined in Section IV. In Section V we summarise the setup of an experiment and the results. Section VI concludes the paper.

II. The target form and the task

For the focused communicative activity we selected a grammatical form and a task with the following considerations in mind: Firstly, we wanted a "non-trivial", demanding structure, i.e. a structure of certain complexity, one which is not aquired at the first stages of learning German. Secondly, a structure for

978-83-60810-27-9/09/$25.00 © 2010 IEEE



Task

This is your agenda. Your task is to make an appointment.
You and a fellow student are working on a project together. He wants to meet you to work on it.
In the dialogue he will propose possible times. Agree or disagree giving him information from your agenda.
Give as much information as possible.

Zeit	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
9	Uni: Vorlesung	Uni: Seminar	Arbeit	Uni: Vorlesung	
10					
11				Deutsch-Sprachkurs	
12					
13		Schwimmen (wenn Schwimmhalle offen)			
14	Arbeit				
15					Radtour (wenn gutes Wetter)
16		Einkaufen bei Ikea (wenn genug Geld auf Konto)			
17					
18					
19	Kino (wenn nicht zu müde)		Fussball-Training	Theater (wenn noch Karten)	
20					

Dialog-Verlauf

A: Hallo. Wir müssen uns treffen, um am Projekt zu arbeiten.
A: Geht es am Montag um 15 Uhr?
B: nein, weil ich muss arbeiten
A: Okay, du kannst nicht, weil du arbeiten musst.
A: Geht es am Dienstag nachmittag um 16 Uhr?
B: nein
A: Warum kannst du denn nicht?
B: ich will einkaufen
A: Oh,du kannst nicht, weil du einkaufen willst.
A: Hast du am Mittwoch abend um 19 Uhr Zeit?

Status: Geben Sie Ihre Antwort ein!

Eingabe: nein

Fig. 1. System screenshot; task material (left), dialogue history and input entry field (right)

which we could find a scenario in which it is natural to use, so that we can create incidental opportunites for the learner to produce it. Thirdly, it has enough distinguishing features to be easily tested in a controlled experiment. Finally, in line with task-based teaching, we wanted a meaningful, realistic scenario and communicative task, useful for the learner.

Given the above criteria we opted to focus on the German word order in causal subordinate clauses and framed the usage of these structures in the context of a "Making appointments" scenario. We introduce the two parameters below.

A. *Form: Subordinate clauses*

Subordinate clauses are clauses that are dependent on another clause (main or subordinate). In German, subordinate clauses are characterized by a specific word order: the position of the finite verb in a subordinate clause is at the end of the clause. This position is obligatory. The canonical placement of the finite verb in the main clause is at the second position.

One of the subordinate clause types is an adverbial clause in which the subordinate clause, taking the function of an adverbial, qualifies the action expressed in the main clause by supplying additional information. An example of an adverbial clause is the causal clause which provides the reason for what is said in the main clause.

Examples (1) and (2) below show a causal clause introduced by the subordinating conjunction *weil* (Eng.: because) in the verb-final position and of a single main clause in which the finite verb **muss** is in second position:

(1) . . . weil ich arbeiten **muss**.
because I work-inf must-fin.
'because I have to work'

(2) Ich **muss** arbeiten.
I must-fin work-inf
'I have to work'

The specific word order of subordinate clauses is problematic for learners and it has been shown that it is the last to be acquired by children and adult learners of German [12].

While subordinate clauses are a useful means to structure content and express relations between different propositions, they are used much less in oral communication than in written text. The proportion of subordinate clauses of all clauses ranges from 0.25 to 0.12 in corpora of spoken language, while it is 0.5 in written language [13],[14],[15], as cited by [16]. A possible reason for this dispreference is that subordinate clauses are more complex and thus require more effort to process, which is harder in spontaneous interaction.

B. *Task: Making an appointment*

In order to elicit causal clauses, we created a task in which the learner had to refuse a proposal and, for pragmatic, politeness considerations (a dispreferred second [17]) would likely provide a reason for the refusal.

The scenario in which we embed the focus on subordinate clauses is about arranging a meeting. The task for the learner is to make an appointment given a set of constraints on the available times: the learner is provided with a schedule with a set of occupied and free slots, as illustrated on the left side of Figure 1, and activites planned in the occupied slots (i.e. the reasons for refusals).[1]

[1] The agenda also includes conditionally busy slots, marked with *wenn* (Eng.: if) which can optionally serve to elicit conditional subordinate clauses characterised by the same verb-final word order.

```
get-user-input
if interpretation-found
  if no-justification-found
     elicit-justification
  else
     if TF-realized
        if TF-incorrect
           recast-TF
        else prompt-for-next-contribution
     else
        recast-justification-using-TF
else
  output 'Sorry, I didn't understand.'
```

Fig. 2. Dialogue strategy in the free production activity

The system would propose appointment times known to be occupied on the learner's schedule, thus expecting the learner to refuse the proposal and give a reason. However, keeping in mind that it is not obligatory to provide the reason at all and that a subordinate causal clause is an optional construction, in one of the activites we designed (described below) the system-side of the dialogue was modelled in such a way that it provided examples of the clauses of interest by embedding them in reformulation/paraphrasing utterances and giving them an appearance of implicit confirmation moves.

III. Task-based Activities

We designed and implemented two variants of a role-play type activity framed within the scenario described above: In both variants it involved a *type-written* dialogue with the system we built, with a goal of making an appointment. The system controls the interaction by means of a state-based dialogue model and explicitly implements form-focusing mechanisms: in one variant, this is done as part of the dialogue model, while in the other, by restricting the input mode.

The dialogue model encodes subdialogues which serve to *elicit the target forms* and it *provides feedback on forms* in case of learner form errors. The two variants of the activity differ in the extent of freedom of language production they offer and the realisation of form-focused feedback: one variant allows learners to freely formulate their dialogue contributions (free production) and provides implicit corrective feedback, while in the other learners are asked to produce only the target forms (constrained production) and the feedback merely informs whether the supplied form was correct. We elaborate on the properties of the respective system variants below.

A. Free language production

In the free-production system, the learner is able to type their utterances freely without any restrictions on the language used. The system implements two input interpretation strategies: one based on a grammar with mal-rules, and a fall-back strategy based on keyword matching; details follow in Section IV. It classifies the learner's input into one of the three categories ("TF" stands for "target form"): TF-realized-correct, TF-realized-incorrect, TF-not-realized. The high-level dialogue and feedback strategy is summarised as pseudo-code in Figure 2.[2]

The system provides implicit feedback in case of learner errors in the TF by reformulating (*recasting*) the learner's utterance (or parts thereof). Recasts are realised in a way so as to give them an appearance of implicit confirmation type of grounding moves, as in **S1**, below, which corrects the error made in **L1**:

(3) **L1:** *Nein, ich kann nicht, weil ich muss arbeiten.
'No, I can't because I have to work.'
S1: Ah, du kannst nicht, weil du arbeiten musst.
'Ah, you can't because you have to work.'

The dialogue model encodes three strategies of eliciting causal clauses if the learner does not use them spontaneously: (**A**) If the learner gives a reason for refusal, but does not produce a subordinate clause the system will recast the refusal into a subordinate clause and put emphasis on the conjunction **weil** by setting it in bold face as illustrated below:

(4) **L2:** Nein, ich kann nicht, ich muss arbeiten.
'No, I can't, I have to work.'
S2: Ah, du kannst nicht, **weil** du arbeiten musst.
'Ah, you can't **because** you have to work.'

(**B**) If the learner fails to give a reason in their refusal the system will ask for one explicitly:

(5) **L3:** Nein, am Montag um 15 Uhr kann ich nicht.
'No, I can't make it on Monday at 3.'
S3: Warum kannst du denn nicht?
'Why can't you make it?'

(**C**) In order to present an example of a causal clause not as part of a recast, but as an original refusal-reason pair the system will refuse any learner-initiated proposal with a reason formulated as a causal clause. If the learner does not initiate a proposal the system will try to elicit one by asking the learner what day and time would suit them.

B. Constrained production

In the constrained system the learner's production is restricted to supplying the target form by putting a set of words in the correct order creating a dialogue turn in this way. The words are given in a random order, as in the example below:

(6) **S4:** Kannst du am Montag um 10 Uhr?
'Are you available on Monday at 10am?'
L4: Nein, ich kann nicht, weil ($\boxed{\text{arbeiten}}$ $\boxed{\text{muss}}$ $\boxed{\text{ich}}$)
'No, I can't because I have to work.'

The learner is allowed three attempts to produce the correct form. In case an invalid form is supplied, the system signals it with a message 'That was wrong!' and subtracts one point

[2] We omit some system turns signalling non-undertanding due to unknown words to simplify the presentation.

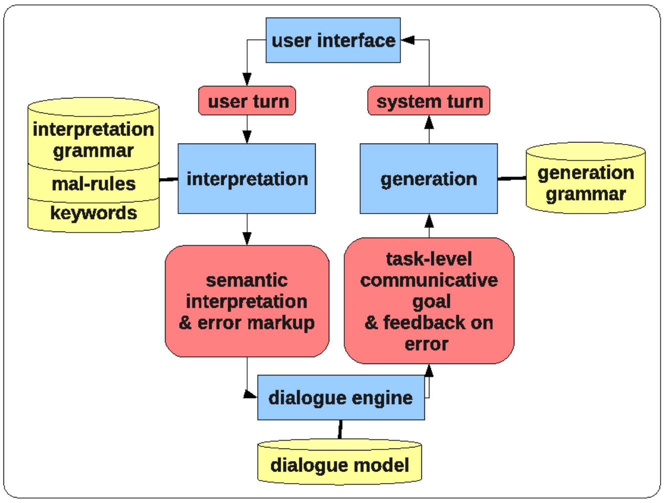


Fig. 3. The system architecture

from a learner's "score" on the activity; correct forms increase the score by one. The feedback and the score are displayed in a designated feedback area. After the third unsuccessful attempt the correct utterance is appended to the dialogue. The system then generates its next turn based on the dialogue model.

The following section summarizes the architecture and the implementation of the system.

IV. The system

Both dialogue activities are implemented on the same system architecture; we concentrate on the components required for the free production activity because the constrained production activity is its simplified variant.

The system maintains a dialogue with the learner by following a dialogue strategy outlined in Section III (see Figure 2). This involves interpreting the learner's input, responding to the learner by selecting a communicative goal according to the dialogue model and the paedagogical strategy, and realizing the goal as a surface string. Specifically for the learning context, the system has to recognize errors in the learner input and generate feedback on them.

Figure 3 shows the system's architecture: the modules and the flow of information between them. We describe each of the functions below.

A. *The task and dialogue model and the dialogue engine*

The dialogue model represents the sets of possible turn transitions: alternating turns produced by the user and the the system. Task-related parameters, the information about the slots in the time-table, are encoded in an external data structure which is imported into the dialogue model.

The dialogue model is implemented as a state machine using State Chart XML (SCXML) as an underlying representation. We use the Java implementation of Apache SCXML.[3] The framework also provides a dialogue execution engine which receives input interpretations and triggers the system responses according to the given model.

B. *Interpretion of learner's input*

In general, interpreting the user input involves mapping a surface string of an utterance to a meaning representation. As typical in small-scale dialogue systems, we implement the system's language model (the set of linguistic expressions) as a context free grammar with semantic tags. For parsing, we use the Java Speech API implementation of the CMU parser which is part of the Sphinx system.[4] The semantic tags encode two types of information: first, the symbolic meaning of utterances, and second, information on violations of grammatical constraints; more on error handling below.

1) Fuzzy matching for unknown words: In order to ensure robustness with respect to typos and spelling errors the system first identifies unknown words in the input and tries to map them to known words by calculating the Levenshtein distance between the unknown word and known words. Candidates for replacement of out-of-vocabulary words are those known words which have a Levenshtein distance within a certain range normalized by word length.

2) Grammatical error handling: Since the system interacts with learners, i.e. non-native speakers of German, their input is likely to contain other errors apart from misspellings, in particular errors in the target structure. An essential requirement of the system is to recognize those errors and give feedback on them. One strategy to deal with errors is to explicitly integrate anticipated errors into the grammar in the form of so called mal-rules, i.e. grammar productions which are outside of the standard rules of the given language. Erroneous utterances are parsed using mal-rules and the parse result contains information about the error.

The drawback of this approach is that it is hard to anticipate all possible errors that might occur. Therefore, our system also implements a fall-back strategy based on keyword spotting: If no parse is found for an utterance, we create a semantic interpretation based on content words, using a keyword lexicon. We encoded a set of mal-rules based on informal prior pre-testing of the system with beginner learners.

C. *Generation of system responses*

The system output realization is performed using a template-based approach. The output is produced by generating a dialogue move selected according to the dialogue model using a context free generation grammar. The grammar associates atomic keys representing communicative goals with sets of possible realizations. Slots in the generation templates are filled using feature-value pairs passed as arguments to the templates along with the communicative goals to be realized.

D. *User interface*

The user interface is implemented as a Java applet embedded in a website. The applet displays the task material,

[3] http://commons.apache.org/scxml

[4] Available at http://cmusphinx.sourceforge.net

an input field for learner, the dialogue history and additional buttons for editing the input utterances and selecting specific tasks. Figure 1 shows the graphical interface including the task-material (the agenda; left) and a part of a dialogue history (upper right) with an example of a system recast (as in (3)) and of an elicitation strategy (as in (5)).

V. Evaluation

In this section we present an evaluation methodology which we use to evaluate learning gains produced by the system variants discussed above. The evaluation is based on in-classroom activities with the system we built.

We have conducted a preliminary small-scale experiment using the experimental design introduced below, however, the results we have obtained so far have not been statistically significant. Therefore, we present the current results only very briefly in Section V-B.

A. Methodology

Design The experimental design we use is a nonrandomized pretest multiple-posttest design involving students from German language classes at the university, taught by different teachers. The classes are split randomly into two sub-groups: one assigned to the free production condition, and the other to the constrained production condition. We are interested in two questions: 1) whether the interactive activities produce learning gains, and 2) whether the free production condition, which requires more computational effort (e.g. in interpreting the learners' turns), produces more gains than the activity in which the learners' production is restricted.

Procedure At the first session, time 1., both groups complete a pretest, then interact with one of the system variants (repeating the exercise twice), and subsequently complete an immediate post-test (posttest1). At the next session a week later, time 2., the groups again perform an in-classroom exercise with the system in a different configuration of the schedule and complete another post-test at the end of the session (posttest2). Finally, time 3., after a couple of week's break (five weeks in the pilot study) the groups complete another post-test (delayed posttest).

After the second session the participants fill out a demographic/learning history survey (anonymous) and a feedback questionnaire on the interaction with the system; we ask about the usability, usefulness, interest in future use, etc.

We provide two different variants of the task for each participant. The basic scenario frame, a weekly time schedule, is kept, but the character of the system is changed: In the first variant, the system takes the persona of a fellow student, whereas in the second it is introduced as a learner's supervisor. The motivation for the latter is that an interaction with a superior might produce behaviour which more closely conforms to the politeness norms, in this case, providing reasons for declining a proposal and hence also the target forms. Also the set of days and times which the system proposes is different for each exercise. In the first repetition of the activity, the system makes 5 different proposals, for the second it makes 4 proposals, thus we expect 9 uses of causal subordinate clauses according to our dialogue script.

Tests We use two types of assessment tests: a timed grammaticality judgment test, targeting implicit knowledge, and an untimed sentence construction test, targeting explicit knowledge. Implicit knowledge refers to knowledge accessible through automatic processing and which learners are intuitively aware of, while explicit knowledge is knowledge accessible through controlled processing [18].[5]

a) Timed grammaticality judgment: Following Ellis [19] we designed a timed grammaticality judgment test to measure implicit knowledge. The test items include causal subordinate clauses of different complexity. The complexity varies as to the amount of additional material present in the clause, e.g. objects, modal verbs, negations or additional modifiers.

The test consists of 6 grammatical, 6 ungrammatical test items and 9 grammatical and 9 ungrammatical distractor items, including a subset of other subordinate clauses. We set the time-limit for the test to 10 seconds per item. This is roughly twice the maximum time a native speaker used. Ellis timed his test at 20% above the average time native speakers needed [19]. Han and Ellis used 3.5 seconds as the time constraint in [20] based on pretesting the items, while Bialystok used an even shorter time limit [21]. Based on our own pretest with native speakers, we performed, already the threshold of 3.5 seconds would have excluded a couple of slow native speakers. Since we are not aware of research which explicitly addresses the issue of the time limit on the timed judgement tasks, we opt for a more generous time-limit.

Each correctly judged item is scored at 1 point, each incorrectly judged item is scored at 0.

b) Sentence construction: For the explicit knowledge test, participants are asked to complete sentences given the beginning of a sentence and a set of unordered uninflected phrases or words as in the example below:

Item: Ich kann nicht (weil, arbeiten, müssen, ich)
Solution: Ich kann nicht, weil ich arbeiten muss.
'I can't because I have to work.'

The test consists of 6 test items for causal conditional clauses. There is no time-limit. The items are scored at 1 point if the word order is correct, 0 otherwise. All form errors other than those in the target structure are neglected.

We created four versions of the tests described above to be administered at the four times of assessment (pretest, posttest1, postest2, delayed postest). The versions differ in the combinations of lexemes, but are otherwise comparable with regard to complexity of the lexical items used. The assignment of a test version to a time varies between participants in order to compensate for unintended differences between versions. Within each test, items are presented in random order.

[5] The tests are prepared and administered using Webexp Experimental Software (`http://www.hcrc.ed.ac.uk/web_exp/`).

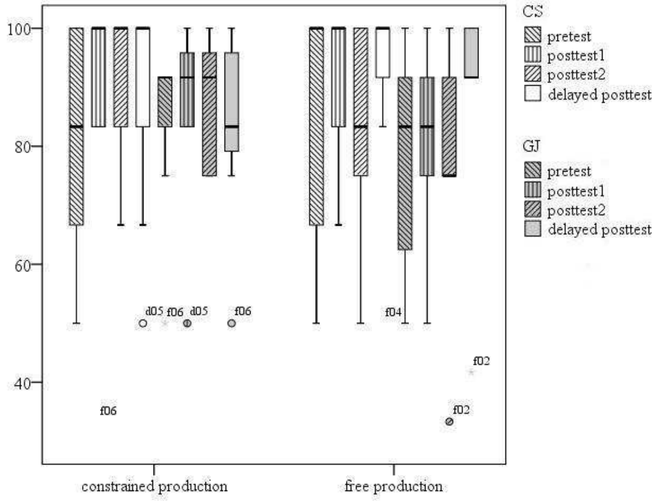


Fig. 4. Overall results for sentence construction (SC; white) and grammaticality judgement (GJ; gray) for both conditions

TABLE I

NUMERICAL RESULTS OF THE SENTENCE CONSTRUCTION TEST: MEANS (M) AND STANDARD DEVIATIONS (SD) FOR PERCENTAGE SCORES

		Pretest		Posttest 1		Posttest 2		Delayed Posttest	
	N	M	SD	M	SD	M	SD	M	SD
Percentage scores Constrained	7	80.95	20.25	85.71	24.40	90.48	16.26	88.10	20.89
Free	7	83.33	21.52	90.48	16.26	83.33	19.24	90.48	18.90

TABLE II

NUMERICAL RESULTS OF THE TIMED GRAMMATICALITY JUDGMENT TEST: MEANS AND STANDARD DEVIATIONS FOR PERCENTAGE SCORES

		Pretest		Posttest 1		Posttest 2		Delayed Posttest	
	N	M	SD	M	SD	M	SD	M	SD
Percentage scores Constrained	7	83.33	15.96	85.71	17.16	86.90	11.64	83.33	17.35
Free	7	77.38	19.07	80.95	17.16	77.38	22.42	88.10	20.89

B. Pilot experiment

We conducted a small-scale experiment, using the setup described above, in order to assess the feasibility of an in-classroom evaluation and in order to get an impression of whether the learners benefit from the system(s). As we had mentioned earlier the results we have obtained so far have *not* been statistically significant in neither of the groups, however, the usability questionnaires and the feedback we got from the participants of the study encourage us to pursue the free-form exercises further. We are therefore planning to conduct another analogous experiment in the coming fall semester, however, we will invite learners from German courses at a lower proficiency level than the groups we had access to for the pilot study. Below, we briefly outline the current results.

Participants For the pilot experiment we had access to 26 learners from two German language courses. The participants came from different language backgrounds, were both male and female, with an average age of 25 years, and had been learning German for an average of about two years prior to experiment.[6] The courses met twice a week for 90 minute sessions. The experiment started 6 weeks (ca. 15 instruction hours) into the course.

The subjects participated in two sessions of in-classroom exercises with one of the system variants with one week's break between the sessions. Each session consisted of at least two repetitions of the activity in different configurations of the task material (the time schedule) as described in Section II. To complete the activity the participants took between 5 and 25 minutes in the free condition and between ca. 2 and 10 minutes in the constrained condition.

With the experiment spanning over a few weeks, subject drop-out was inevitable. Due to a high course drop-out rate (42%), at this point we have data for only 15 subjects for all the four assessment points for both tests: 7 subjects in the free production condition and 8 in the constrained production. We removed a set of data for one subject in the constrained condition who obtained full scores at all the assessment points obtaining a data set with two groups of 7 subjects.

Analysis Because of the small sample size and because of the violation of parametric assumptions[7] we perform non-parametric analyses: in order to compare within subject differences we use the Friedman test.[8] For between groups comparisons we use the Mann-Whitney U test. We set the significance level at 0.05.

Results and discussion The overall results are shown in Figure 4. Because of the small sample size, the skewed distribution of scores (as seen in the figure) we cannot draw any ultimate conclusions: In both groups the repeated measures statistic turned out to be not significant. The reason for this is likely to be the fact that both of our groups started off with relatively high scores. There was no significant between-group difference on the pretest, according to Mann-Whitney U test, i.e. the groups started off at the same level. However, this pretest level was at an average of 81% and 83% of the total scores on the sentence construction test in the constrained and free production condition respectively, and 83% and 77% in the free production. That is, the subjects were perhaps too familiar with the target structure.

Table I and Table II show the percentage scores' means and standard deviations for the pretest, posttest1, posttest2, and delayed posttest for the *sentence construction test* and the *grammaticality judgement test*. The general pattern in the scores on both tests is the same: Both groups increased accuracy in the use of subordinate clause word order from pretest to posttest1, however, while the constrained production group further increased between posttest1 and posttest2, the

[6] Their German proficiency level was classified as ranging from A2 to B1+ CEF level, based on scores on an initial course placement test.

[7] According to Shapiro-Wilk and Levene tests both the normality assumption and the assumption of homogeneity of variance were violated on at least some of the within-subject and/or between-subject variables on either tests in the pilot study.

[8] Since we did not obtain a statistically significant result, we did not perform post doc tests at this time.

free production group declined. The accuracy then declined between posttest2 and delayed posttest in the constrained condition, but improved in the free production condition. As mentioned above, these differences were, however, not statistically significant at the level we had set. Neither were the differences in between-group comparisons of the scores. Some of the differences were, however, marginally significant at a more liberal level of 0.1.

As mentioned at the beginning of this section, we do not draw ultimate conclusions as to the learning gains based on the pilot study results. We believe that the experiment is definitely worth re-running in a course at a lower proficiency level than the one we were working with at this time. We are planning such an experiment for the coming fall. The free production system was rated significantly higher on the questionnaire than the constrained production system and some of the lower scoring learners did declare that they would like to use such a system for at-home exercises, were it available.

VI. Conclusion

We presented an architecture of a dialogue system for interactive computer-based exercises for the German subordinate clause word order. The exercises are designed based on an established communicative-teaching methodology of focused tasks. The system implemets elicitation mechanisms which cue the learner on using the target structure of interest. In one exercise variant the target structures are ellicited in an unobtrusive way, while the other exercise has more of a drill-like character. We also presented an evaluation methodology for assessing learning gains upon interaction with the system.

While our small-scale pilot study was not conclusive, we believe our appoach is worth pursuing further based on encouraging feedback from the participants of the pilot study. One conclusion we might perhaps draw is that the exercise mode we propose is more suitable at an earlier stage of acquisition of the form we target, i.e. around the time when the learners are first introduced to the subordinate clause word order, rather than being familiar with it already. However, we would certainly need another experiment to confirm this and we do intend to conduct a larger scale study using the methodology we described.

Based on the same architecture and a modified implementation, we also built another dialogue activity for exercising the German locative use of the Dative case, framed in a "Directions giving" scenario. In an analogous pilot study, using the same evaluation methodology as described in Section V, we found significant improvement in the use of Dative upon exercising with our system [22].

Acknowledgements

We would like to thank Dr. Kristin Stezano Cotelo and Meike van Hoorn, language instructors at the German courses of the International Office at the Saarland University, as well as their students who participated in the experiments, for the help in conducting this study.

Magdalena Wolska's position at Saarland University is funded through the INTERREG IV A project ALLEGRO (Project No.: 67 SMLW 1 1 137).

References

[1] M. H. Long, "Input, interaction and second language acquisition," in *Native language and foreign language acquisition*. Annals of the New York Academy of Sciences, 1981, vol. 379, pp. 259–78.

[2] S. D. Krashen, *Input Hypothesis: Issues and Implications*. London: Longman, 1985.

[3] M. Swain, "Communicative competence: Some roles of comprehensible input and comprehensible output in its development," in *Input in second language acquisition*, S. Gass and C. Madden, Eds. MA: Newbury House, 1985, pp. 235–53.

[4] ——, "Three functions of output in second language learning," in *Principle and practice in applied linguistics: Studies in honour of H.G. Widdowson*, G. C. . B. Seidlhofer, Ed. Oxford: Oxford University Press, 1995, pp. 125–144.

[5] R. Ellis, *Task-based Language Learning and Teaching*. Oxford University Press, 2003.

[6] V. M. Holland, J. D. Kaplan, and M. A. Sabol, "Pre-liminary tests of language learning in a speech-interactive graphics microworld," *Calico Journal*, vol. 16, no. 3, pp. 339–359, 1998.

[7] W. Harless, M. Zier, and R. Duncan, "Virtual dialogues with native speakers: The evaluation of an interactive multimedia method," *Calico Journal*, vol. 16, no. 3, pp. 313–37, 1999.

[8] C. Wang and S. Seneff, "A Spoken Translation Game for Second Language Learning," in *Proceedings of the Conference on Artificial Intelligence in Education*, 2007, pp. 315–322.

[9] T. Lech and K. de Smedt, "Dreistadt: A language enabled moo for language learning," in *Proceedings of the ECAI-06 Combined Workshop on Language-enabled Educational Technology and Development of Robust Spoken Dialogue Systems*, 2006, pp. 38–44.

[10] W. L. Johnson and S. Wu, "Assessing aptitude for learning with a serious game for foreign language and culture," in *Intelligent Tutoring Systems*. Springer Berlin / Heidelberg, 2008.

[11] S. Seneff, C. Wang, and J. Zhang, "Spoken conversational interaction for language learning," in *Proceedings of the InSTIL Symposium on Computer Assisted Language Learning*, 2004, pp. 151–154.

[12] M. Pienemann, *Language Processing and Second Language Development: Processability Theory*. Benjamins, 1998.

[13] K. Bayer, "Verteilung und funktion der sogenannten parenthese in texten gesprochener sprache," in *Forschungen zur gesprochenen Sprache und Möglichkeiten ihrer Didaktisierung*, Goethe-Institut, Ed. Kemmler, 1971, pp. 200–214.

[14] U. Engel, "Syntaktische besonderheiten der deutschen alltagssprache," in *Gesprochene Sprache*. Pädagogischer Verlag Schwann, 1974.

[15] J. Weijenberg, *Authentizität gesprochener Sprache in Lehrwerken für Deutsch als Fremdsprache*. Groos, 1980.

[16] F. Bubenheimer, "Grammatische besonderheiten gesprochener sprache und didaktische konsequenzen für den daf-unterricht," Published online at http://www.deutschservice.de/felix/daf/gesprkom.html.

[17] S. C. Levinson, *Pragmatics*. Cambridge University, 1983.

[18] R. Ellis, S. Loewen, and R. Erlam, "Implicit and explicit corrective feedback and the acquisition of L2 grammar," *Studies in Second Language Acquisition*, vol. 28, pp. 339–368, 2006.

[19] R. Ellis, "Modelling learning difficulty and second language proficiency: The differential contributions of implicit and explicit knowledge," *Applied Linguistics*, vol. 27, no. 3, pp. 431–463, 2006.

[20] Y. Han and R. Ellis, "Implicit knowledge, explicit knowledge and general language proficiency," *Language Teaching Research*, vol. 2, pp. 1–23, 1998.

[21] E. Bialystok, "Explicit and Implicit Judgements of L2 Grammaticality," *Language Learning*, vol. 29, pp. 81–103, 1979.

[22] M. Wolska and S. Wilske, "Form-focused task-oriented dialogues for computer assisted language learning: A pilot study on german dative," in *Proceedings of the Interspeech-2010 Workshop "Second Language Studies: Acquisition, Learning, Education and Technology"*, 2010, To Appear.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Polish Phones Statistics

Bartosz Ziółko, Jakub Gałka,
Department of Electronics
AGH University of Science and Technology
al. Mickiewicza 30, 30-059 Kraków, Poland
www.dsp.agh.edu.pl
{bziolko,jgalka}@agh.edu.pl

***Abstract*—The phonemic statistics were collected from several large Polish corpora. The paper presents methodology of the acquisition process, summarisation of the data and some phenomena in the statistics. Triphone statistics apply context-dependent speech units which have an important role in speech technologies. The phonemic alphabet for Polish, SAMPA, and methods of providing phonemic transcriptions are described with detailed comments.**

***Index Terms*—Natural language processing, triphone statistics, speech processing, automatic speech recognition, Polish**

I. Introduction

The paper describes processing of linguistic data and constructing the statistics of Polish phone system by analysing large amount of text corpora using Cyfronet high performance computer cluster. There is a trade-off of quality of such statistics and time spent on calculations. The high performance computers enable obtaining the linguistic rules from the vast number of texts in reasonable time.

Statistical linguistics at the phone, word and sentence level are under considerations for several languages [2], [4], [9], [1]. The frequency of phonetic units appearance can be used in several speech processing applications, for example modelling in automatic speech recognition (ASR). Models of triphones which are not present in a training corpus of a speech recogniser can be prepared using phonetic decision trees [11]. The list of possible triphones has to be provided for a particular language along with phones' categorisation. The triphone statistics can also be used to generate hypotheses used in recognition of out-of-dictionary words including proper names or to provide additional probabilities in speech modelling (Fig. 1).

Some similar statistics collected from a few large corpora: Rzeczpospolita corpus (containing articles from a newspaper) [13], literature corpus [12] and Wikipedia corpus [14] (over 250 000 000 words) have been already presented. However, a space was included in the list of phones in the previous publications. This was not necessarily a good decision because coarticulation happens between words as well. The process of speaking combines toegether several words and the borders between words are often indistinguishable on phonetic level.

Context-dependent modelling can significantly improve speech recognition quality. Each phone varies slightly depending on neighbouring phones due to a natural phenomena of coarticulation. There are no strict boundaries between phones because they overlap each other. It results in interference of acoustic properties. Speech recognisers based on triphone models rather than phone ones are much more complex but give better results [10]. Let us introduce examples of different ways of transcribing word *zero*. The phone model would be *z e r o* while the triphone one is **-z+e z-e+r e-r+o e-r+**. In case a specific triphone is not present, it can be replaced by a phonetically similar triphone (phones of the same phonetic group interfere in a similar way with their neighbours) using phonetic decision trees [11] or biphones (applying only left or right context) [8].

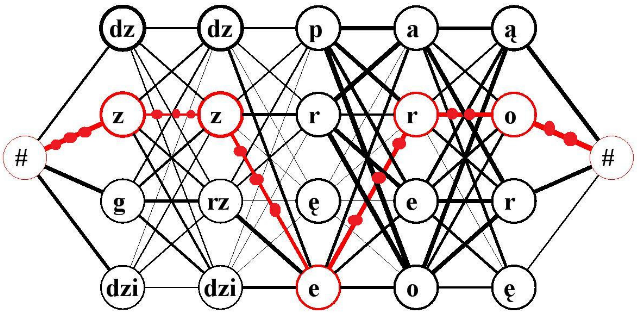


Fig. 1. An example of applying biphone statistics to speech recognition. The graph presents four phone hypotheses for each time slot with different probabilities (the highest row has the highest probabilities). The recognition based on audio information only would be *dz dz p a aę*, which is not close to any Polish word. The best path after including biphone probabilities on edges is *z z e r o*, marked with extra dots. The word *zero* was actually spoken.

II. Algorithm

Sophisticated rules and methods are necessary to obtain the phonetic information from an orthographic text data. Transcription of text into phonetic data was applied by PolPhone [3] using extended SAMPA phonetic alphabet with 39 symbols (plus space) and pronunciation rules for cities Poznań and Kraków. Then the spaces were removed and our own digit symbols corresponding to SAMPA symbols were used, instead of typical ones in the aim of distinguishing phones easier while analysing received phonetic transcriptions. Stream editor (sed) was applied for this tasks.

Afterwards, statistics can be simply collected by counting the number of occurrences of each symbol, pair, and triple in an analysed phonetic transcription, where each character represents a phone. It was conducted using Matlab.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

TABLE I
PHONES IN POLISH (SAMPA [3]), WHERE 1 % CORRESPONDS TO AROUND 11 190 000 OCCURENCES. THE LAST COLUMN PRESENTS THE RESULTS OBTAINED FROM MUCH SMALLER CORPUS A FEW DECADES AGO [6] (N.A. - NOT APLICABLE)

SAMPA	example	transcr.	%	[6]
a	pat	pat	9.584	9.3
e	test	test	9.108	10.2
o	pot	pot	8.994	9.1
t	test	test	4.489	4.4
r	ryk	rIk	4.674	3.6
n	nasz	naS	4.443	4
i	PIT	pit	4.359	3.9
j	jak	jak	3.796	4.5
I	typ	tIp	3.648	4.1
v	wilk	vilk	3.782	3.5
s	syk	sIk	3.638	3
u	puk	puk	3.345	3.4
p	pik	pik	3.263	3.1
m	mysz	mIS	2.988	3.5
k	kit	kit	2.976	2.7
d	dym	dIm	2.888	2.2
l	luk	luk	2.642	2.1
n'	koń	kon'	2.088	2.6
z	zbir	zbir	1.947	1.8
w	łyk	wIk	1.636	2.2
f	fan	fan	1.683	1.5
g	gen	gen	1.547	1.5
t^s	cyk	t^sIk	1.692	1.5
b	bit	bit	1.497	1.5
x	hymn	xImn	1.427	1.1
S	szyk	SIk	1.215	2
s'	świt	s'vit	0.965	1.5
Z	żyto	ZIto	0.944	1.2
t^S	czyn	t^SIn	0.955	1.2
t^s'	ćma	t^s'ma	0.662	1.3
w∼	ciąża	ts'ow∼Za	0.673	0.7
c	kiedy	cjedy	0.698	n.a.
d^z'	dźwig	d^z'vik	0.554	0.8
N	pęk	peNk	0.329	0.8
d^z	dzwoń	d^zvon'	0.261	0.2
J	giełda	Jjewda	0.260	n.a.
z'	źle	z'le	0.195	0.2
j∼	więź	vjej∼s'	0.112	0.1
d^Z	dżem	d^Zem	0.040	0

The necessity of investigating large text corpus pointed to the use of the Polish phonetic transcription system PolPhone [5], [3]. The transcription process is performed by a table-based system, which implements the rules of transcription. Matrix $T \in S^{m \times n}$ is a *transcription table*, where S is a set of strings and the cells meet the requirements listed precisely in [3]. The first element $t_{1,1}$ of each table contains currently processed character of the input string. For every character (or character substring) a table is defined. The first column of each table $\{t_{i,1}\}_{i=1}^{m}$ contains all possible character strings that could precede currently transcribed character. The first row $\{t_{1,j}\}_{j=1}^{n}$ contains all possible character strings that can proceed. All possible phonetic transcription results are stored in the remaining cells $\{t_{i,j}\}_{i=2,j=2}^{m,n}$. A particular element $t_{i,j}$ is applied as a transcription result, if $t_{i,1}$ matches the substring preceding $t_{1,1}$ and $t_{1,j}$ matches the substring proceeding $t_{1,1}$. The longer context is always prefered for transcription, to increase accuracy. Additional tables handle exceptions.

TABLE II
MOST COMMON POLISH BIPHONES. 1% CORRESPONDS TO AROUND 11 190 000 OCCURENCES. THE THIRD COLUMN PROVIDES INFORMATION ON AN INDEX OF A PARTICULAR BIPHONE IN ŁOBACZ AND JASSEM STATISTICS [7]

biphone	%	[7]
je	1.7253	1
ov	1.1829	12
na	1.1632	3
st	1.0791	7
po	1.0479	10
ra	0.9189	14
ro	0.9155	21
on	0.8756	18
n'e	0.8438	2
ta	0.8035	4
va	0.8012	33
ar	0.7545	48
ko	0.7337	25
er	0.7237	44
an	0.6991	20
en	0.6768	27

biphone	%	[7]
ej	0.6620	13
do	0.6459	34
or	0.6413	103
ja	0.6367	5
te	0.6229	9
ne	0.60803	57
em	0.60411	11
at	0.60024	
li	0.58227	68
to	0.58148	8
re	0.5705	92
al	0.5654	35
aw	0.5595	32
no	0.5410	19
od	0.5386	71
ka	0.54	39

TABLE III
THE REST OF MOST COMMON POLISH BIPHONES. 1% CORRESPONDS TO AROUND 11 190 000 OCCURENCES

biphone	%
eg	0.529
n'i	0.526
vy	0.526
av	0.523
go	0.515
ow∼	0.506
ty	0.506
za	0.497
ny	0.493
os	0.489
es	0.489
jo	0.488
ol	0.485
am	0.480
sp	0.473
ma	0.470
pr	0.458
Se	0.456
en'	0.453
le	0.449
an'	0.449
ci	0.447
ot	0.445

biphone	%
ek	0.445
vj	0.442
in	0.434
aj	0.427
pS	0.425
ad	0.423
tu	0.421
op	0.419
as	0.415
ed	0.410
da	0.409
t^se	0.401
mi	0.394
ap	0.388
ez	0.387
nt	0.386
ku	0.383
la	0.383
yx	0.378
ak	0.373
wa	0.370
ru	0.368
mj	0.368

biphone	%
vo	0.366
ep	0.361
ev	0.360
et	0.356
at^s	0.355
el	0.349
ym	0.345
Ze	0.345
ve	0.342
is	0.339
om	0.337
wo	0.333
vi	0.332
de	0.326
n'a	0.326
uv	0.321
az	0.321
ok	0.316
s'e	0.310
mo	0.307
ur	0.305
ob	0.304

The phonetic alphabet used in PolPhone does not differentiate h and χ. However, it does differentiate w∼ and j∼. It can be seen as an unusual phonetical decision, but we are forced to use the existing tool as it is.

Several Rzeczpospolita (Polish daily journal) and Wikipedia articles were used as input data in our experiment. Due to their character, they contain quite many names and places, including foreign ones, what may influence the results slightly. The corpus consists also of several literature books in Polish. Some of them are translations from other languages, so they also contain foreign words. The whole corpus consists of around 267 000 000 words of over 3 000 000 word tokens.

TABLE IV
MOST COMMON POLISH TRIPHONES. 1% CORRESPONDS TO AROUND 11 190 000 OCCURENCES. THE THIRD COLUMN PROVIDES INFORMATION ON AN INDEX OF A PARTICULAR TRIPHONE IN ŁOBACZ AND JASSEM STATISTICS [7]

triphone	%	[7]	triphone	%	[7]
ova	0.3801	10	nyx	0.1673	
ego	0.3655	2	spo	0.1627	96
sta	0.3287	9	an’e	0.1586	16
vje	0.3159	1	pol	0.1538	
pSe	0.2969	6	os’tˆs’	0.1533	138
mje	0.2503	5	jej	0.1514	168
cje	0.2484		tur	0.1448	25
ovy	0.1942		jer	0.1433	86
jon	0.189	79	jow∼	0.143	
ent	0.1842	76	ovj	0.1404	
pro	0.1807	41	ona	0.1381	38
ost	0.1785	19	ist	0.1371	204
ontˆs	0.1749		en’e	0.1354	14
sci	0.1735		sto	0.1347	31
est	0.1734	8	an’a	0.1347	
ana	0.1722	21	ktu	0.1311	
ove	0.1712		ter	0.131	
pra	0.1681	33	s’c’i	0.130	

TABLE V
REST OF THE MOST COMMON POLISH TRIPHONES. 1% CORRESPONDS TO AROUND 11 190 000 OCCURENCES

triphone	%	triphone	%	triphone	%
jeg	0.130	ado	0.108	tra	0.0946
apo	0.130	ont	0.107	n’em	0.0941
nov	0.129	odo	0.106	era	0.0940
epo	0.129	any	0.106	n’ej	0.0938
jed	0.127	ora	0.106	jen’	0.0929
ajo	0.126	ntˆse	0.106	end	0.0928
ast	0.124	ata	0.105	ano	0.0922
tov	0.124	ska	0.104	ejs	0.0922
van	0.123	pot	0.104	stf	0.0920
ina	0.122	neg	0.103	min	0.0912
pov	0.122	ratˆs’	0.103	ami	0.0912
ali	0.122	awa	0.102	nte	0.0909
yst	0.121	oli	0.102	rek	0.0905
pje	0.120	tem	0.102	val	0.0902
ena	0.120	rav	0.1016	n’ik	0.0895
scj	0.120	rov	0.1009	avj	0.0892
pSy	0.118	nej	0.1008	gra	0.0890
dov	0.118	en’a	0.1002	ada	0.0886
ale	0.116	opo	0.0996	atˆs’j	0.0885
ste	0.116	naj	0.0991	sko	0.0884
ovo	0.115	mja	0.0989	an’i	0.0881
van’	0.114	jen	0.0978	eta	0.0881
kon	0.113	ako	0.0976	jez	0.0876
tor	0.112	oku	0.0968	ate	0.0873
kov	0.110	art	0.0965	tan	0.0862
str	0.110	ane	0.0965	ama	0.0852
pod	0.108	rod	0.0964	oje	0.0849
zna	0.108	nym	0.0963	nap	0.0843

III. RESULTS

The frequency of phones is quite similar to the presented in [6] (Table I). The comparison to the other statistics [7] is not simple, because they include a space. It should be mentioned here, that it is an acoustic space, which we could rather call short pause. It means, that it does not appear between all words, but only where a speaker took a breath. A general correlation can be seen, however, the exact order of the statistics differs quite a lot (Tables II and IV).

The total number of around 1 119 000 000 phones were analysed with 39 tokens being specified. Exactly 1 397 biphonetokens (Fig. 2 and Table II) for 1 521 possible combinations were found, which constitutes 91.8%.

38 708 triphone tokens (see Table IV) were detected. The list of the most common triphones is presented in Table IV. With 39 phone tokens there are 59 319 possible triples. It leads to a conclusion that around 65% of possible triples were detected as triphones. It corresponds very well to Young [10], who estimates that in English, 60-70% of possible triples exist as triphones. It allows to make an assumption that all or nearly all triphone tokens were detected in our experiment.

Some values are similar to statistics given by Jassem a few decades ago and reprinted in [1]. We applied computer clusters, so our statistics were calculated for much more data. On the other hand, Jassem’s work was based on manual transcription, while ours uses an automatic method to provide phone transcription, which might be less acurate.

Our results were compared with [7]. The phone statistics are quite similar, because they can be extracted from a small corpus and be representative. Because of this similarity we believe that the grapheme-to-phone automatic method we used has quality close to hand transcriptions done by [7]. The biphone statistics from [7] are less correlated to ours, with triphone ones quite different. We concluded that with correlation on uniphone level, the differences for biphones and triphones are due to much larger corpus which was analysed in our work. Biphone and triphone statistics from [7] are probably quite dependant on transcriptions which were used in their experiment. Our experiment was conducted on much larger corpus so it is much more data independent.

Besides the frequency of triphone occurrence, we are also interested in distributions of their frequencies. These are presented in logarithmic scale in Fig. 3. We have found around 2 200 triphones which occurred once, around 1 200 which occurred twice, and 900 three times. There are quite a lot of triples which occured few times but also there are generally much more triples than for a similar experiment with a space as a phone because there are new triples on words connections. Some threshold can be set and the rarliest triphones can be removed as errors caused by unusual Polish word combinations, acronyms, slang and other variations of dictionary words, onomatopoeic words, foreign words, errors in the process of introducing phone transcriptions and typographical errors in the text corpora.

Zipf’s law states that given some corpus of natural language utterances, the frequency of any word is inversely proportional to its rank in the frequency table. In the case of triphones it seems to be not the case. The changes in frequency between the common triphones are smaller then would be expected from Zipf’s law, while the changes between rare triphones are larger. This type of distribution is very good from practical point of view. It can be estimated that triphones which are in the right part of the Fig. 3, where differences are very sharp are errors.

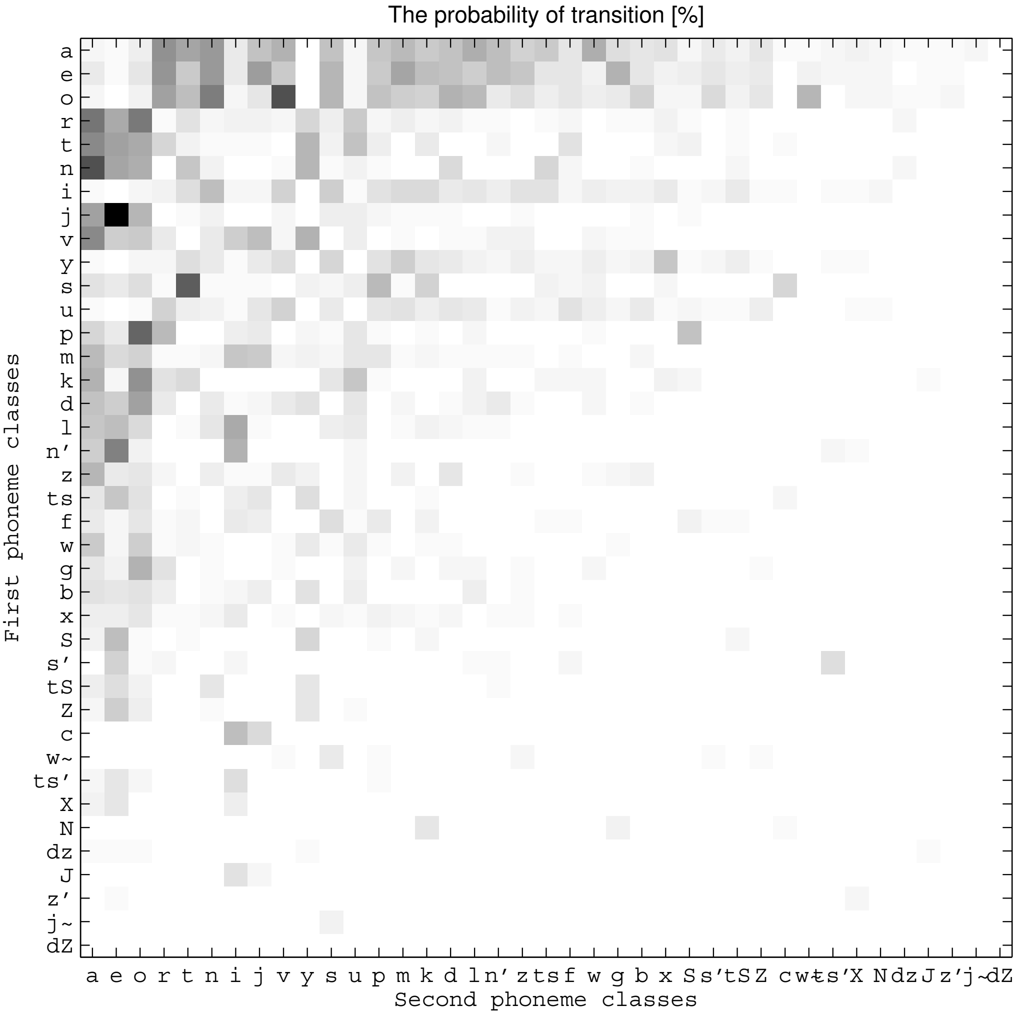


Fig. 2. Frequency of biphones in Polish

Spaces appear between written words but they rarely apear in spoken language. However, pauses can indeed occur between, e.g., longer phrases. Our statistics were collected using phonemic transciption based on written texts. One would say that a better way would be to use transcriptions made directly from audio records. However, in that case the amount of investigated material would be significantly smaller what would have impact on quality of the statistics, especially the triphones.

Entropy

$$H = -\sum_{i=1}^{40} p(i) \log_2 p(i), \qquad (1)$$

where $p(i)$ is a probability of a particular phone, is used as a measure of the disorder of a linguistic system. It describes how many bits in average are needed to describe phones. According to Jassem in [1], the entropy for Polish is 4.7506 bits/phone but including a space. From our calculations, the entropy for phones is 4.7322, for biphones 8.6832 and 12.1987 for triphones. The fact that they do not follow a 1:2:3 proportion is an indication of some level of correlation.

IV. Conclusions

250 000 000 words from different corpora: newspaper articles, Internet and literature were analysed. Statistics of Polish phones, biphones and triphones were created. They are not fully complete, but the corpora were large enough, that they can be successfully applied in speech processing applications. The collected statistics are the largest for Polish, one of the most common Slavic languages, of this type of linguistic computational knowledge. It has several phones different than English and the statistics of phones are also different.

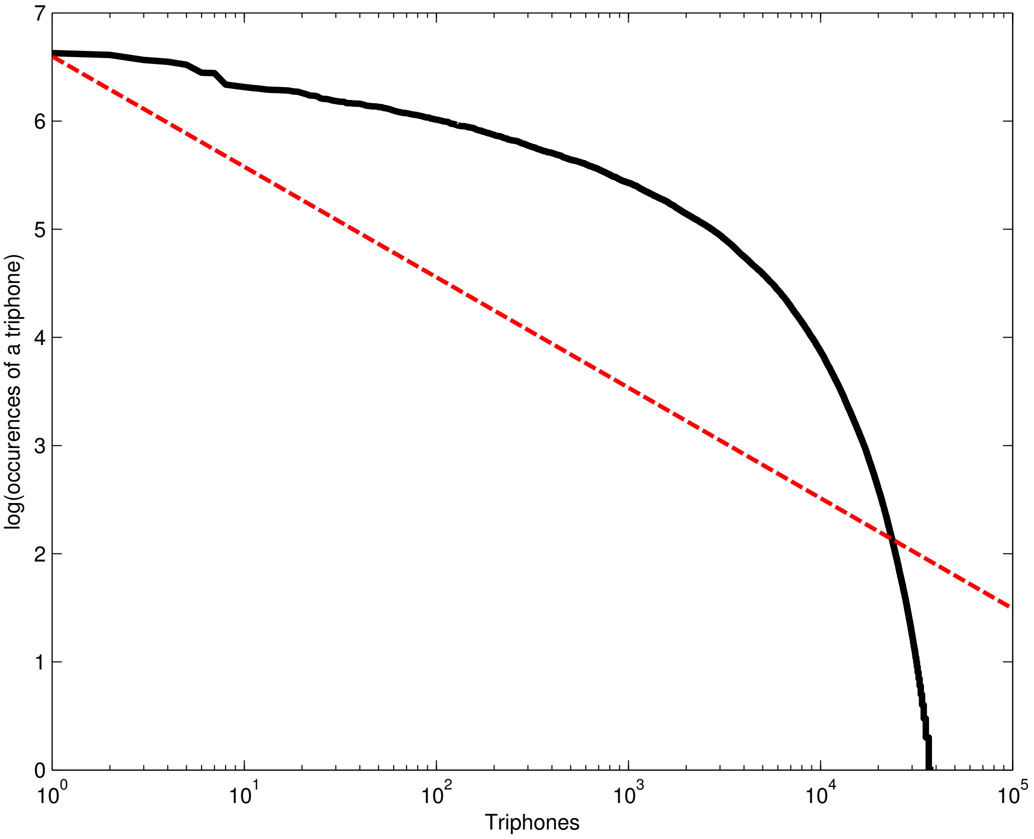


Fig. 3. The solid black line represents phone occurrences distribution while the red, dashed one is an ideal Zipf's law distribution (1/x). It can be estimated that the triphones which fall under the (1/x) Zipf's line are errors rather then real triphones and can be removed from the statistics

V. Acknowledgements

This work was supported by MNISW grant OR00001905. We would like to thank Institute of Linguistics, Adam Mickiewicz University in Poznań for providing PolPhone.

References

[1] C. Basztura, *Rozmawiać z komputerem (Eng. To speak with computers).* Wrocław: Format, 1992.

[2] J. R. Bellegarda, "Large vocabulary speech recognition with multispan statistical language models," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 1, pp. 76–84, 2000.

[3] G. Demenko, M. Wypych, and E. Baranowska, "Implementation of grapheme-to-phoneme rules and extended SAMPA alphabet in Polish text-to-speech synthesis," *Speech and Language Technology, PTFon, Poznań*, vol. 7, no. 17, 2003.

[4] P. B. Denes, "Statistics of spoken English," *The Journal of the Acoustical Society of America*, vol. 34, pp. 1978–1979, 1962.

[5] K. Jassem, "A phonemic transcription and syllable division rule engine," *Onomastica-Copernicus Research Colloquium, Edinburgh*, 1996.

[6] W. Jassem, *Podstawy fonetyki akustycznej (Eng. Rudiments of acoustic phonetics).* Warszawa: Państwowe Wydawnictwo Naukowe, 1973.

[7] P. Łobacz and W. Jassem, "Fonotaktyczna analiza mówionego tekstu polskiego," *B. Rocławski, Wybór materiałów do studiowania fonologii, fonetyki, fonotaktyki i fonostatystyki języka polskiego*, 1979.

[8] L. Rabiner and B. H. Juang, *Fundamentals of speech recognition.* New Jersey: PTR Prentice-Hall, Inc., 1993.

[9] E. J. Yannakoudakis and P. J. Hutton, "An assessment of n-phoneme statistics in phoneme guessing algorithms which aim to incorporate phonotactic constraints," *Speech Communication*, vol. 11, pp. 581 – 602, 1992.

[10] S. Young, "Large vocabulary continuous speech recognition: a review," *IEEE Signal Processing Magazine*, vol. 13(5), pp. 45–57, 1996.

[11] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *HTK Book.* UK: Cambridge University Engineering Department, 2005.

[12] B. Ziółko, J. Gałka, and M. Ziółko, "Phone, diphone and triphone statistics for polish language," *13th International Conference on Speech and Computer SPECOM, St. Petersburg*, 2009.

[13] ——, "Phoneme ngrams based on a polish newspaper corpus," *WORLDCOMP, Las Vegas*, 2009.

[14] ——, "Phonetic statistics from an internet articles corpus of polish language," *International Joint Conference Intelligent Information Systems, Kraków*, 2009.

ISBN 978-83-60810-27-9
ISSN 1896-7094

APyCA: Towards the Automatic Subtitling of Television Content in Spanish

Aitor Álvarez, Arantza del Pozo
Vicomtech Research Centre
Mikeletegi pasealekua, 57
Miramon Teknologia Parkea
20009 Donostia-San Sebastian, Spain
Email: {aalvarez, adelpozo}@vicomtech.org

Andoni Arruti
The University of the Basque Country
Dept. of Computer Architecture and Technology
Manuel de Lardizabal Pasealekua 1
20018 Donostia-San Sebastian, Spain
Email: andoni.arruti@ehu.es

Abstract—**Automatic subtitling of television content has become an approachable challenge due to the advancement of the technology involved. In addition, it has also become a priority need for many Spanish TV broadcasters, who will have to broadcast up to 90% of subtitled content by 2013 to comply with recently approved national audiovisual policies. APyCA, the prototype system described in this paper, has been developed in an attempt to automate the process of subtitling television content in Spanish through the application of state-of-the-art speech and language technologies. Voice activity detection, automatic speech recognition and alignment, discourse segment detection and speaker diarization have proved to be useful to generate time-coded colour-assigned draft transcriptions for post-editing. The productive benefit of the followed approach heavily depends on the performance of the speech recognition module, which achieves reasonable results on clean read speech but degrades as this becomes more noisy and/or spontaneous.**

I. Introduction

SUBTITLING plays an important role in the increasingly multimedia and globalised world we live in. Its usefulness extends from the enrichment of TV content – in order to make it more accessible for people with hearing difficulties or to facilitate audiovisual information retrieval – to its application in noisy environments such as airports and transit stations, where it is not possible to hear TV broadcasts. In addition, subtitling has also become a priority need for many Spanish TV broadcasters, who will have to broadcast up to 90% of subtitled content by 2013 to comply with recently approved national audiovisual policies[1].

However, subtitling is a labor-intensive and economically costly process. As a general rule, manual production of high-quality subtitles can be assumed to take between 8 and 10 times the length of the video material [1]. Nevertheless, mainly due to the higher demands, the time allotted to production of the subtitled material has decreased in recent years [1], [2].

Experienced professionals currently employ dedicated subtitling software tools to help them generate subtitles faster. However, these tools simply display the subtitles on the computer screen as they will appear on the television or movie screen and facilitate purely mechanical functions, such as cueing the subtitles, spell-checking and other basic text processing functions [3]. Only recently speaker-dependent automatic speech recognition has become popular for live subtitling through re-speaking, a technique in which a professional subtitler is trained to dictate live subtitles as the programme happens. Products such as Protitle Live® (NINSIGHT)[2] and WinCAPS® (Sysmedia)[3] allow trained speakers to dictate live subtitles into trained ASR engines. Nevertheless, there is still no ASR-based system in use for fully automated subtitling.

The application of the following state-of-the-art technologies can also contribute to making the subtitling process more automatic and productive:

A. Voice Activity Detection (VAD)

TV content presents a wide range of acoustic conditions: e.g. music, clean speech, outdoor speech, speech with background music, sound effects, noise, etc. However, only those segments that contain speech are to be subtitled. In addition, the different acoustic conditions might require different kinds of processing.

VAD technology can be used to automatically detect the audio segments containing speech. VAD segmentations can also be used to automatically classify and group audio segments with similar acoustic characteristics for further processing.

B. Automatic Speech Recognition (ASR) and alignment

ASR can be employed to obtain automatic transcriptions of the spoken information. Even though ASR can potentially save a lot of time, it is a difficult task mainly due to the high variability of the spoken environments, speakers and speech types present in TV content. Spoken environments vary from clean (studio recordings) to noisy (outdoor recordings, speech mixed with background music or sound effects). The type of speech may differ from dictation (newsreader) to spontaneous (debate or interview). The combination of these

[1] S. Government, “Spanish Audiovisual Law on Subtitles. http://www.cesya.es/es/normativa/legislacion/Financiacion_Radio_TV,” 2008.

[2] http://www.ninsight.fr/FR/

[3] http://www.sysmedia.com/

978-83-60810-27-9/09/$25.00 © 2010 IEEE

possibilities seriously challenges ASR technology, which also needs to deal with speaker independence and the uncontrolled vocabulary of TV programs.

The time-stamps output by the ASR system can also be employed to align the recognised transcripts to the audio signals. In cases where the transcripts already exist, forced alignment can be used instead of recognition to obtain more accurate synchronizations between audio and text.

C. *Discourse segment detection (DSD)*

The detection of entities, relationships or individual events of speech and its segmentation into sentences and phrases is a crucial step for the transition from speech recognition to its full understanding. Unless explicitly dictated, speech recognisers output strings of words without a right segmentation of the output into discursive segments. As a result, ASR transcriptions consist of raw text that is quite difficult to understand for the reader.

DSD techniques can be used to automatically segment ASR transcriptions into segments which contain whole meaning, in order to make them more readable.

D. *Speaker diarization (SD)*

SD is the task of segmenting a multi-speaker audio signal into homogeneous parts and clustering them into different groups, each containing the voice of a single speaker.

In the context of subtitling, SD can be employed to automatically assign a specific color to the subtitles spoken by each speaker.

APyCA, the prototype system described in this paper, integrates the four technologies described above in a unique application, whose aim is to facilitate the manual production of subtitles by experienced professionals, reducing as a result the high cost of subtitle production.

The paper is structured as follows. Section 2 describes the state-of-the-art of the technologies involved and Section 3 presents the resources and tools developed and integrated within the project. Section 4 then describes the implemented prototype. Evaluation of the different modules is presented in Section 5 and finally, Section 6 discusses the main conclusions and further work.

II. State of the art

Much work has been made on the four main technologies involved in APyCA: Voice Activity Detection (VAD), Automatic Speech Recognition (ASR), Discourse Segments Detection (DSD) and Speaker Diarization (SD).

A. *Voice Activity Detection (VAD)*

With increasing demand for voice interfaces, the ability to distinguish human speech from other sounds is becoming crucial. Many works have attempted to discover characteristic features of human voices that are present only in speech. Since such characteristic features have not yet been discovered, short-time energy, zero crossing rate (ZCR), low-variance spectrum (LVS), spectral entropy (SE), periodicity, and so on have been used instead [4], [5]. While it is true that speech has such characteristics, the problem is that they can also be present in some non-speech sounds. This leads to a high false acceptance rate for specific kinds of noise. For example, loud white noise can also have high energy and ZCR.

For these reasons, statistical pattern classification approaches such as Gaussian Mixture Models (GMMs) have gained wider acceptance [6], [7]. In statistical VAD methods, both speech and noise models are trained via corresponding training data. Then, log likelihood ratio tests are applied to input data for speech and noise discrimination. These VAD methods have been shown to exhibit superior performance than the previous approach.

B. *Automatic Speech Recognition (ASR)*

There have been several projects focused on the development of ASR technology for the automatic transcription of Broadcast News (BN). However, most of them were developed for languages other than Spanish, such as English [8], French [9], Portuguese [10] or German [11]. As a result, there is not much data available in Spanish to train a robust speech recogniser for the automatic transcription of broadcast content. Several studies [9], [12] state that at least 100 hours of annotated and transcribed data is required for the adequate training of BN ASR engines and practical development works tend to use as much data as possible. For example, [13] uses up to 1000 hours of training speech data for Persian while [14] employs 81 hours for English, 52 for Portuguese and, in comparison, only 15 for Spanish. This lack of data is the main reason why we decided to use a commercial ASR engine within the APyCA prototype, and to explore adaptation of its default models to improve performance.

Despite the improvement of automatic speech recognisers, developing a system for the automatic transcription of content broadcasted in radio or television is still a challenge for many research groups. A system aimed at the automatic transcription of Portuguese BN, working in a real application scenario currently is [10]. It is based on a hybrid acoustic modelling approach that combines the temporal modeling capabilities of Hidden Markov Models (HMMs) with the pattern discriminative classification capabilities of Multilayer Perceptrons (MLPs). Such acoustic modelling combines phoneme probabilities generated by several MLPs trained on distinct feature sets resulting from different feature extraction processes. The feature extraction methods are PLP, Log-RASTA and MSG. The training of the language model is done using both, Portuguese newspaper texts combined with the transcriptions used for acoustic model training.

With regard to the performance of the ASR systems developed for the different languages on the broadcast domain, the resulting error rates reflect in general the varying level of the acoustic and linguistic complexity of the recordings [11]. WERs range from 16.1% to 64.5%. For Spanish, [14] achieved a mean WER of 18.9%, while [12] managed to decrease it up to 10% by restricting the recognition domain.

The vast majority of the previous studies consider classifying, labeling and structuring the acoustic signal into homogeneous segments essential to optimise the training of the acoustic models of the recogniser, for its subsequent proper operation [15].

C. Discourse Segments Detection (DSD)

The most widely investigated two sources of information to resolve the problem of detecting discursive segment boundaries are word transcriptions (what the speakers say) and prosody (how they say it). It is common to use two statistical models: language and prosodic models. In general, the language model gives the probability of a segmental boundary occurring in a context, while the prosodic model expresses the relationship between prosodic features and segmental boundaries.

Most previous works on discourse segment detection, e.g. [16], are based on the combination of these two information sources. [17] presents a system for punctuation generation which combines both prosodic and linguistic information, in addition to acoustic models. [18] and [19] use a general HMM framework that allows the combination of lexical and prosodic information to recover punctuation marks. A similar approach was used to detect sentence boundaries in [20] and [21].

D. Speaker Diarization (SD)

The varied and wide applicability of speaker diarization technology has led different research groups to develop several systems. The SD process often consists of three main phases: front-end acoustic processing, initial segmentation and final speaker clustering and refinement. The pre-processing step has two main goals. The first one is to normalise the signal in order to remove corrupting noise. In [22], for example, Wiener filtering is applied on each audio channel with that purpose. The second one is to parameterise the signal. Mel Frequency Cepstum Coefficients (MFCC) and Linear Frequency Cepstrum Coefficients (LFCC) are commonly used parameter features, in vectors of several dimensions which often include deltas and/or deltas-deltas.

The initial segmentation phase aims to provide an approximate speaker turn labeling to initialise and speed-up the subsequent segmentation and clustering stages. Several distance criterions can be used in this step. While [23] applies a classical GLR speaker turn detection criteria, [24] uses a segmentation similar to the KL2 metric, measuring the maxima of a local Gaussian divergence between two adjacent sliding windows of five seconds.

The most common clustering method employed in the speaker clustering and refinement phase is the Bayesian Information Criteria (BIC) or a variation called ΔBIC [22, 24]. Initial clusters are generally modelled by single Gaussians with full covariance matrices estimated on the acoustic frames of each segment output by the initial segmentation step. The BIC or ΔBIC metrics are commonly used both, to measure inter-cluster distances and as stop criterions.

III. Resources, tools and applications developed

The main tools integrated in APyCA are: (1) a VAD module; (2) a large vocabulary continuous speech recognition module for recognition and alignment and modules for (3) the detection of discursive segment boundaries and (4) speaker diarization.

A. Voice Activity Detection (VAD)

In order to feed the speech recogniser with audio segments containing speech, a previous segmentation and classification of the audio signal is required. This classification should be as comprehensive as possible, to ensure that no misclassified speech segments are lost.

APyCA segments and classifies the input audio into four different acoustic types: speech, speech plus noise, noise and silence, based on the speech detection functionality of the open source LIUM_SpkDiarization tool [25]. Such segmentation is obtained through Viterbi decoding of one-state Hidden Markov Models (HMMs) trained for the different acoustic conditions on the ESTER broadcast news corpus.

B. Automatic Speech Recognition (ASR) and alignment

APyCA employs the Windows Speech Recogniser (WSR) 8.0 as its ASR engine, integrated through the SAPI 5.3 functionality on the .NET Framework 3.5 environment.

In order to improve its performance, default models have been adapted with acoustically similar (i.e. clean speech vs. noisy speech) and/or TV genre-specific data by feeding the system with the corresponding audio recordings and text transcripts.

As well as for generating textual transcriptions of the spoken information, the ASR module is also used to obtain word-level time-stamps to align the audio and the text. In those cases where transcriptions already exist and the recognition step is not required, audio and text synchronization is computed by an alignment module developed using the HTK Toolkit [26]. The alignment module is a monophone recogniser trained on the Albayzin corpus [27], which extracts 39-dimensional feature vectors containing MFCC, delta and delta-delta coefficients on 25ms windows every 10ms and uses the Spanish version of SAMPA as its phoneme set, plus silence and short pause models. Each monophone (except from the short pause model) consists of non-emitting start and end states plus three emitting states, connected left-to-right with no skips and modelled by a single Gaussian. Viterbi is used for decoding.

C. Discourse Segment Detection (DSD)

Any subtitling platform integrating a speech recognition engine requires the development of algorithms for the automatic segmentation of the recognised output into dircursive segments.

APyCA has four different ways to automatically predict discourse segment boundaries: two of them are related to the acoustic and prosodic processing of the speech signal, another one is based on the linguistic analysis of the transcribed text and the last one combines the previous three approaches. The different techniques employed are presented in more detail in the following sections.

1) DSD based on Acoustic Information

Acoustic pauses are detected by analysing word start and end time-stamps produced by the recogniser or the alignment module during the recognition and alignment processes respectively. Whatever the difference, any non-coincidence in time between the end of a word and the start of the next has been taken as a potential acoustic pause.

It is important to emphasise at this point that even if acoustic pauses do not always correspond to discursive breaks, their relationship is evident in many cases.

2) DSD based on Prosodic Information

Acoustic pauses are not always grammatically correct as they may coincide with breathings, stops or speech difluencies that are not always related to true discursive boundaries. Discourse segment detection based on prosodic information can help resolve this problem.

The implemented algorithm detects discursive segment boundaries based on CART classifiers trained with the Waikato Environment for Knowledge Analysis (WEKA) tool [28]. Three different classes have been used: “silence”, “question” and “nothing” - corresponding to the cases where a word is followed by silence, question mark or nothing, respectively. Each class is trained on prosodic features extracted for each word of the Multext Prosody corpus [29] using the Purdue Prosodic Feature Extraction Tool (PPFE) [30]. 232 prosodic features are extracted around each word. These features are mainly related to:

•***Duration:*** the duration and normalised duration of each word and word boundary are extracted. In addition, the duration and normalised duration of the last vowel and rhyme before a word boundary are also measured.

•***Pitch:*** several different types of F0 features are computed, based on the stylized pitch contour.

○ *Range features:* these include the minimum, maximum, mean, and last F0 values of each word and reflect its pitch range.

○ *Movement features:* measure the movement of the F0 contour within the voiced regions of the words preceding and following a boundary. The minimum, maximum, mean, first and last stylized F0 values of each word are computed and compared to those of the following word, using log differences and ratios.

○ *Slope features:* the last slope value of a word preceding a boundary and the first slope value of a word following a boundary are also calculated.

•***Energy:*** similar to the F0 features, a variety of energy related range features, movement features, and slope features are computed, using various normalization methods.

Each word in the training corpus was manually labeled to belong to one of the three classes defined above.

3) DSD based on Linguistic Information

The linguistic algorithm has the same purpose as the prosodic and acoustic algorithms, i.e. estimating discourse segmentations of the transcribed text. The philosophy used to develop this module has been based on two types of heuristics: grammatical and structural.

On the one hand, a probabilistic part-of-speech (PoS) tagger based on Hidden Markov Models (HMM) has been developed in order to grammatically categorise each word. It has been trained on a proprietary lexical database that includes thousands of grammatical categories of words. In addition, heuristic rules have been developed to detect combinations of grammatical categories, before or after which it is more likely to have segment discourse boundaries.

On the other hand, the most frequent and meaningful structural elements present in the Multext Prosody corpus before or after which it is highly likely to have a discourse segment boundary have been identified. Based on this information, heuristic rules to detect discursive boundaries have been designed.

The latter approach has been found to be more robust than the former one in the automatic subtitling scenario, since recognition errors can lead to grammatical miscategorisations which weaken the designed grammatical heuristic rules.

4) DSD based on Combined Information

The global APyCA DSD system is modular in nature. This means that the input text can be independently segmented into discourse segments using any of the modules designed: acoustic, prosodic and/or linguistic.

A combined model has also been developed, which takes the predictions and confidence measures provided by the three modules described above and gives a final result based on their weighted combination. In general, if two modules detect a pause in a word boundary with enough confidence, we take it as a real pause. This approach exploits the complementarity of the three very different sources of information used for the detection of discourse segments.

D. Speaker Diarization (SD)

This module aims at segmenting the acoustic signal according to the speaker identities, so that each speaker can be assigned a different subtitle colour.

APyCA uses the LIUM_SpkDiarization open source tool [25] to solve the task of speaker diarization. Signals are parameterised using 13 Mel Frequency Cepstral Coefficients (MFCC) including coefficient C0 as energy, computed with the Sphinx 4 tools [31]. 20 ms windows are employed with an overlap of 10 ms. Cepstral Mean Normalisation (CMN) is not applied, due to its tendency to increase the error rate of the diarization task.

The diarization process consists of three main phases. Instantatenous signal change points corresponding to segment boundaries are detected first, using distance-based segmentation metrics which combine the Generalised Likelihood Ratio (GLR) and Bayesian Information Criterion (BIC). GLR is computed using full covariance Gaussians estimated on sliding windows of five seconds and followed by a second ΔBIC pass, which also uses full covariance Gaussians, to fuse consecutive segments of the same speaker.

Then, nonadjacent segments of the same nature and speaker are brought together in clusters using a hierarchical agglomerative clustering algorithm with a stopping criterion based on the ΔBIC metric.

Finally, Viterbi decoding is performed to generate improved segmentations. Each cluster is modeled by a one-state HMM, represented by a GMM with 8 components and a diagonal covariance matrix learned by Expectation-Maximization Maximum-Likelihood (EM-ML) over the segments of the cluster. The log-penalty between two HMMs is fixed experimentally.

IV. Integration of Components into a Demo Application. Description of the Prototype

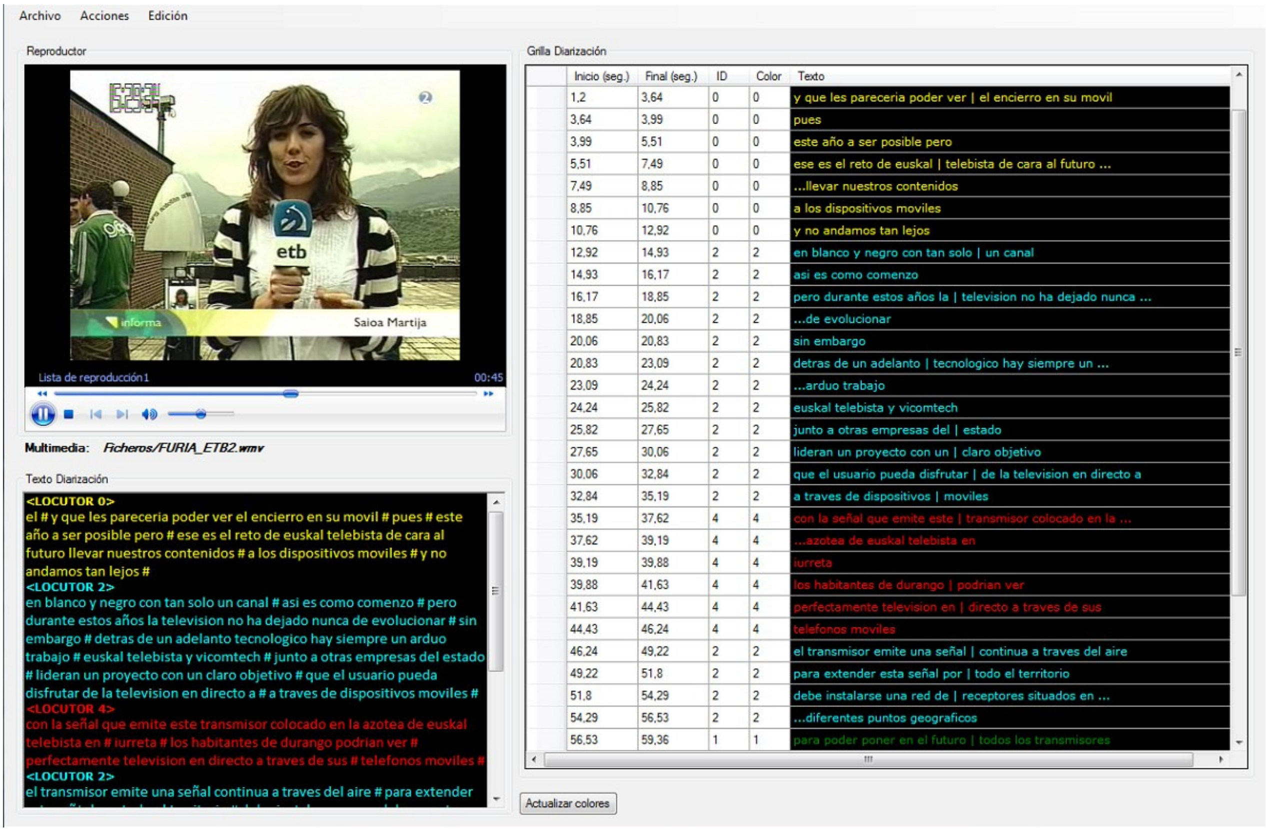
APyCA is a prototype oriented towards the automatic transcription of TV content in Spanish which integrates the technologies described in the previous sections of the paper and aims to serve the professional subtitlers as a tool to facilitate the creation and editing of subtitles. The following sections describe the main features and architecture of the developed demo application.

A. Features

Its input is TV content in the form of video or audio. It supports many different formats, including the main standards used by television producers, such as *mpeg2, h.264, aac or wav.*

Its output is a well-formed STL (binary) or SRT subtitle file, which respects the maximum number of characters allowed per line and includes colours to differentiate speakers. Time-spotting is based on the estimated word time-stamps and discourse segment boundaries. If needed, these subtitle files can be easily edited further using commercial software for subtitle generation, e.g. WinCAPS, FAB Teletext and Subtitling, Subtitle Workshop, etc.

The technologies involved have been grouped into three automatic functionalities: transcription, time-spotting and



Multimedia: *Ficheros/FURIA_ETB2.wmv*

Texto Diarización

<LOCUTOR 0>
el # y que les pareceria poder ver el encierro en su movil # pues # este año a ser posible pero # ese es el reto de euskal telebista de cara al futuro llevar nuestros contenidos # a los dispositivos moviles # y no andamos tan lejos #
<LOCUTOR 2>
en blanco y negro con tan solo un canal # asi es como comenzo # pero durante estos años la television no ha dejado nunca de evolucionar # sin embargo # detras de un adelanto tecnologico hay siempre un arduo trabajo # euskal telebista y vicomtech # junto a otras empresas del estado # lideran un proyecto con un claro objetivo # que el usuario pueda disfrutar de la television en directo a # a traves de dispositivos moviles #
<LOCUTOR 4>
con la señal que emite este transmisor colocado en la azotea de euskal telebista en # iurreta # los habitantes de durango podrian ver # perfectamente television en directo a traves de sus # telefonos moviles #
<LOCUTOR 2>
el transmisor emite una señal continua a traves del aire # para extender

Inicio (seg.)	Final (seg.)	ID	Color	Texto
1,2	3,64	0	0	y que les pareceria poder ver | el encierro en su movil
3,64	3,99	0	0	pues
3,99	5,51	0	0	este año a ser posible pero
5,51	7,49	0	0	ese es el reto de euskal | telebista de cara al futuro ...
7,49	8,85	0	0	...llevar nuestros contenidos
8,85	10,76	0	0	a los dispositivos moviles
10,76	12,92	0	0	y no andamos tan lejos
12,92	14,93	2	2	en blanco y negro con tan solo | un canal
14,93	16,17	2	2	asi es como comenzo
16,17	18,85	2	2	pero durante estos años la | television no ha dejado nunca ...
18,85	20,06	2	2	...de evolucionar
20,06	20,83	2	2	sin embargo
20,83	23,09	2	2	detras de un adelanto | tecnologico hay siempre un ...
23,09	24,24	2	2	...arduo trabajo
24,24	25,82	2	2	euskal telebista y vicomtech
25,82	27,65	2	2	junto a otras empresas del | estado
27,65	30,06	2	2	lideran un proyecto con un | claro objetivo
30,06	32,84	2	2	que el usuario pueda disfrutar | de la television en directo a
32,84	35,19	2	2	a traves de dispositivos | moviles
35,19	37,62	4	4	con la señal que emite este | transmisor colocado en la ...
37,62	39,19	4	4	...azotea de euskal telebista en
39,19	39,88	4	4	iurreta
39,88	41,63	4	4	los habitantes de durango | podrian ver
41,63	44,43	4	4	perfectamente television en | directo a traves de sus
44,43	46,24	4	4	telefonos moviles
46,24	49,22	2	2	el transmisor emite una señal | continua a traves del aire
49,22	51,8	2	2	para extender esta señal por | todo el territorio
51,8	54,29	2	2	debe instalarse una red de | receptores situados en ...
54,29	56,53	2	2	...diferentes puntos geograficos
56,53	59,36	1	1	para poder poner en el futuro | todos los transmisores

Fig 1: System screen capture

speaker diarization – which can be applied and edited independently through dedicated graphical user interfaces.

•The **Automatic Transcription** screen shows the raw text returned by the ASR engine for those segments that contain speech. It also allows playback and editing of the transcriptions, so that subtitlers can manually correct the errors of the recogniser. The fewer the transcription mistakes, the better the time-spotting will be.

•The **Automatic Time-Spotting** functionality chunks the transcribed text into discoursive segments and aligns them with the audio. The start times, end times and text of each subtitle can be edited manually.

•The **Speaker Diarization** screen automatically assigns different colours to the subtitles spoken by different speakers, also allowing their manual edition.

These three functionalities can be combined to suit the needs of the professional subtitlers. It is possible, for example, to skip the automatic transcription step and upload already transcribed audiovisual content. Or to generate the subtitle files without speaker diarization information.

The prototype has been developed entirely using the Microsoft .NET platform, the C# programming language and several Perl scripts for text processing.

A screen capture of the system is shown in Fig. 1.

B. Architecture

Fig. 2 illustrates how the different modules interact within the system and with the user.

The system supports the input of TV content in video or audio formats, as well as with or without its corresponding textual transcription. The FFmpeg [32] tool is used to extract the audio from the video in different formats and configurations. If the transcription does not exist, the audio will be recognised. If the transcription exists, forced alignment will be applied instead to obtain word time-stamps. In any case, time-stamps are required for the discourse segment detection and speaker diarization modules. Output subtitle files can be generated after recognition/alignment, after discourse segment detection or after speaker diarization.

The modular architecture of the system will allow simple integration of additional modules providing new functionalities in the future.

V. Evaluation

A. Data

The main modules of the APyCA prototype have been evaluated individually on two different genres of TV content: weather forecasts and political interviews.

Weather forecasts do not present difficulties related to the the spoken environment, the type of speech used, the number of speakers involved or the quantity and quality of their interventions. In fact, they contain just one anchor presenter following a previously written and rehearsed script in a noise-free recording studio. However, the employed vocabulary is very specific of the meteorological domain. Political interviews involve many different types of spoken environments (studio, parliament, street), types of speakers and speech (presenters following pre-prepared scripts and/or spontaneous interviewees) and a very domain specific vocabulary, with many mentions to names of politicians.

Ten programs of each type were recorded and used for training (8) and testing (2) the different modules of the system. Their reference transcripts were obtained by manually correcting the transcriptions output by the WSR 8.0 with default models.

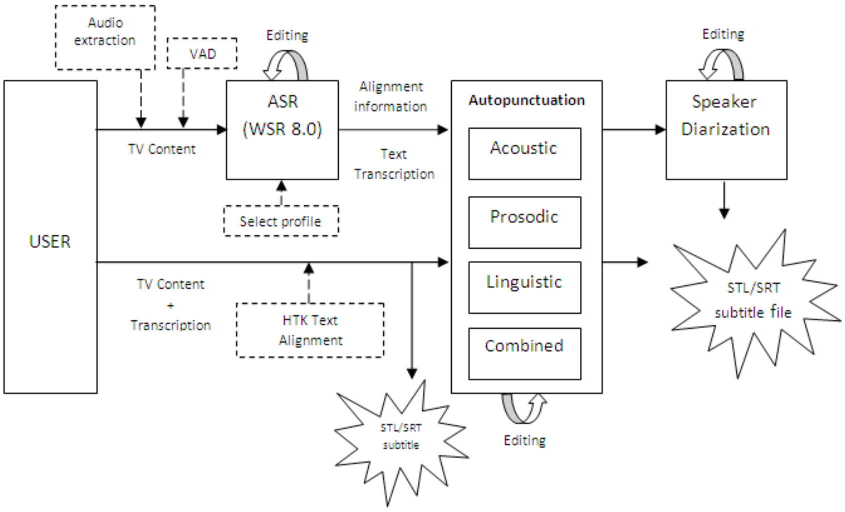


Fig 2: System architecture

B. Automatic Speech Recognition (ASR)

The performance of the WSR 8.0 recogniser was tested in three different conditions: (i) using the default recogniser models, (ii) using clean and noisy speech profiles adapted to the clean and noisy acoustic conditions found in each corpus, and (iii) using TV genre-specific profiles trained for the weather forecast and political interview domains.

Results for *the weather forecast* corpus are shown in Table 1. The average percentage of words correctly recognised overall is especially promising. The column labeled *Baseline* shows the performance of the default profile of the commercial WSR 8.0 engine. The column labeled *TV-genre profile* corresponds to the recognition rate achieved using a profile trained with all the training content of the weather forecast corpus. The *Acoustic profiles* column shows the results obtained after applying the recognition profiles trained with clean and noisy speech. Contrary to expectations, acoustic profiling does not achieve the best results probably due to the loss of context caused by the more detailed audio segmentation involved.

Table I.
Average recognition rate in the weather forecast corpus

Baseline	*TV-genre profile*	*Acoustic profiles*
81.3 %	96.65 %	92.34 %

Results for the *political interview* corpus are shown in Table 2. Less satisfactory recognition rates were obtained with this corpus overall, due to the inherent difficulty of the content type. It is remarkable that the application of acoustic profiling did not improve the results obtained by the default recognition profiles. On the other hand, TV-genre profiling only improves baseline results slightly.

Table II.
Average recognition rate in the political interview corpus

Baseline	*TV-genre profile*	*Acoustic profiles*
79.54 %	79.80 %	78.60 %

C. Discourse segment detection (DSD)

Results concerning the evaluation of the different DSD modules are shown in Tables III, IV and V.

Each module was evaluated individually, against manually labeled reference test files. Acoustic labels take into account acoustic silences and short pauses. Prosodic labels are based on the intonation of the related sound files. Linguistic labels consider syntactic information of the associated text.

According to the followed evaluation methodology, "*Matching breaks*" refers to the percentage of breaks that match the reference file, while "*Unassigned breaks*" relates to the percentage of breaks present in the reference labels which have not been assigned by the different modules. The percentage of extra breaks assigned by the DSD modules that do not appear in the reference files is counted as "*Extra breaks*".

Table III.
Results of the acoustic module

Matching breaks	*Unassigned breaks*	*Extra breaks*
92.67 %	7.33 %	16.60 %

Table IV.
Results of the prosodic module

Matching breaks	*Unassigned breaks*	*Extra breaks*
64.49 %	35.51 %	63.64 %

Table V.
Results of the linguistic module

Matching breaks	*Unassigned breaks*	*Extra breaks*
51.92 %	48.08%	1.44 %

Results show that in 92.67% of the cases, acoustic segmental boundaries were assigned correctly, 7.33% of the acoustic pauses were not detected and 16.80% were wrongly assigned, particularly those matching breathing stops and speech disfluencies. Spontaneous speech was the main enemy of the prosodic module, mainly trained under a database of read speech. Nevertheless, it achieved a non negligible 64.49% accuracy rate. As for the linguistic module, its performance was penalised by the recognition errors which affect the designed heuristic rules. Overall, the acoustic module has proved to be the most efficient to detect discourse segment boundaries, due to its high speed and hit rate.

D. Speaker Diarization (SD)

The speaker diarization module achieved very good performance. Even in the rich acoustic environment of the political interview corpus, results achieved 87% success rate. Errors were mainly due to background acoustic changes, which caused the same speaker to be classified as two in some cases where the background acoustic environment was different, since the BIC criterion employed in APyCA for speaker diarization was actually designed to classify those segments as different.

VI. Conclusions and further work

Voice activity detection, automatic speech recognition and alignment, discourse segments detection and speaker diarization technologies have been developed, customized and integrated in a prototype to support the subtitle generation process of Spanish TV content.

Objective evalutation of the different modules has shown that the proposed approach is feasible and applicable to generate automatically time-coded and colour-assigned draft transcriptions for post-editing. The commercial WSR 8.0 engine has shown adequate performance for the task. Adaptation of the default profiles to each TV-genre has shown to improve recognition accuracy. However, transcription performance degrades overall as the input speech becomes more noisy and/or spontaneous. Acoustic discourse segment detection has been found to be very efficient in terms of high speed and hit rate for time-spotting. The LIUM_SpkDiarization tool has also shown good results in the colour assignment task.

However, there is still quite a lot of room for improvement. Techniques to enhance ASR accuracy in noisy and/or spontaneous environments could be integrated. The prosodic DSD module could be trained on spontaneous and/or emotional speech corpora to better match intonation patterns of certain TV content. Finally, feature normalization techniques could be added to the speaker diarization module to obtain one-to-one relationships between clusters and speakers. Although some positive informal usability tests have been done with professional subtitlers, a more comprehensive assesment should be carried out in order to verify the feasibility and quantify the time and money savings which could be provided by a software tool similar to the developed prototype.

Acknowledgment

This work has been partially funded by the Basque Government. The authors would like to thank Mixer, Irusoin and ETB for providing the corpora and giving usability feedback.

References

[1] M. Flanagan, “Human Evaluation of Example-Based MT of subtitles for DVD,” Dublin City University, 2009.

[2] M. Carroll, “Subtitling: Changing standards for new media? LISA Newsletter Global Insider, XIII, 3.5. 2004. http://www.lisa.org/globalizationinsider/2004/09/subtitling_chan.htm,”

[3] L. Bowker, Computer-aided Translation Technology: A Practical Introduction, Ottawa: University of Ottawa Press, 2002.

[4] J.L. Shen, J.W. Hung, and L.S. Lee, “Robust Entropy-Based Endpoint Detection for Speech Recognition in Noisy Environments”, Proc. Int. Conf. Spoken Language Process., paper 0232, 1998.

[5] I.D. Lee, H.P. Stern, S.A. Mahmoud, “A Voice Activity Detection Algorithm for Communication Systems with Dynamically Varying Background Acoustic Noises,” Proc. Veh. Technol. Conf., 1998.

[6] J. Sohn, N.S. Kim, and W. Sung, “A Statistical Model-Based Voice Activity Detection”, IEEE Signal Process. Lett., vol. 6, no. 1, pp. 1-3, 1999.

[7] A. Davis, S. Nordholm, R. Togneri, “Statistical Voice Activity Detection Using Low-Variance Spectrum Estimation and an Adaptive Threshold” , IEEE Trans. on Signal Proc., vol 14, no 2, pp. 412-424, 2006.

[8] J.S. Garofolo, J.G. Fiscus, W.M. Fisher, “Design and preparation of the 1996 hub-4 broadcast news benchmark test corpora,” in Proceedings of the DARPA Speech Recognition Workshop., pp. 15–21, 1997.

[9] S. Galliano, E. Geoffrois, G. Gravier, J.F. Bonastre, D. Mostefa, K. Choukri. "Corpus description of the ESTER Evaluation Campaign for the Rich Transcription of French Broadcast News". In Proceedings of the 5th International Conference on Language Resources and Evaluation 2006.

[10] H. Meinedo, D. Caseiro, J. Neto, I. Trancoso. “AUDIMUS.MEDIA: a broadcast news speech recognition system for the European Portuguese language”. In Proceedings of PROPOR 2003, Portugal, 2003.

[11] D. Baum, B. Samlowski, T. Winkler, R. Bardeli, Schneider: “DiSCo - a speaker and speech recognition evaluation corpus for challenging problems in the broadcast domain”. Proceedings of the GSCL Symposium 'Sprachtechnologie und eHumanities' 2009.

[12] J. Loof, Ch. Gollan, S. Hahn, G. Heigold, B. Hoffmeister, Ch. Plahl, D. Rybach R. Schluter and H. Ney. "The RWTH 2007 TC-STAR Evaluation System for European English and Spanish". Interspech 2007.

[13] C. Gollan, H. Ney, “Towards automatic learning in LVCSR: Rapid development of a Persian broadcast transcription system,” Interspeech' 08.

[14] F. Batista, I. Trancoso, N. J. Mamede. "Comparing Automatic Rich Transcription for Portuguese, Spanish and English Broadcast News". In Automatic Speech Recognition and Understanding Workshop, 2009.

[15] J.-L. Gauvain, L. Lamel, C. Barras, G. Adda, and Y. de Kercadio, “The Limsi SDR systemfor TREC-9,” in Proc. 9th Text Retrieval Conference, TREC-9, pp. 335–341, Gaithersburg, Md, USA, 2000.

[16] Y. Liu, E. Shriberg, A. Stolcke, D. Hillard, M. Ostendorf, B. Peskin, and M. Harper. “The ICSI-SRI-UW Metadata Extraction System”. ICSLP 2004, International Conf. on Spoken Language Processing, Korea. 2004.

[17] J.H. Yim. “Named Entity Recognition from Speech and Its Use in the Generation of Enhanced Speech Recognition Output”. Darwin College, University of Cambridge and Cambridge University Engineering Department. 2001.

[18] H. Christensen, Y. Gotoh, and S. Renals, “Punctuation annotation using statistical prosody models,” in Proc. of the ISCA Workshop on Prosody in Speech Recognition and Understanding, pp. 35–40, 2001.

[19] J. Kim, P. C. Woodland, “The use of prosody in a combined system for punctuation generation and speech recognition,” Proc. Eurospeech' 01.

[20] Y. Gotoh and S. Renals, “Sentence boundary detection in broadcast speech transcripts,” in Proc. of the ISCA Workshop: ASR-2000.

[21] E. Shriberg, A. Stolcke, D. Hakkani-Tür, and G. Tür, “Prosody based automatic segmentation of speech into sentences and topics,” Speech Communications, vol. 32, no. 1-2, pp. 127–154, 2000.

[22] T. L. Nwe, H. Sun, H. Li, S. Rahardja, "Speaker Diarization in Meeting Audio", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, April 19-24, 2009.

[23] J. Huang, E. Marcheret, K. Visewswariah, G. Potamianos, “The IBM RT07 Evaluation Systems for Speaker Diarization on Lecture Meetings”, in Multimodal Technologies for Perception of Humans, Springer, 2008.

[24] C.Wooters, M. Huijbregts. “The ICSI RT07s Speaker Diarization System”. In Rich Transcription 2007 Meeting Recognition Workshop.

[25] S. Meignier, T. Merlin. “LIUM_SpkDiarization: An Open Source Toolkit For Diarization”. CMU Sphinx Workshop 2010, Dallas, 2010.

[26] Hidden Markov Model Toolkit (HTK) 3.2, Cambridge University Engineering Department. http://htk.eng.cam.ac.uk/, 2002.

[27] F. Casacuberta, R. Garcia, J. Llisterri, C. Nadeu, J.M. Pardo, A. Rubio: “Development of Spanish Corpora for Speech Research (Albayzin)”. Workshop on International Cooperation and Standarization of Speech Databases and Speech I/O Assesment Methods, Italy, 199.1

[28] 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.

[29] E. Campione, (Ed.) Multext-Prosody. A multilingual prosodic database. CD-ROM Distributed by ELRA/ELDA. 1999.

[30] Z. Huang, L. Chen, M. Harper. “Purdue Prosodic Feature Extraction Toolkit on Praat”. Spoken Language Processing Lab, Purdue University. 2006.

[31] Sphinx-4. “A speech recognizer written entirely in the Java programming language”. http://cmusphinx.sourceforge.net/sphinx4/

[32] FFmpeg. “A complete, cross-platform solution to record, convert and stream audio and video”. http://www.ffmpeg.org/

10th International Multidisciplinary Conference on e-Commerce and e-Government

We would like to invite original papers concerning all aspects of electronic commerce, electronic government and related issues. The conference is meant to be of interest to the academic community as well as representatives of business, industry, government, NGOs and the information technology sector. Thus – apart from research papers – practical presentations of existing solutions are also invited.

The areas of e-Commerce, e-Governance, e-Government etc. are interdisciplinary by nature: they involve people whose background is in economy, management, artificial intelligence, computer science, sociology, psychology, law etc. We feel that a meeting of specialists in those areas may help to cross the boundaries between the traditional disciplines, but also between the communities of theorists and practitioners. This can also create a better understanding of mechanisms underlying electronic markets, electronic administrations and networked organizations.

The general list of topics includes:

- electronic markets and electronic marketing
- business models and processes in e-Commerce and e-Government
- languages and models for e-Commerce and e-Government
- Artificial Intelligence in e-Commerce and e-Government
- electronic contracting and public procurement
- legal aspects of e-Commerce and e-Government
- electronic interaction and negotiation
- Virtual Enterprises and Knowledge Management
- technology for e-Commerce and e-Government
- Internet computing, networked enterprises and networked governments
- social aspects of e-Commerce and e-Government
- futurology of e-Commerce and e-Government
- e-Inclusion and its influence on e-Commerce and e-Government
- national and cross-boarder e-Government services
- productivity and efficiency in e-Commerce and e-Government

Program Committee

Hofreiter Birgit, University of Liechtenstein, Liechtenstein

Junaid Ahsenali Chaudhry, University of Hail, Saudi Arabia

Jen-Yao Chung, IBM, USA

Fausto Fasano, University of Molise, Italy

Francisco Flores, University of Huelva, Spain

Marianne Hickey, Hewlett Packard, United Kingdom

Natascha Hoebel, Goethe University, Germany

Christian Huemer, TU Vienna, Austria

Wojtek Jamroga, University of Luxembourg, Luxembourg

Pawel J. Kalczynski, California State University, USA

Jürgen Karla, RWTH Aachen University, Germany

Agnes Koschmider, Karlsruhe Institut of Technology, Germany

Franz Lehner, University of Passau, Germany

Yinsheng Li, Fudan University, China

Kwei-Jay Lin, UC Irvine, USA

Claudia Linnhoff-Popien, Ludwig-Maximilians-Universitaet Muenchen, Germany

Antonio G. Lopez-Herrera, University of Granada, Spain

Giuseppe Lugano, University of Jyväskylä, Finland

Yuxin Mao, Zhejiang GongShang University, China

Vincent Ng, The Hong-Kong Polytechnic University, HONG-KONG

Gang Pan, Zhejiang University, China

Zeljko Panian, University of Zagreb, CROATIA

Elvira Popescu, University of Craiova, Romania

Roy Rada, The University of Maryland, Baltimore County, USA

Andrew Ravenscroft, London Metropolitan University, United Kingdom

Chunming Rong, University of Stavanger, Norway

Jeanne Schreurs, Hasselt University, Belgium

Tim Scott, SearchFit Inc., USA

Murali Subbarao, Billeo Inc., USA

Urszula Swierczynska-Kaczor, Jan Kochanowski University in Kielce, Poland

Rana Tassabehji, University of Bradford, United Kingdom

Arthur Tatnall, Victoria University, Australia

Mauro Tortonesi, University of Ferrara, Italy

Jacek Wachowicz, Gdańsk University of Technology, Poland

Rolf Wigand, University of Arkansas at Little Rock, USA

Zhong Yuansheng, China

Organizing Committee

Jacek Wachowicz (Chairman), Gdańsk University of Technology, Poland

ISBN 978-83-60810-27-9
ISSN 1896-7094

Trusted Data in IBM's MDM: Accuracy Dimension

(Work in progress)

Przemyslaw Pawluk
York University
Toronto ON, Canada
Center of Advanced Studies
IBM, Toronto
Email: pawluk@cse.yorku.ca

***Abstract*—A good data model designed for e-Commerce or e-Government has little value if it lacks accurate, up-to-date data [22]. In this paper data quality measures, its processing and maintenance in *IBM InfoSphere MDM Server* and *IBM InfoSphere Information Server* is described. We also introduce a notion of *trust*, which extends the concept of data quality and allows businesses to consider additional factors, that can influence the decision making process. In the solutions presented here, we would like to utilize existing tools provided by IBM in an innovative way and provide new data structures and algorithms for calculating scores for persistent and transient quality and trust factors.**

I. Introduction

MANY organizations have come to the realization that they do not have an accurate view of their business-critical information such as customers, vendors, accounts, or products. As new enterprise systems are added, silos are created resulting in overlap and inconsistency of information. This varied collection of systems can be the result of systems introduced through mergers and acquisitions, purchase of packaged applications for enterprise resource planning (ERP) or customer relationship management (CRM), different variant and versions of the same application used for different lines of business or home grown applications. Data in these systems typically differs both in structure and in content. Some data might be incorrect, some of it might just be old, and some other parts of it might show different aspects of the same entity (for example, a home vs. a work address for a customer).

Master Data Management (MDM) is an approach that decouples master information from the applications that created it and pulls it together to provide a single, unified view across business processes, transactional and analytical systems. Master data is not about all of the data of an organization. It is the data that deals with the core facts about the key entities of a business: customers, accounts, locations and products. Master data is high value data that is commonly shared across an enterprise – within or across the lines of business. MDM applications, such as IBM's InfoSphere Master Data Management Server, contain functionality to maintain master data by addressing key data issues such as governance, quality and consistency. They maintain and leverage relationships between master data entities and manage the complete lifecycle of the data and support multiple implementation approaches. MDM system itself is designed to support enterprise in the master data processing, integration and analysis.

As very important, quality of master data requires special attention. Different aspects or dimensions of quality need to be considered and maintained in all processes of the enterprise. *Trust scores*, introduced by this paper, can provide important information to the decision makers. Our approach to the quality of data is slightly different than described so far in the literature [4], [6], [9]. Our goal is to provide the user with the estimates of data quality and trust. Trust in this case is the aggregated value of multiple factors, and is intended to cover quality and non-quality aspects of master data. We are not making the attempts to build fixes nor enforce any quality policy. The information provided by us is intend to identify weaknesses of data quality or trustworthiness. The data quality enforcement should be then improved based on this information.

This paper focuses on the creation of measures, or trust factors, that serve to determine the trustworthiness of data being managed by MDM applications, specifically those being introduced in IBM's InfoSphere MDM Server. We would like to define here the model for data quality and methods of their processing in MDM. This new notion involves creating *trust scores* for trust factors that enhance the notion of data quality and the more broad quality-unrelated features such as lineage, security, and stewardship. All these have one goal – to support businesses in the decision making process, or data stewardship by providing information about different aspects of data. We would like to be more focus in this work on the quality aspects of data, especially the accuracy, which is one of the most commonly used quality dimension in the literature [2], [4], [12], [16], [20], [21], [26], [29], [31].

This paper is organized as follows. Section II presents the underpinning principles of Master Data Management (MDM), related concepts as well as the tools we used to prepare the trust scoring prototype. Section III provides a short overview of data quality and introduces the notion of trust. Section IV presents structures and methods used to acquire and store information about one of the most commonly used qualoity factor – accuracy. Section V describes our approach to accuracy estimation in SQL query processing.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

II. MDM AND INFORMATION SERVER

Master data management is a relatively fast growing software market. Many customer acknowledge they have data quality and data governance problems and look for solutions to these problems. Crucial parts of such MDM solutions are data quality and data trust mechanisms [8], [10], [23]. In this section we present the MDM environment and the comprehensive approach to data trust and data quality that utilizes tools provided by IBM.

A. Definitions

Master Data Management (MDM) provides the technology and processes to manage Master Data in an organization. Master Data is the data an organization stores about key elements or business entities that define its operation. An MDM solution enables an enterprise to govern, create, maintain, use, and analyze consistent, complete, contextual, and accurate master data information for all stakeholders. Master data is typically high value information that an organization uses repeatedly across many business processes. For these to operate efficiently, this master data must be accurate and consistent to ensure correct business decisions. Unfortunately in many organizations, master data is fragmented across many applications, with many inconsistent copies and no plan to improve the situation.

Master Data Management (MDM) products differs from traditional approaches to the masted data, that include the use of existing enterprise applications, data warehouses and even middleware. It *is not* domain-centric approach such as CRM application for the customer domain or ERP application for the product domain. Some MDM products decouple data linked to source systems so they can dynamically create a virtual view of the domain, while others include the additional ability to physically store master data and persist and propagate this information. Some products are not designed for a specific usage style, while others provide a single usage of this master data. Even more advanced products provide all of the usage types required in today's complex business-collaborative, operational and analytic-as out-of-the-box functionality. These products also provide intelligent data management by recognizing changes in the information and triggering additional processes as necessary. Finally, MDM products vary in their domain coverage, ranging from specializing in a single domain such as customer or product to spanning multiple and integrated domains. Those that span multiple domains help to harness not only the value of the domain, but also the value between domains, also known as relationships. Relationships may include customers to their locations, to their accounts or to products they have purchased. This combination of multiple domains, multiple usage styles and the full set of capabilities between creating a virtual view and performance in a transactional environment is known as multiform master data management.

Achieving a high level of data quality is key prerequisite for many of the MDM objectives. Without high quality data the best analytics and business intelligence applications are still going to deliver unreliable input to important business decisions. Another key aspect of the management of the master data is achieving a high level of trustworthiness in the data. It is a key factor for customers to have reliable information about the data. Information about the quality, the origin, the timeliness and many other factors influence the business decisions based on the provided data.

The introduction of *data governance* in the organization is a vital prerequisite to come to more trusted information. Moving to master data management can be the cornerstone of a data governance program. It is important however to note that at the same time, moving to MDM cannot be successful without data governance.

Data governance is defined as "the orchestration of people, process and technology to enable an organization to leverage information as an enterprise asset" [15]. It manages, safeguards, improves and protects organizational information. The effectiveness of data governance can influence the quality, availability and integrity of data by enabling cross-organizational collaboration and structured policy-making.

B. MDM Tools

We will present here some tools provided by IBM that are very useful in terms of quality assessment and management. All of them are parts of the IBM InfoSphere platform.

IBM InfoSphere MDM Server is an application that was built on open standards and the Java Enterprise Edition(JEE) platform. It is a real-time transactional application with a service-oriented architecture that has been built to be scalable from both volume and performance perspectives. Shipping with a persistent relational store, it provides a set of predefined entities supporting the storage of master data applicable to each of the product's predefined domains. It also includes the MDM Workbench – an integrated set of Eclipse plug-ins to IBM Rational Software Architect/Developer that support the creation of new MDM entities and accompanying services, and a variety of extensions to MDM entities. *IBM InfoSphere Information Analyzer* that profiles and analyzes data so that the system can deliver trusted information to users. The Information Analyzer (IA) is used to scan or sample data stored in diverse sources to assess its quality. MDM also uses some complementary tools: *IBM InfoSphere QualityStage*, which allows us to define rules to standardize and match free-form data elements which is essential for effective probabilistic matching of potentially duplicate records, and *IBM WebSphere AuditStage*, which enables us to apply professional quality control methods to manage the accuracy, consistency, completeness, and integrity of information stored in databases. We also use statistics provided by *IBM InfoSphere DataStage* to compute chosen quality and trust factors.

This set of tools provides a comprehensive approach to data quality and data trust management. This approach not only resolves some problems during the data acquisition but also allows us to control the level of data trust and to give up-to-date information about the trustworthiness to a user. This comprehensive approach is novel. Moreover our solution does not require any specialized hardware or operating system and is able to cooperate with any commercial data base systems.

1) IBM InfoSphere MDM Server: The InfoSphere Master Data Management Server has a new feature allowing users to define and add quality and trust factors to the data of their enterprise. This new data structure enables the user to store metadata required to compute scorings for trust and quality of data. Provided wizards allows user to modify the data model in a simple way.

2) IBM Information Server: IBM Information Server addresses the requirements of cooperative effort of experts and data analysts with an integrated software platform that provides the full spectrum of tools and technologies required to address data quality issues [1]. It supplies users and experts with the tooling that allows the detailed analysis of data through profiling (*IBM InfoSphere Information Analyzer* and *IBM InfoSphere AuditStage*), cleansing (*QualityStage*) and data movement and transformation (*DataStage*). In this paper we concentrate on data profiling and analysis handled mostly by Information Analyzer (IA), AuditStage (AS), and partially on QualityStage (QS).

IA, as an important tool of *data quality assessment* (DQA) process, aids the exposing technical and business issues. The technical issues detection is a simpler part of the process based on technical standards and covers important problems including different or inconsistent standards in structure, format, or values, missing data and default values, spelling errors, data in wrong fields, and buried information in free-form fields. Business quality issues are more subjective and are associated with business processes such as generating accurate reports. They require the involvement of experts.IA helps the expert in systematic analysis and reporting of results, thereby allowing him to focus on the real problem of data quality issues. This is done through tasks like column analysis, key analysis(Primary and Foreign Key) and cross-table analysis.

a) QualityStage: IBM InfoSphere QualityStage (QS) complements IA by investigating free-form text fields such as names, addresses, and descriptions. QS allows users to define rules for standardizing free-form text domains which is essential for effective probabilistic matching of potentially duplicate master data records. It provides user with functions such as free-form text investigation, standardization, address verification and record linkage and matching as well as survivorship that allows best data across different sources to be merged.

b) AuditStage: IBM WebSphere AuditStage (AS) enables user to apply professional quality control methods to manage different subjective quality factors of information stored in databases such as accuracy, consistency or completeness. By employing technology that integrates Total Quality Management (TQM) principles with data modeling and relational database concepts, AS diagnoses data quality problems and facilitates data quality improvement effort. It allows performing assessment of the completeness, validity of critical data elements and business rule compliance. AuditStage is very useful tool for assessment of the *consistency* factor allowing cross-table rules validation.

III. The Notion of Trust

Trust is an extension of data quality. We are looking for additional factors because data quality is not the only factor influencing the trustworthiness of data and these two concepts are not necessarily correlated. Low-quality data may be trusted in some situations and high-quality data may have low trustworthiness in other. The value of trust strongly depends on the user requirements and usage context. In this section we discus a data quality and we describe the notion of trust.

A. Data Quality

The data quality concept has been widely discussed in literature [2]–[4], [6], [9], [11], [19]–[21], [25], [27], [28], [32], [33] usually in the context of a single data source. Some work tough has been also done in the context of integrated data [5], [7], [12], [13], [18], [24] emphasizing the importance of data quality assurance in this area. Batini and Scannapieco [4] have given three examples of organizational processes where DQ aspects are particularly important.

- Customer matching – it is a common issue in organizations where more than one system with overlapping databases exists. A typical result is an inconsistent and duplicate information.
- Corporate house-holding – is a problem of identifying members of household (or related group). This context-dependent issue is widely described in [30].
- Organization fusion – is the issue of integration legacy software in case of organizations or units merge.

The definition of the quality that we are using in this work originates from the one provided by Naumann [18] as an attempt to provide an operational definition of DQ as an aggregated value of multiple IQ-criteria (Information Quality Criteria). IQ-criteria are there classified into four sets:

- Content-related – intrinsic criteria, concerned with the retrieved data,
- Technical – criteria measuring aspects determined by software and hardware of the source, the network and the user,
- Intellectual – subjective aspects of data that shall be projected to the data in the source,
- Instantiation-related – criteria related to the presentation of the data.

We follow the Naumann's approach by defining data quality as a aggregated value of multiple DQ-factors. Later we will extend this definition introducing the trust notion.

B. Trust Definition

Following Naumann's definition of data quality, we define trust (data trust, DT) through factors that influences the trustworthiness of data.

Definition 3.1 (Data Trust): Trust is the aggregated value of multiple Data Trust factors.

This definition provides flexibility when defining trust for a specific industry and user requirements. The trust factor (DT-factor) may be a DQ-factor, or non-quality (NQ) factor.

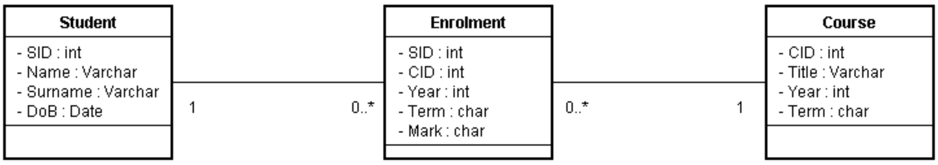


Fig. 1. Sample database schema representing enrollment of students into courses

Here we concentrate on data quality factors, and especially on accuracy dimension. However other factors like data lineage, security or trust of data source can be considered.

C. Accuracy dimension

Accuracy is included by most data quality studies as a key factor [2], [4], [12], [16], [20], [21], [26], [29], [31]. Although the term has an intuitive appeal, there is no commonly accepted definition of what it means exactly [29]. Ballou and Pazer [2] describe accuracy as ”the recorded value being in conformity with the actual value.” Kriebel [16] characterizes accuracy as ”the correctness of the output information.” Thus, it appears the term is viewed as equivalent to correctness.

In [4] accuracy is defined as ”the closeness between a value v and a value v', considered as the correct representation of the real-life phenomenon that v aims to represent.” The simple example can be the name of the city $Toronto$, the value $v = Tronto$ is incorrect (inaccurate) and $v' = Toronto$ is correct (accurate).

Parssian et al. [20], [21] formalize the notion of accuracy and propose quality metrics to assess the quality of basic queries. However definition proposed by Parssian et al. describes accuracy on the higher level of granularity. They consider the tuple as a whole, we in a contrast considering each attribute separately.

The accuracy may be calculated in three possible ways:

1) Inspection by expert – the expert reviews records and marks inaccurate ones;
2) Automatic or Semi-Automatic inspection – the system refine set of candidate records based on some predefined set of rules. Expert may or may not review and correct the result;
3) As a distance function utilizing dictionary sets – we may calculate the minimal number of operation required to transform the value v into the closest dictionary entry v'. Based on that calculation experts may identify low quality entries and apply changes to improve quality.

In all cases we consider only measurement. Experts and system is not allowed to perform any changes in data content. Moreover the (semi-)automatic inspection may be done only if some patterns apply to the field or dictionary may be defined. Examples of fields for which we can define some accuracy rules are SIN, postal code or phone numbers where some common rules apply. The dictionary, sometimes called code table, may be defined for fields like city, country etc. However, there are fields where we are unable to define neither rules nor dictionary. I.e. notes defined as a text field may contain any string. Moreover it is not possible to confront in the automatic way the stored data with real world to confirm the accuracy.

The accuracy and inaccuracy of data may be perceived in many different ways depending on the domain. At this point we will consider two approaches *error bar* and *boolean*. The first one, borrowed from engineering, gives some flexibility but is applicable only to ratio- or interval-scaled types [14]. The second one considers the value strictly in ”black and white” – it may be accurate or inaccurate, one or zero, and there is nothing in between.

1) Error bar: In many areas of science, especially in physics and engineering, the metrology defines a way of calculation of measurements' error and its propagation in calculations. We can see data stored in database as such measurements' results taken with some error and can define *inaccuracy* of data as the relative error. Value of this error is normalized. Such definition applies to numerical fields, ratio- and interval-scaled, where the distance between two values is meaningful, and expresses the relative distance between stored value and real world value. For text fields we can define distance using number of atomic changes required to transform our value into real world value divided by length of real world value.

Accuracy defined in this way however is applicable only to limited types of attributes (preferably numerical) and in specific domains. In general this kind of interpretation is inapplicable in database environment, where we are not provided with any information about inaccuracy of stored data.

2) Boolean accuracy: Boolean approach originates from the idea or rather assumption that each stored record has some source in the real world, and that this source can be ”linked” in an unambiguous way to the stored tuple. In such case we consider the value accurate if the value of stored instance is equal to the value of its real world source. This approach bounds the instance and source, however does not provide any flexibility. The stored instance is either accurate or inaccurate, and there is nothing in between of those extremes, even though this approach can be applied widely.

3) Hybrids: The third approach merges two ideas. We allow our stored instance to vary slightly from the real world source but the result remains boolean. The stored value v is considered accurate in this case if $|v - v'| \leq \epsilon$, where v' is a real world value and ϵ is some predefined, acceptable variation. This variation may also be expressed as a relative value (i.e. percentage). Then value is accurate if $\frac{|v-v'|}{v'} \leq \epsilon$.

In the following consideration we restrict the calculation methods to boolean interpretation of the accuracy; however

TABLE I
SAMPLE DATA IN TABLES STUDENT, COURSE AND ENROLLMENT FROM THE DATABASE FROM FIGURE 1. GRAY FIELDS CONTAIN INACCURATE VALUES

Student

CID	Name	Surname	DoB
111111111	AAAAAAAA	AAAAAAAAAA	1971-01-01
222222222	BBBBBBBB	BBBBBBBBBB	1981-01-01
333333333	CCCCCCCC	CCCCCCCCCC	1973-05-02
444444444	DDDDDDDD	DDDDDDDDDD	1971-09-10

Course

CID	Title	Year	Term
CSE2222	Software Tools	2009-10	F
CSE2222	Software Tools	2009-10	W
CSE2222	Software Tools	2009-10	S
CSE1111	Data Bases	2009-10	F

Enrollment

SID	CID	Year	Term	Mark
111111111	CSE2222	2009-10	F	A
222222222	CSE1111	2009-10	F	B
333333333	CSE2222	2009-10	F	A
222222222	CSE2222	2009-10	W	C

hybrid interpretation may be used as well. The only requirement is that method should be defined before, so we can use it to determine the accuracy of data.

D. Accuracy – Definitions

The base concept of the accuracy here is the confrontation of stored values with their real world sources. We assume that each tuple can be unambiguously linked with the real world element through the accurate key. If we can identify this real world element, then we can compare stored values and real world values to determine the accuracy of stored values. This real world element is called *source*. The linkage with real world entity is a key property in accuracy semantic and is necessary to provide the interpretation of derived accuracy values. We are using the distinction between keys and measures used commonly in warehouses. We would like to provide the definitions for both those types. The key attribute can be seen as a link connecting stored data with source entities. The ability of unambiguous identification of the source by the key is the base for accuracy calculation.

Example 1 (Accuracy interpretation): Let's consider the sample schema presented on Figure 1. This schema consists of three relations:

- $Student(\underline{SID}, Name, Surname, DoB)$
- $Course(\underline{CID}, \underline{Year}, \underline{Term}, Title)$
- $Enrolment(\underline{SID}, \underline{CID}, \underline{Year}, \underline{Term}, Grade)$

There are defined also two foreign key constraints on the table Enrollment:

- FK(SID) – where SID is a key in Student relation
- FK(CID, Year, Term) – where CID, Year and Term are compound key in Course relation

The accuracy of any field in Student relation can be determined only if we can determine the real world source. This determined value can be also interpret only in presence of the key. For example, let's consider student with SID=111111111. The accuracy of the Surname can be determined only by comparison of stored value and source value. If the source value is different than the stored one we say that stored value is inaccurate.

Now considering student record with SID=333333333, which is marked as inaccurate, we are not able to link this entry with source. In such situation we cannot compare stored values with source, because source is unknown.

Base on the above example we see that we have to distinguish two types of accuracy:

- accuracy of key – determines the accuracy of the entire row,
- accuracy of measure – is determined by two factors: the accuracy of the key and the source (by comparison of the stored and source values)

In this example we also see that inaccurate key breaks the linkage between stored entity and source entity making impossible the assessment of the accuracy of measures.

Definition 3.2 (Accuracy of key): The key attribute is accurate if and only if it unambiguously identifies the source object. If the key is a composite the accuracy is calculated for the entire key, and is inherited by each key element.

Example 2 (Accuracy of key): Let's consider relations `Student` and `Curse` presented by Table I. Relation `Student` has a key attribute `SID`. It can be the same number that appears on the students id and it allows us to identify a student in unambiguous way. If we cannot identify the student (i.e. there does not exist student with chosen SID) all data related to this key will be inaccurate. Relation `Course` has a compound key that consists `CID`, `Year` and `Term`. This triple allows us to identify courses. If one of those elements is inaccurate (i.e. term is equal `'T'` which is incorrect value), entire key is inaccurate and all data related to this key is considered as inaccurate. Moreover, if `SID` or one or more key elements from `Course` are inaccurate, then all related entries in `Enrollment` will be inaccurate.

The accuracy of measure is dependent on the accuracy of the key. The accuracy of the measure is based on the idea that the data to be accurate should match source data.

Definition 3.3 (Accuracy of measure): We consider the measure accurate if and only if the key of the tuple ($x.X_0$) is accurate and measure's value ($x.X_i$) is equal to the real world value ($x'.X_i$) identified by the key of the tuple that both key and measure belong to.

$$Acc(x.X_i) = \begin{cases} 1 & \text{if} \quad x.X_i = x'.X_i \wedge Acc(x.X_0) = 1 \\ 0 & \text{if} \quad x.X_i \neq x'.X_i \vee Acc(x.X_0) = 0 \end{cases} \tag{1}$$

IV. Acquisition and Processing of Accuracy

Trust and quality processing described below is one of the most novel aspects of our work. An important advantage of our approach is the use of existing set of tools, slightly modified or extended to serve in the new context. We extended these tools by creating data structures to store and process meta-data describing data quality and trust. We have designed mechanisms for assessing accuracy of data. Our approach is view-based estimation of accuracy of a SQL answer.

In this section we will explain how the information about the accuracy of data is gathered and processed. We will use the view-based model to estimate the accuracy of data in each view's attribute, then the estimated accuracy will be inherited by view's attribute elements.

A. Quality Data Structures

MDM provides a mechanism that enables an extension of the existing data with trust/data quality factors. These extensions may be defined as *persistent object* and stored in the database or be as *transient objects* calculated at run time. We would like to use this mechanism and employ it to tag stored data with the accuracy score.

B. Views

Thinking about the accuracy of data we can easily notice that checking each value and confronting it with the source is not applicable approach, especially in the context of huge governmental or commercial databases. From the other hand big picture defined by some general statistical analysis is often not enough. Our idea inspired by Motro and Rakov's work [17] is employing views over base tables to represent external knowledge about the quality of data.

Definition 4.1: View is a intentional or extensional set of tuples chosen from base table.
In other words the view can be defined by specifying a query or by specifying an extension – set of tuples – by pointing directly members of this set.

Views can overlap. It means that one tuple from the base table can be a member of one or more view. We assume also that each tuple is a member of at least one view. This assumption can always hold, because the entire base table can be seen as one, very general view.

We preserve the relational semantic of view, which can be seen as a set of attributes (in exactly the same way as a table). For each attribute then, the accuracy metric can be calculated. This metric expresses the statistical likelihood of choosing accurate value. The value of this metric is inherited by all elements of the attribute.

Value of the accuracy metric is calculated base on accuracy measure. When operating on sufficiently small set, we can test each value to calculate metric, otherwise sampling and statistical methods can be used to determine the likelihood of choosing accurate value.

The view overlapping rises some problems. Without detailed knowledge of error distribution we are not able to assess the accuracy of the intersection of two views. As an intersection we understand a set that is a subset of both views. In such case we assume that smaller view provide us with more detailed information hence we will use its metrics to tag tuples from the intersection.

Example 3 (View-based accuracy): Let's consider the relation Course (Figures 1 and I). Let's also assume that we have information that the accuracy of entries inputed for the summer term 2009-10 have lower accuracy than average. Base on that knowledge we define view V containing all courses having Term='S' and Year='2009-10'. For each attribute we calculate two metrics: one general, which covers entire attribute, the second one over view V. Each value of the attribute is tagged with appropriate value – if it is covered by view V, it is tagged with metric calculated for the view's attribute, if not, the general metric is used.

C. View elements tagging

In previous subsection we have defined accuracy metrics, which are calculated for each attribute of the view. Here we will explain how view's elements are tagged with the accuracy scores (value of accuracy metric).

Let's remind the assumptions we have made: (1) Views may overlap; (2) Inaccurate elements are distributed over attributes in uniform way. Value of the accuracy calculated for the attribute of the view is inherited by each element of the attribute. Those tags can be aggregated at the end of the processing. The average calculated over all tags will be an equivalent to the weighted average over views. We will use values assigned to elements rather than view in the estimation of the accuracy of query answer.

V. Query processing with accuracy

The trust alone is just yet another piece of data given to the user. The really important question is *What can be done with this information?* Lets consider now some use cases showing usage of the trust in the system.

We have shown that the trust score can be incorporated in our meta-data and linked with each field in the database if desired. This information can be then returned to the user. Even though this information is very detailed, it is not practically useful in all cases. Without algorithms to propagate trust in the query processing, we can only annotate a tuple and return it to the user. However we can build some statistics over this information that can be used later.

One of the problems that are currently unsolved is propagation of trust scores in the query processing. We are currently working on methods allowing us to estimate the trust of the result of the SQL operator based on the estimated trust of entry set. We are using estimates in this context because it is significantly less expensive than reaching out each time for the data.

Analyzing the quality of the query response, the key operation to be considered is selection. All other operations, except projection, rely somehow upon selection. For example *join* operation can be expressed as Cartesian product followed by

TABLE II
DIFFERENT CASES OF THE ACCURACY DEPENDING ON THE AGGREGATION AND TYPE OF SELECTION

	With aggregation		**Without aggregation**
	Count	Other	
Non-trivial	$Acc(X_i) \cdot 1$	$Acc(\sigma) \cdot Avg(Acc(X_i))$	$Acc(\sigma) \cdot Acc(X_i)$
Trivial	$1 \cdot 1$	$1 \cdot Avg(Acc(X_i))$	$1 \cdot Acc(X_i)$

selection and *group-by* operation can be seen as a bunch of selections followed by aggregations.

The accuracy of measure can be determined and interpreted only in the presence of the accurate key. Because of this strong relation between key and accuracy of measures, the latter can be seen as functionally dependent on the key.

Definition 5.1: The accuracy context is a set of key attributes allowing for an unambiguous assignment of accuracy for a field of specific entity or aggregate over a set of entities.

Example 4 (Context preservation): Let's consider the relation Student (Figure I.) and two queries:

- `SELECT * FROM Student`
- `SELECT Name, Surname, DoB FROM Student`

Both queries return all records from the relation Student. The first query however returns entire relation (all attributes) when the second query eliminates the key of the relation (SID). As a result of the elimination of SID from the answer we are unable to connect the arbitrary element from the output with its source entity, hence we cannot calculate the accuracy.

A. Considered query types

In this work, the following types of queries will be considered:

- equi-selects – we consider the equi-select with the key of selection being an attribute which has nominal or ordinal type, preferably a relation key;
- range selects – can be done over attributes which have ratio- or interval-scaled type (the distance between values has a meaning) or over ordinal attributes (the order has a meaning), but not over nominal attributes;
- select with aggregation – both equi- and range select can consist the aggregation. We have found two subsets of this type of queries:
  - count – the value of the attribute (if not NULL) does not influence the result;
  - other aggregates – values have an influence on the result, the scale of the impact depends on the aggregate function.

When considering range selects following problem arises: some ranges may be defined in a way that the answer is entire relation. We consider this class of queries because even though is seems that some comparison is required to determine if tuple should be returned, in fact, no comparison is necessary.

Example 5 (Trivial range selection): Let's consider the relation Student and following range query:

```
SELECT * FROM Student
WHERE
DoB BETWEEN 1900-01-01 AND 2100-12-31
```

The range of DoB is from 1971-01-01 to 1981-01-01. We clearly see that the WHERE clause of the query covers entire relation and because of that the query is semantically equivalent to the query `SELECT * FROM Student`. In such case, comparisons are not necessary to derive the answer.

B. Accuracy components

We have identified two main components of the accuracy of the selection's result. The first one is the accuracy of the attribute factor, denoted as $Acc(X)$ where X is an attribute. Including this component seems to be natural and obvious. The second component is the accuracy of selection, denoted as $Acc(\sigma)$, which expresses the likelihood that an arbitrary tuple has been accurately selected. It is the likelihood that the arbitrary element from the selection key is accurate. This component has to be covered and propagated over all derived attributes.

We have to explore how those components interfere in different types of queries. It can be easily noticed that selection component will not appear in trivial selections since we cannot make any mistake. On the contrary in the query employing counting operation the accuracy of attribute being counted does not meter because we are interested only in the number of elements, not the exact values. Table II gathers all cases that we have identified. It also presents our preliminary proposition of calculation of the accuracy of the selection's result. It is based on the assumption that attributes are independent in terms of errors' distribution. In such case we can see the accuracy of derived element as a result of two independent events: accurate selection and accurate value of the considered attribute.

VI. CONCLUSIONS

Measuring data quality and data trust is one of the key aspects of supporting businesses in decision making process or data stewardship. Master Data Management in other hand supports sharing data within and across lines of business. In such case trustworthiness of the shared data is extremely important. Our investigation has resulted in consistent method of gathering and processing quality and trust factors.

In this work we have presented the *IBM InfoSphere MDM Server* and elements of *IBM Information Server* such as DataStage, QualityStage, AuditStage and Information Analyzer, and their ability to handle data quality and data trust. We have also presented the new notion of data trust. The process of gathering and computing data quality and trust factors has been described and explained using example.

This work covered only the introduction to the accuracy processing. We consider select as a most basic operation, which is necessary to proceed with other operations and though should be explored soon. We have proposed the data model for the accuracy storing and processing. In the future we are going to cover in our consideration all SQL operators including accuracy neutral operators (Cartesian product, projection) and other accuracy sensitive operators (join, group-by and set operations). We suspect that projection and Cartesian product are accuracy-neutral because those operations does not relay upon accuracy of input. They do not "touch" data values.

Acknowledgments

Author would like to thank Jarek Gryz and Parke Godfrey from York University for support and constructive criticism, Stephanie Hazlenwood and Paul van Run from IBM Toronto Lab for support and valuable discussions.

This work has been supported by IBM Center for Advanced Studies and NSERC.

References

[1] Alur, N., Joseph, R., Mehta, H., Nielsen, J. T., and Vasconcelos, D. *IBM WebSphere Information Analyzer and Data Quality Assessment*. Redbooks. International Business Machines Corporation, 2007.

[2] Ballou, D., and Pazer, H. Modeling data and process quality in multi-input, multi-output information systems. *Management Science 31*, 2 (1985), 150–162.

[3] Ballou, D., Wang, R., Pazer, H., and Tayi, G. K. Modeling information manufacturing systems to determine information product quality. *Manage. Sci. 44*, 4 (1998), 462–484.

[4] Batini, C., and Scannapieco, M. *Data Quality: Concepts, Methodologies and Techniques*. Data-Centric Systems and Applications. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[5] Bouzeghoub, M., and Kedad, Z. *Quality in Data Warehousing*. Kluwer Academic Publisher, 2002.

[6] Crosby, P. B. *Quality is free : the art of making quality certain / Philip B. Crosby*. McGraw-Hill, New York :, 1979.

[7] Cui, Y., Widom, J., and Wiener, J. L. Tracing the lineage of view data in a warehousing environment. *ACM Trans. Database Syst. 25*, 2 (2000), 179–227.

[8] Dreibelbis, A., Hechler, E., Mathews, B., Oberhofer, M., and Sauter, G. Master data management architecture patterns. http://www.ibm.com/developerworks/data/ library/techarticle/dm-0703sauter/index.html, 2007.

[9] English, L. Information quality improvement: Principles, methods, and management. Seminar, 1996. 5th Ed., Brentwood, TN: INFORMATION IMPACT International, Inc.

[10] Fan, W. Dependencies revisited for improving data quality. In *PODS '08: Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems* (New York, NY, USA, 2008), ACM, pp. 159–170.

[11] Foley, O., and Helfert, M. The development of an objective metric for the accessibility dimension of data quality. In *Proceedings of International Conference on Innovations in Information Technology* (Dublin, 2007), IEEE, pp. 11–15.

[12] Gertz, M., and Schmitt, I. Data Integration Techniques based on Data Quality Aspects. In *Proceedings 3. Workshop "Föderierte Datenbanken", Magdeburg, 10./11. Dezember 1998*, I. Schmitt, C. Türker, E. Hildebrandt, and M. Höding, Eds. Shaker Verlag, Aachen, 1998, pp. 1–19.

[13] Gupta, A., and Widom, J. Local verification of global integrity constraints in distributed databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., May 26-28, 1993* (1993), P. Buneman and S. Jajodia, Eds., ACM Press, pp. 49–58.

[14] Han, J., and Kamber, M. *Data Mining: Concepts and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers, 2001.

[15] IBM. Ibm master data management: Effective data governance. ftp://ftp.software.ibm.com/software/uk/itsolutions/ information-management/information-transformation/ master-data-management/master-data-management-governance.pdf, 2007.

[16] Kriebel, C. H., and Moore, J. H. Economics and management information systems. *SIGMIS Database 14*, 1 (1982), 30–40.

[17] Motro, A., and Rakov, I. Not all answers are equally good: estimating the quality of database answers. 1–21.

[18] Naumann, F. *Quality-driven query answering for integrated information systems*. Springer-Verlag New York, Inc., New York, NY, USA, 2002.

[19] Olson, J. *Data Quality: The Accuracy Dimension*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[20] Parssian, A., Sarkar, S., and Jacob, V. S. Assessing information quality for the composite relational operation join. In *IQ* (2002), pp. 225–237.

[21] Parssian, A., Sarkar, S., and Jacob, V. S. Assessing data quality for information products: Impact of selection, projection, and cartesian product. *Manage. Sci. 50*, 7 (2004), 967–982.

[22] Radcliffe, J. Magic quadrant for master data management of customer data. Tech. Rep. G00167733, Gartner, Inc., 2009. http://mediaproducts.gartner.com/reprints/ oracle/article78/article78.html.

[23] Radcliffe, J., and White, A. Key issues for master data management. Gartner Master Data Management Summit, Chicago, IL, 2008.

[24] Reddy, M. P., and Wang, R. Y. Estimating data accuracy in a federated database environment. In *CISMOD* (1995), pp. 115–134.

[25] Redman, T. C. *Data quality : the field guide*. Digital Pr. [u.a.], Boston, 2001.

[26] Shin, B. An exploratory investigation of system success factors in data warehousing. *J. AIS 4* (2003).

[27] Tayi, G. K., and Ballou, D. P. Examining data quality. *Commun. ACM 41*, 2 (1998), 54–57.

[28] Tupek, A. R. Definition of data quality, 2006.

[29] Wand, Y., and Wang, R. Y. Anchoring data quality dimensions in ontological foundations. *Commun. ACM 39*, 11 (1996), 86–95.

[30] Wang, R. Y., Chettayar, K., Dravis, F., Funk, J., Katz-Haas, R., Lee, C., Lee, Y., Xian, X., and Bhansali, S. Exemplifying business oppurtunities for improving data quality from corporate household research. In *Advances in Management Information Systems - Information Quality (AMIS-IQ) Monograph* (April 2005).

[31] Wang, R. Y., Pierce, E. M., and Madnick, S. E. *Information quality*, vol. 1 of *Advances in management information systems: Information Quality*. M.E. Sharpe, 2005.

[32] Wang, R. Y., and Strong, D. M. Beyond accuracy: what data quality means to data consumers. *J. Manage. Inf. Syst. 12*, 4 (1996), 5–33.

[33] Yang, L. P., Lee, Y. W., and Wang, R. Y. Data quality assessment. *Communications of the ACM 45* (2002), 211–218.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Multicriteria Evaluation of DVB-RCS Satellite Internet Performance Used for e-Government and e-Learning Purposes

Andrzej M. J. Skulimowski
AGH University of Science and Technology, Chair of Automatic Control, Decision Sciences Laboratory,
Al. Mickiewicza 30, 30-059 Kraków, Poland
and
Technology Transfer Centre, Progress and Business Foundation, Kraków, Poland
Email: ams@agh.edu.pl

Abstract— **In this paper we report the findings of the EU 6th Framework Programme Project "Rural Wings" concerning the selection, performance and evaluation of the satellite internet pilot sites, based on the case studies of ten such sites in Poland. First, we present the methodology of ex-ante assessment of specific needs concerning the intensity and the scope of use of the DVB-RCS bidirectional satellite internet technology in the mountain and rural areas that led to the selection of rural sites, where the bidirectional satellite terminals were installed . Then we review the operation of the pilot sites and their final performance evaluation. We compare the rankings resulting from the initial needs assessment with that one derived from the final evaluation and analyse the divergences. Finally, we propose a learning scheme resulting from ex-post evaluation of the initial ranking procedure, which allows to assess the adequacy of the multicriteria decision-making approach applied to derive the initial pilot sites' ranking.**

I. Introduction

OVER 25 million citizens of Europe live in the remote mountain, island or rural areas, where ground-based broadband internet is hardly available - or not available at all. An improvement of the internet infrastructure is therefore necessary to bridge the digital divide by ensuring the access to information, e-government, e-commerce and e-learning applications to the members of rural communities. In some European regions, such as low-populated Bieszczady mountains in Poland, Greek islands, or northernmost parts of Sweden or Finland, the satellite internet can provide a competitive, or the only solution to cope with this problem. Consequently, the European Commission has approved a series of research projects devoted to analyze the needs and find the best technical and organizational solutions for the deployment of satellite technologies for citizens, institutions and businesses in rural areas. One of them, the 8,8 MEUR project "Rural Wings" (www.ruralwings-project.net, www.pbf.pl/ruralwings), financed by the European Union within its 6th Framework Programme, started in January 2006, with the main aim to select, launch, and maintain pilot stations of satellite internet in the bi-directional DVB-RCS (Digital Video Broadcasting - Return Channel via Satellite) technology [1], [2] in 16 countries of the world. The stations should be situated in the remote rural or mountain areas, with no – or only poor quality – broadband internet access, and serve local communities in accessing the web, supporting thus a move from the traditional agriculture to the development of innovative and sustainable agriculture and other branches of rural economy such as agrotourism. The achievement of specific goals of each pilot station, related mostly to e-government, e-learning, and e-commerce, but also to the dissemination of research carried out at some of the stations, should had been assisted, monitored, improved, advised and reported by the partners of the above project assigned the role of National Coordinator in each country. The infrastructure of pilot stations consisted of the satellite terminal (D-Star) with dedicated software and hardware. Last, but not least, most sites have been connected to Local Area Networks and WiFi networks. The technical supervision comprised measurements of the technical performance of the satellite internet terminals (D-Star) and satellite transponders (in Poland's case Eutelsat Atlantic Bird 1B), including the transmission rates for down- and uplinks, investigation of transmission failures, and tracing the sources of disturbances.

The present paper reports the methodology of evaluation of the project's results in Poland, where the Progress and Business Foundation from Cracow, an academic NGO (www.pbf.pl) has been selected to play the role of the National Coordinator. As the assessment of a network of satellite internet local hubs had not been described before in the literature, the methodology elaborated for the network of ten sites in Poland has been also presented and made available to all other Project's partners..

The paper is organized as follows: first, we will present the methodology of assessment of specific needs concerning the intensity and the scope of use of satellite internet in the DVB-RCS technology in the remote mountain areas in Southern Poland. We will review the operation of the pilot sites where the bidirectional satellite terminals were installed, then we will analyse their final performance evaluation. In the final section we will compare the rankings resulting from the initial needs assessment with that one derived from the final evaluation. While the technical criteria of needs

The project Rural Wings was supported in part by the European Commission, grant No. FP6-IP-516161. The assessment of the applicability of its outcomes in Poland was co-funded by the ERDF within the Innovative Economy Operational Programme, 2007-2013, project No. WND-POIG.01.01.01-00-021/09: „Scenarios and development trends of selected information society technologies until 2025" carried out by the Progress & Business Foundation, Cracow, Poland, www.pbf.pl

978-83-60810-27-9/09/$25.00 © 2010 IEEE

assessment and evaluation were different, the target goals, i.e. intensity of use, measured as the transmission volume, the number of users and their regularity, measured by the standard deviation of the above, as well as the second goal: filling-in the gaps in the broadband coverage, and the third-one: the number of users willing to use the applications made available within the project, could be compared as *ex-ante* expectations at the stage of selecting the pilot sites, and as *ex-post* evaluation results. To make such a comparison possible, the criteria actually measured need to be transformed to the measures of the above goals, taking into account the stochastic character of the initial expectations. The method here proposed allows to derive a set of relevant needs assessment criteria from a larger set of criteria considered at the selection phase of the project. It is also possible to assess the adequacy of the multicriteria decision-making approach applied to derive the initial pilot sites' ranking. The above approach is universal in the sense that it can be applied to any similar problem, involving an initial *ex-ante* and *ex-post* rankings based on the actual performance.

II. The Needs and Analysis and Selection of Pilot Sites in the Beskidy Mountains

One of the first tasks of the project was the selection of pilot sites. Taking into account that the only geographical regions in Poland where the problems with low broadband access was related to the topography and geographical isolation were the Beskidy mountains in southern Poland, we made the decision to restrict the selection to the Małopolskie and Podkarpackie Voivodships, specifically to their counties situated in the mountains and fulfilling initial low-broadband-access assumptions.

Following the initial indications concerning the site selection criteria provided by the responsible project's partners, a site selection methodology tailored to the specific country needs was elaborated. The main three groups of selection criteria are listed below:

- **Geographical situation of sites** – according to the Project's goals, the more remote and isolated the site, the more eligible for the Project, but the diversification of geographical locations and an even distribution of sites over target areas were taken into consideration as well.
- **Existing internet infrastructure –** the pilot stations should be installed in areas without broadband facilities or – at least - the internet access provided by third parties should have lower transmission rates or be less reliable than the Rural Wings system. On the contrary, other existing IT infrastructure, such as school computer labs, LAN etc. would be of advantage for the project.
- **Availability of potential end-users** willing and capable to engage in the activities of the pilot station.

The assessment of each of the above criteria was based on the evaluation of a set measurable subcriteria, which were calculated from the data gathered in two questionnaires assessing the community and individual needs at potential pilot sites. The forms and criteria were enhanced by country-specific additional issues, in Poland relating a.o. to the ability to supply research results via satellite internet. The method to aggregate the data gathered in the questionnaires to the measures of the above criteria, as well as the choice of the multicriteria outranking method to establish the final sites' ranking was the task of the National Coordinators.

The pilot site selection was performed in four-steps:

(i) First, the country needs as a whole and the needs of regions, sectors and types of institutions were assessed in order to choose most appropriate target groups and areas.

(ii) Then the questionnaires were mailed to the selected target groups: local authorities, schools, national parks, research establishments, and business support organisations providing adult training. The data gathered were then verified during field visits and otherwise.

(iii) Taking into account the volume and the quality of data gathered, we elaborated a method to transform the questionnaire data into the measures of subcriteria. Then we used the multicriteria outranking method based on the reference sets approach [3] to establish a preliminary ranking of sites.

(iv) The selected sites, were visited in the order yielded by the outranking procedure, to present the Rural Wings project to the appropriate local authorities. Based on the technical feasibility of installing the DVB-RCS equipment at the selected sites, the results of negotiations, and the final assessment of the viability and usability of site derived from the field visits, the Selection Committee could either decide to eliminate a site from the list, e.g. in case of the lack of support from the school authorities for this site, or to change its rank. The latter might happen usually when the final decision to sign the agreement was postponed by local authorities. Then it followed the installation of DVB-RCS D-Star terminals according to the finally derived order.

Consequently, the above selection process started in Poland in early 2006 from a detailed country needs analysis in the areas of DVB-RCS technology, e-government, e-learning and other e-services and applications, which were to be provided by the project to the end-users. The above mentioned applications have been the top priorities of the „ePoland" [6] programme, first phase of which was realised prior to launching the Rural Wings project. It created thus a background for the implementation of the Project providing IT training of teachers, creating multimedia information centres, educational content and portals, electronic libraries, and content servers.

Based on the statistical data concerning the broadband access in different regions of Poland we refined the study to identify geographical areas and institutions relevant to the project, such as research institutes, national parks, local cultural centres, schools, public libraries etc. with no or only a poor-quality broadband access (nb. the dial-up connections were already commonly available via the National Telecom since the 90's). The systematically updated maps of broadband penetration were used to veryfy the needs. Future plans to establish commercial wireless broadband access by local providers were considered as well.

Thus, it was decided that the Polish sites should be situated in Southern Poland, except two potential locations in Polish off-shore research stations in Svalbard and Antarctic St.

George Island. The other potential implementation region in Northern Poland is more likely to be covered by wireless internet as it is situated in the lowlands. Furthermore, the study allowed to determine the main categories of potential satellite internet pilot stations in Poland, namely:

- public access points in local government offices, cultural centres and telecentres,
- rural schools and public libraries,
- remote research stations and national parks, and
- touristic establishments in mountain areas.

Other categories considered were policlinics and fire brigade centers in remote rural areas that perform also educational activities, for which, as it turned out, there already existed governmental or regional plans to endow them with modern IT infrastructure, including broadband access.

As only two sites had to be installed in phase I of the project, the country needs analysis was extended and updated prior to the phase II in 2007. The needs analysis in the areas of DVB-RCS technology, e-government, e-learning and other Information Society services, technologies and applications, has been performed using the IST foresight results for Poland elaborated during the 5th Framework Programme FISTERA project [5],[6], while the analysis of future applications has been performed within the ERDF-financed foresight project quoted in the footnote above.

In the second step, the questionnaires were distributed to over 400 institutions in Małopolskie and Podkarpackie regions (Voivodships) identified as potential pilot sites. From over 120 replies, about 70 seemed to be eligible pilot stations. Individual site visits and interviews reduced this number to about 40, eliminating non-public institutions or multiple institutions in one village.

Further, in Step (iii), the sites' selection criteria and selection methodology have been refined using the multicriteria analysis. We have elaborated a dedicated multicriteria site selection methodology which uses the reference sets approach [3] and takes into account the specific Poland's needs.

A preliminary ranking using a set of target reference points was established, so that for each of the above listed four categories of pilot stations a model pilot site was defined, characterized by desirable values of each of the above three metacriteria. The models played the role of non-attainable (ideal) target reference points in the above outranking method, namely for each of the complete questionnaires received form an eligible pilot site we calculated the distance to the model target point representing an appropriate category. At the same time minimal requirements, specified within the Project, had to be fulfilled by all sites considered; they played the role of status quo reference points, as defined in [9] and had been used as constraints. Ten sites with the highest distance scores were thus found to be the candidates for the 1st phase installations of the Rural Wings project in Poland. Only two of them had to be actually installed, while the others could remain candidates for the subsequent phases two and three.

TABLE I.
THE INITIAL RANKING OF RURAL WINGS PILOT SITE CANDIDATES IN POLAND

Rank	Name of the Pilot Site	Place, Region	Type	Users	Employees with IT background	Internet available
1.	Arctowski Research Station	St. George Island, Antarctica	Research	a) researchers b) visitors	All researchers at the Station	NONE
2.	Babiogórski National Park	Zawoja, Małopolska (Beskidy Mts.)	Wildlife reservation/ Research / Tourism	a) scientists b) visitors c) local community members	Two IT-specialists	Dial-up connection 64Kb/sec (DSL)
3.	Polish Polar Station – Hornsund	Hornsund Fjord, Svalbard	Arctic Research	a) scientists c) visitors	All researchers working at the Station	(Satellite, high disturbances)
4.	Polana Primary School	Polana, Podkarpackie (Bieszczady Mts.)	Primary School	a) schoolchildren b) farmers c) tourists	One IT teacher responsible for the hardware and LAN	NONE
5.	Magurski National Park	Krempna, Małopolska, Beskid Niski Mts.,	Wildlife reservation/ Tourism/ Research	a) scientists b) visitors c) local community members	One IT-engineer	Radio ADSL connection (disturbances)
6.	Gładyszów Primary School	Gładyszów South-Eastern Małopolska	Primary School	a) schoolchildren b) farmers c) tourists	One IT teacher responsible for the hardware and LAN	Ground-based ADSL connection

Source: Progress & Business Foundation (2009)

In the final, fourth, phase of the selection procedure we investigated the synergy with other projects of similar nature and goals as Rural Wings, that - if carried out in the same area - would constitute a challenge to the Project. Each such situation had been investigated in detail and the results actually influenced the final ranking. For instance, sites endowed already with broadband access from earlier programmes, such as “Ikonka” or “Interkl@sa”, were not taken into account for the Rural Wings pilot phase, that allowed to reduce the number of initial highest-ranked candidates to six.

During the selection process we took into account new opportunities that arose from the contacts with the leaders of local action groups, institutions hosting pilot sites, and local administration. A possibility to establish bi-directional scientific and educational data interchange between several Polish research stations: the polar station in Hornsund, the Antarctic research station on St. George Island, the Magurski, and Babiogórski National Parks, has been investigated as an additional chance and added value of the Project requiring the use of specialised software tools. The local administration representatives have been interested mostly in e-government applications and in teaching the members of local communities how to use them.

The results of the above presented selection procedure – six locations in the Antarctica, Svalbard and in the Beskidy mountains - are presented in Tab.1 above.

By the end of June 2006 the phase I of the pilot sites selection was completed and working contacts established with the representatives of all selected candidate sites. However, the Antarctic Arctowski Research Station, ranked 1, could not get the satellite connection because of the technical infeasibility of the Atlantic Bird 1 B satellite to reach the St. George Island. Furthermore, heavy snowfalls, that came as early as in September, made the installlation of the D-Star terminal in Hornsund, Svalbard, impossible. Finally, sites ranked 2 and 4: the Babiogórski National Park and Polana Primary School turned out to be successful winners of the phase I of the selection process. A similar procedure has been performed for the 2nd phase pilot sites, yielding a list of next-best eight pilot stations. All they are presented in the next Sec.3.

III. The operation of the DVB-RCS pilot sites in Poland

First two pilot sites, selected within the above described procedure, were installed in March 2007: the Babiogórski National Park with the site in the Park's headquarters in Zawoja, Beskidy Mountains, Małopolskie Region, and the Polana Primary School, situated in the Bieszczady Mountains, Podkarpackie Region. The selection of sites performed for the second project period yielded eight subsequent sites. All selected sites are shown in the Fig.1 and Tab.2.

The phase II installations started in two municipalities in in the Beskidy Mountains in Małopolska in 2008: Wiśniowa with the sites at the Lubomir mountain – the didactic astronomical observatory of the Jagiellonian University and in the Wiśniowa Secondary School, and the municipality of Raba Wyżna with sites at the Rokiciny Podhalanskie Telecenter and at the Public Library in Skawa. Following a needs' analysis, the sites have been endowed with the software and hardware suitable for the selected scenarios of use, and the terminals were connected to WiFi networks.

The remaining four pilot sites were established at the Primary School in Nowy Łupków, the Secondary School No. 9 in Kęty, finally, the Primary Schools in Myczkowce and Harkabuz in 2009. The latter sites are situated in the mountains as well: Kęty in the Beskid Żywiecki Mountains, Nowy Łupków and Myczkowce in the Bieszczady Range in South-Eastern Poland, and Harkabuz in the Central Beskidy Mountains. Basic characteristics of all above-mentioned pilot sites is presented in Tab.2 on the next page.

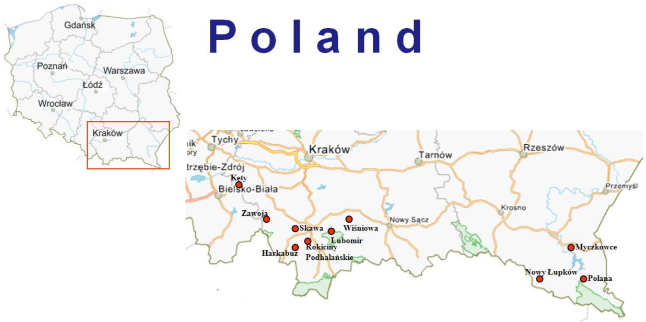


Fig 1. The situation of Polish satellite internet pilot sites selected for the Rural Wings project

The technical and training support activities provided by the National Coordinator were supplemented by monitoring the operation of the pilot sites and collecting the data for final evaluation. They included a.o. on-site consultancy, measurement of transfer rates, consulting concerning hardware of satellite terminals and WiFi installations, monitoring the use of applications supplied by other project's partners.

Further needs analysis concerning the hardware and software, based on the actual users' needs elicited during the pilot phase, had been carried out.

The National Coordinator's team reviewed the users' scenarios available in the Rural Wings project in order to plan a most effective project implementation and to assist the users. A new package "Improving internet access to public services and the electronic office (e-government) features learning" was defined and included in the scenario portfolio as this has been expected to be the most relevant issue in Poland at time when the project was carried out. Then the recommended scenarios for each pilot sites were chosen by the local community leaders assisted by the national Coordinator. The initial assignment of scenarios is presented in Tab. 3 in the next Section.

The *ex-ante* expectation concerning the use of satellite internet shown in Tab.3 served as a base for the *ex-post* assessment of the operation of sites and project's goals achieved that are discussed in the next Section.

IV. Final evaluation of the project's results in Poland

The evaluation of the project's results in Poland presented in this paper was based on the data gathered by the National Coordinator during the period of 2006-2010 when the project was carried out. Having selected and implemented the pilot sites, the evaluation process had a twofold character: on one hand it was performed as a project's internal workpackage, according to the same rules and criteria for all countries and regions.

The core of the corresponding evaluation methodology was provided by the Coordinator of the project and by the Consortium Partner responsible for the evaluation, while this methodology was continuously improved during the exchange of views at all Consortium meetings and the e-mail discussions afterwards. This part of the evaluation process was user-centered, based on the evaluation forms that had been sent to the end-users, filled-in at the training seminars and during other project's events, and were available on the web to be filled-in on-line.

Other streams of the monitoring and evaluation activities were linked to the technical aspects of the current operation of pilot sites, especially the users' feedbacks, have been analyzed by the National Coordinator's experts. The data on the transmission rates (down- and uplink) were continuously monitored and gathered for a later overall evaluation. Similarly, the information about the use of the terminals, such as the applications used, number and social structure of end-users, time at use, technical problems encountered, were gathered as well. The functionality of applications used and their responsiveness to the needs of different groups of end-users have been also evaluated.

While multicriteria outranking approaches based on weighting the individual criteria are commonly prevalent in deriving the rankings from individual scores, we have found out that a similarity measure to the ideal and satisfactory objects

Table II.
The pilot stations of the 6th FP Rural Wings project in Poland

ID	Place, host institution	Start of operation	Geographical situation	Description and type of the pilot site	No. of inhabitants / users	Goals to be achieved by this site and main target groups
POL 01	ZAWOJA, Babiogórski National Park	2007.03.28	longitude: 19°54, latitude: 49°65, altitude: 650 m asl	The pilot site is situated in the headquarters of the Babia Góra National Park (BGPN). close to Babia Góra (1725 m asl) Site type: Research Station & Wildlife Reservation	6200 / 2000	1. Supporting local administration in environmental issues 2. Better environmental education, enriching science learning and scientific activity in remote rural areas: high school and university students, teachers, ‘green schools’ participants 3. Using internet access for carrying out scientific research. Specific applications: producing and disseminating videos with observations of the Park's wildlife
POL 02	POLANA, Polana Private Salesian School	2007.03.29	longitude: 22°35, latitude: 49°18, altitude: 639 m asl	Polana is a small village in the south-eastern edge of Poland. It is situated in the central area of the Bieszczady Mountains. Site type: rural school	400 / 80	1. Using internet access for learning at school. 2. Using internet access for learning at home. Specific applications: local e-office access, WebTV - schoolchildren use cameras to record school events, then upload them to the web, e-learning applications supplementing the biology and English lessons
POL 03	WIŚNIOWA, The Private Adult Secondary School	2008.02.26	longitude: 20°12, latitude: 49°79, altitude: 380 m asl	The Wiśniowa Municipality is located in the mountains. 50 km to the south from Krakow It is surrounded by Site type: rural school	1900 /40	1. Using internet access for learning at school 2. Using internet access for learning at home Specific applications: UNITE platform -used during English lessons, and supplementing the biology and geography lessons
POL 04	MT. LUBOMIR, The Astronomical Observatory of the Jagiellonian University	2008.02.27	longitude: 20°08, latitude: 49°75, altitude: 904 m asl	Mt. Lubomir is situated in the north-eastern part of Beskid Makowski Mountains. The site is endowed with indoor WiFi for employees, and outdoor one for visitors. Site type: remote research station	1200 /100	1. Using internet access for carrying out scientific research. 2. Using internet access for learning. Specific applications: Discovery Space (D-Space) and similar applications allowing to transmit astronomical data from Lubomir via internet and to join the network of similar observatories. With D-Space the visitors of the Observatory are able to compare the views of astronomical objects from different telescopes
POL 05	ROKICINY PODHALAŃSKIE, Raba Wyżna Telecenter	2008.02.28	longitude: 19°91, latitude: 49°57, altitude: 550 m asl	Rokiciny Podhalańskie is a village in the Raba Wyżna municipality, about 70 km south from Krakow, in the Beskidy Mts. The pilot site serves the local telecenter. The outdoor WiFi, reaches the nearby recreational area. Site type: rural telecenter	Raba Wyżna Municipality: 13632, Rokiciny Podhalańskie: 1450 / 250	1. Intensive e-government-oriented training programs 2. The use of municipal web service of Raba Wyżna 3. Using internet access for learning at school 4. Using internet for learning at home and during leisure time 5. Using internet access for learning at work Specific applications: Electronic consultations on local community matters proposed by municipal authorities, WebTV - users record relevant events from the life of Raba Wyżna municipality, then upload it to the web.
POL 06	SKAWA, The Municipal Public Library	2008.02.29	longitude: 19°90, latitude: 49°62, altitude: 514 m asl	Skawa is a village in the Beskid Zachodni Mountains about 70 km south from Kraków. Site type: rural library	4000 / 300	1. Using internet access for learning at school 2. Using internet access for learning at home 3. Using internet access for learning at work Specific applications supplement the lectures on ecology and biology at the Library, and municipal web service of Raba Wyżna (e-govt)
POL 07	NOWY LUPKOW, Primary School	2008.11.06	longitude: 22°05, latitude: 49,15, altitude: 592 m asl	Nowy Łupków is situated in the Podkarpackie Voivodship, in the Bieszczady Mountains. Site type: rural school	390 /60	1. Better education: Enriching science learning and scientific activity in remote rural areas 2. Rural school teachers’training Specific applications: WebTV, schoolchildren record school events, then upload the movies to the web
POL 08	KĘTY, Secondary School No.9 in Kęty	2008.11.07	longitude: 19°90, latitude: 49°62, altitude: 514 m asl	The Community of Kęty is located in the Soła river valley, at the foot of the Beskidy mountains. Site type: county-level school	19500 / 300	1. Better education: Enriching science learning and scientific activity in remote rural areas 2. Rural school teachers’ training Specific applications: Teachers` IT training seminars. Xplora, UNITE platform, Experinet have been used to enhance natural science lessons
POL 09	MYCZKOWCE Primary School	2009.02.06	longitude: 22°24, latitude: 49°26, altitude: 364 m asl	Myczkowce is a village in the Podkarpackie Voivodeship in south-eastern Poland. Site type: rural school.	510 /120	1. Better education: Enriching science learning and scientific activity in remote rural areas, 2. Rural school teacher training Specific applications: e-govt-oriented training, Cret@quarium, WebTV
POL 10	HARKABUZ, Primary School	2009.03.18	longitude: 19°50, latitude: 49°32, altitude: 809 m asl	Harkabuz is a village in the Raba Wyżna municipality, in the Beskid Zachodni Mountains. Site type: rural school	530 / 75	1. Better education: Enriching science learning and scientific activity in remote rural areas, 2. Rural school teacher training Specific applications: municipal web service of Raba Wyżna (e-govt), Cret@quarium, WebTV

Source: Progress & Business Foundation (2009)

defined at the selection phase would better correspond to the ideas underlying the functioning of pilot sites.

First, data characterizing the individual pilot stations' performance, containing 12 indicators, were gathered and indicators grouped into three groups: technical, intensity-of-use-related, and qualitative, Then the groups of criteria were aggregated to three synthetic objectives: those technical, based on transmission quality, those related to the intensity of use, and those describing the quality of fulfilling the Project's goals, based on interviews and qualitative assessments. The final ranking was derived by comparing the values of synthetic objectives with the corresponding values for reference sets containing the model and satisfactory objects.

The results of the *ex-post* evaluation, as of December 31, 2009, reported until March, 2010, are presented in Tab.4. For reference, in Cols. 9 and 10 we include a comparison with the results of *ex-ante* assessment and scenario expectations presented in Tabs. 1 and 3.

It is to be noted that, despite the deficiencies of the weighting methods, all technical criteria in Tab. 4 were aggregated based on an equal weighting. Thus the overall technical assessment criterion (col. 7 in Tab.4) was a result of rounding and normalising to the scale [0,…10] of the linear combination of the absolute transfer rates (positive weights) with the unreliability of the link expressed by the standard deviation of the down- and uplink rates (with negative weights).

Observe that the highest mean values received for downlink in the site POL05 were accompanied by highest connection risk that reduced the technical score. The main,

TABLE III.
A REVIEW OF PLANNED IMPLEMENTATION OF USERS' SCENARIOS IN POLISH PILOT SITES

No.	**Scenario description**	**Satellite Internet Pilot Sites in Poland**									
		POL 01	POL 02	POL 03	POL 04	POL 05	POL 06	POL 07	POL 08	POL 09	POL 10
1.	Entrepreneurship education: A rural e-shop run by students	N	P	P	N	P	P	N	XP	P	P
2.	Access to education: A virtual music school for rural students	N	P	XP	N	P	XP	N	P	P	P
3.	Students broadcasting local affairs through their own TV programme	XP	P	XP	XP	XP	XP	XP	XP	XP	XP
4.	Better education: Enriching science learning and scientific activity in remote rural areas	XP	P	XP	XP	XP	XP	XP	XP	XP	XP
5.	Rural school teacher training	P	XP	XP	XP	XP	XP	XP	XP	XP	XP
6.	On-the-field personalized communication and training services for farmers	P	P	P	N	XP	P	P	N	P	P
7.	Health emergency training	P	P	D	N	P	D	P	P	P	P
8.	Addressing change and innovation competences in rural communities	XP	P	P	XP	XP	P	XP	XP	P	P
9.	Individual learning (5 scenarios for different social groups)	XP	XP	P	P	XP	XP	P	P	P	P
10.	Improving internet access to public services and the electronic office (e-government)	P	P	P	P	XP	P	P	XP	P	XP
Overall *ex-ante* expectation of the dominant users' scenario		R	IAP	T-EL	R	E-G	IAP	T-EL	T-EL	T-EL	T-EL

The numbering of sites (POL01-POL10) in Tab. 3 is the same as in Tabs. 1 and 2. The other symbols used in Tab.3 are explained below:

XP - e**X**tensive use during the pilot phase highly **P**robable
P - **P**otential or Planned use at a later date
D - to be **D**etermined, depending on a possible expansion of the target group
N – **N**o use expected during the project's duration

E-G – Predominant use for **E**-**G**overnment
R -**R**esearch
IAP- **I**nternet **A**ccess **P**oint
T-EL- On site **T**eaching and **E**-**L**earning platform access

or for most sites even the only reasons of a lower-than-expected use of dedicated applications were lower-than-expected transmission rates, that made the use of some applications more difficult, and the initial lack of the most wanted e-government-oriented learning applications, tailored to Polish circumstances, that was supplied by the National Coordinator in the third year of the project.

Neither the National Coordinator nor the sites themselves had an influence on the technical criteria contained in cols. 3-6 in Tab. 4. The transmission rate values assumed in the project were 4 Mbaud for downlink and 2 Mbaud for uplink, so the values reached actually differ substantially from those initially assumed. Intensity of using Project's dedicated applications by all sites were measured by access time to the specific web pages and supplied by the project's partner responsible for these measurements. These data were normalised for all sites, taking into account the transmission rates at this site and the assumed number of potential users. It is to be noted that the ranking in Tab.4, should be regarded as touching upon only the aspects specified in this paper and it might not reflect the overall assessment of the sites' operation. Therefore the differences between the *ex-ante* (selection-stage) and *ex-post* evaluation ranks contained in col. 10 of Tab. 4 may serve exclusively as an illustration of the evaluation method and not as the final assessment. On the other hand, the values obtained show a good compliance between the expectations, that served as reference values, and actual results.

An additional aspect of the above evaluation resulting from the interdependence between the (*ex-ante*) needs analysis and *ex-post* evaluation of pilot sites, is to justify the rationale for the initial selection of sites. While the technical criteria of needs assessment and evaluation were different, the target goals, i.e. intensity of use, measured as the transmission volume, the average number of users per week and its stability, measured by the standard deviation of the above, as well as the second goal: filling-in the gaps in the broadband coverage, and the third-one: the number of users willing to use the dedicated applications made available within the project, could be compared as *ex-ante* expectations at the stage of selecting the pilot sites, and as *ex-post* evaluation results. To assess each individual site, its final outcomes were adjustted to the potential capacity of this site, rather than measured in absolute numbers. The difference between *ex-ante* expectations at the stage of selecting the pilot sites, and *ex-post* evaluation results, was taken into account, to measure the progress achieved. To assess the entire sites' selection process, the assessment of the overall benefits, resulting from the operation of all sites, can be taken into account as a separate criterion as well.

V. CONCLUSIONS

Based on the data gathered at the satellite internet pilot sites by the National Coordinator for Poland, and the studies carried out afterwards, we can conclude that the *'most probable'* Polish Information Society (IS) development scenario until 2025 and beyond contains a considerable use of satellite internet access in selected remote rural areas, where the duty of the dominating telecom operator to provide a broadband access to all citizens, that is imposed by law, can be

TABLE IV.
RESULTS OF EX-POST TECHNICAL EVALUATION OF THE RURAL WINGS PILOT SITES' OPERATION IN POLAND

No.	Site name and code	Mean dow-link rate (kb)	Standard deviation of downlink (kb)	Mean uplink rate (kb)	Standard deviation of uplink (kb)	Overall technical assessment score [0…10]	Intensity of using dedicated applications	Deviation from ex-ante scenario assignment (Tab.2)	Difference between the ex-post and ex-ante ranks
1.	2	3	4	5	6	7	8	9	10
1.	ZAWOJA, POL01	1031,795	486,723	68,375	38,715	5	lower	none	0
2.	POLANA, POL02	902,946	538,364	N/A	N/A	4	lower	none	-1
3.	WIŚNIOWA, POL03	1133,273	508,9445	44,300	7,328	5	lower	none	-1
4.	MT. LUBOMIR, POL04	720,581	562,458	347,413	304,282	3	lower	later launch	0
5.	ROKICINY PODHALAŃSKIE, POL05	1417,943	1525,783	58,661	49,041	5	standard	intensive e-govt	1
6.	SKAWA, POL06	1344,20	741,87	49,37	16,88	6	lower	none	0
7.	NOWY LUPKOW, POL07	1552,985	350,599	112,845	31,502	8	standard	none	1
8.	KĘTY, POL08	1323,77	665,22	88,01	47,22	7	higher	none	1
9.	MYCZKOWCE, POL09	823,546	467,881	58,091	71,147	4	lower	later launch	-1
10.	HARKABUZ, POL10	922,07	479,05	N/A	N/A	3	standard	later launch	0

fulfilled by the DVB-RCS or a similar bi-directional satellite technology in a most efficient and economical way.

The recommendations and conclusions resulting from the implementation and evaluation of the Rural Wings project in Poland, can be summarised as follows:

- The installation and operation of the satellite internet access points in the DVB-RCS technology overlaps with the middle-term Information Society (IS) policy goals, as specified in the related Polish IS policy documents, and with the statutory duties of dominating telecoms,
- With the forthcoming new generation of satellites, the DVB-RCS internet can provide safe and affordable e-infrastructure with geographically unlimited access,
- The successful operation of the D-Star terminal installations depends on the availability of comprehensible e-government, e-learning platforms, content and services,
- A growing role of e-health-related applications, that have to be made available to everyone, without geographical limits, is to be taken into account when designing the future deployment plans of the DVB-RCS technology,
- The satellite internet access can foster the extension of common intellectual sphere in the e-space and eliminate digital divide in rural areas;
- The development of local information societies in rural areas based on the satellite DVB-RCS technologies will support a move from the traditional agriculture to the development of future innovative, sustainable, environment-friendly agriculture that supplies high-quality end products and uses sophisticated IT, such as robotics, monitoring and control. Similarly, other branches of rural economy, such as tourism, specifically agrotourism, will benefit from the satellite broadband access.

Furthermore, the conclusions resulting from the *ex-post* evaluation of the initial ranking procedure give an insight on the overall evaluation methodology for the satellite internet access points. The *ex-post* evaluation of the initial ranking, based on the comparison of initial ranking and the *ex-post* assessment criteria studied in Sec.IV, may allow to derive a learning scheme, which – in turn – can be useful to assess the adequacy of the selection criteria, the adequacy of choice and correctness of using of the multicriteria outranking procedure applied in the selection process and the credibility of data supplied by the applicants. To make such a comparison possible, the actually observed criteria need to be transformed to the measures of the above goals, taking into account the uncertain character of the initial expectations. The evaluation process of the functioning of ten DVB-RCS pilot sites in Poland allows to derive a subset of relevant needs assessment criteria from a larger set of criteria considered at the site selection phase.

A similar approach to the presented above can also be applied to assess the adequacy of the multicriteria decision-making method applied to derive the final (*ex-post*) pilot sites' ranking. The core of the procedure consists in calculating the distance between the initial and final rankings, which reflects the efforts that would be needed to get the results anticipated, as seen *ex-post*. An aggregated value function, describing the quality of assessment and quality of performance of selected sites can be defined as well. In case of repeated decision problems, its outcomes can be used as inputs for a learning scheme. This approach is universal in the sense that it can be applied to any similar problems, involving an initial ranking, and the performance of *N* selected objects based on this ranking. Specifically, a general methodological framework to select satellite internet access points could be based on the scheme here proposed.

Finally, let us mention that based on the economic forecasts and scenarios for Poland [6], the infrastructural IT satiation in Poland will come between 2014 and 2017 [7], together with the development of commonly accessible e-economy, e-government, e-health, and e-learning applications [8]. This breakthrough will set the high-quality broadband access and common IT literacy among citizens to one of the top policy priorities.

To sum up, there is a considerable potential for satellite-based e-government, e-learning and e-health in Poland, the latter especially for preventive medicine and m-health applications [4]. One can expect that m-health in remote and sparsely populated areas shall be based on the satellite internet, which is regarded as more reliable that the wireless-ground-based connections. It is also worth to mention that e-learning, e-government and e-health-related actions, activities and public funding have been included in the strategic local policy documents in Polish regions concerned, i.e. in Małopolska and Podkarpackie Voivodships, which makes future implementations plans, resulting from the Rural Wings project feasible.

References

[1] Hu, Y., Li, V. O.: Satellite-based Internet: A tutorial, IEEE Communications Magazine 39(3), 154–162 (2001)

[2] Sastri, L., Pahlavan, K. K., Leppänen, P. A.: Broadband satellite communications for Internet access, Springer, 456p. (2004).

[3] Skulimowski A. M. J.: Methods of Multicriteria Decision Support Based on Reference Sets. In: Caballero R., Ruiz F., Steuer R.E. (Eds.), Advances in Multiple Objective and Goal Programming, Lecture Notes in Economics and Mathematical Systems 455, Springer, Berlin-Heidelberg-New York, pp. 282—290 (1997)

[4] Skulimowski A. M. J.: M-Health as a Challenge to the Medical Decision Making System, IPTS Report, No.2 (March 2004), 3–11 [available in English, French, German and Spanish]

[5] Skulimowski A. M. J.: Framing New Member States and Candidate Countries Information Society Insight. In: Compano R. and Pascu C. (Eds.) Prospects for a Knowledge-Based Society in the New Members States and Candidate Countries, Publishing House of the Romanian Academy, pp. 9—51 (2006)

[6] Skulimowski A. M. J.: Future Prospects in Poland: Scenarios for the Development of the Knowledge Society in Poland, ibid., pp. 114–159 (2006)

[7] Skulimowski A. M. J.: Needs and perspectives of telecenters in Europe. In: Sotiriou S., Koulouris P. (Eds.) Bridging the digital divide in rural communities: practical solutions and policies, Athens, Greece, 15–16 May, 2008. Athens, Ellinogermaniki Agogi, pp. 117–-150 (2008)

[8] Tadeusiewicz R.: A need for a scientific reflexion on the development of the Information Society. In: "Społeczeństwo informacyjne 2005", G. Bliźniuk, J.S. Nowak (Eds.), Katowice, PTI Oddział Górnośląski, pp. 11–38 (2005)

ISBN 978-83-60810-27-9
ISSN 1896-7094

INFOMAT-E – public information system for people with sight and hearing dysfunctions

Wojciech Górka
Instytut Technik Innowacyjnych EMAG
Katowice
Email: w.gorka@emag.pl

Adam Piasecki
Instytut Technik Innowacyjnych EMAG
Katowice
Email: a.piasecki@emag.pl

Beata Sitek
Instytut Technik Innowacyjnych EMAG
Katowice
Email: b.sitek@emag.pl

Michał Socha
Instytut Technik Innowacyjnych EMAG
Katowice
Email: m.socha@emag.pl

Abstract*—The article features the results of two initial stages of the Infomat-E project. The project is to provide access to information to people with sight and hearing dysfunctions through a hardware-software solution. So far, a number of analyses have been conducted within the project with respect to the method in which the contents of information is presented as well as interaction with the devices that present this information. These included the analysis of suitable colours, font sizes, ergonomic layout of screen menu bars, and ergonomic keyboards – to make them most convenient for people with sight and hearing dysfunctions. There were also analyses conducted how written texts are understood, especially in the case of the deaf. The project assumes integration of elements which were results of separate research projects. Within the project, the following will be used: speech synthesis, speech analysis, presentation of ideas with the use of the sign language. The project will result in the Infomat-E system which will present information in kiosks specially designed to suit the needs of people with sight and hearing dysfunctions. The article features the results of the conducted analytical works which lay at the basis of the technical concept of the system. This concept is presented in the article too.

I. Introduction

MODERN public administration requires that the customers should have more and more knowledge, especially in the field of administration issues. It is necessary to know regulations, administrative procedures, to be familiar with one's own rights and duties resulting from the national and local laws. To meet the clients' expectations in this field, administration offices provide information in different ways. There are guidebooks, web sites, public information newsletters, information kiosks, and other methods to provide assistance in everyday contacts between citizens and public administration.

Among people who use public administration services there are handicapped people whose dysfunctions cause not only health problems but also communication problems. Among the handicapped, 30% are people with sight dysfunctions, while about 14% – with hearing dysfunctions [1]. As there are about 5.5 million handicapped people in Poland[1], the group of people with sight or hearing dysfunctions is considerably large, and this group is particularly affected by communication problems as far as the access to information or information exchange are concerned. Many institutions lack solutions that would allow the blind and the deaf to deal with administration issues easily and, what is even more important, by themselves. In the case of blind people, there is no information that would contain, for example, a set of guidelines how to move inside a building in order to get to the place where a given issue is dealt with.

While the problems of blind people seem to be regarded as evident, it is often understood that written information is an alternative for the deaf. However, deafness significantly affects the way a deaf person functions in a society, causing difficulties in understanding both written and spoken information and in expressing oneself. This affects particularly those who have been deaf from birth as the Polish language is a foreign one to them – their mother tongue is the sign language which has totally different syntax and grammar regulations and whose terms have different semantic ranges than those characteristic of Polish [2]. Vocabularies are significantly different too as far as the number of words is concerned. Generally speaking, sign languages used in Poland have only several thousand signs each. Information provided by different institutions, not only administration offices, is often too complicated (due to such reasons as: too sophisticated words, too long sentences, the use of compound-complex sentences). Such statements are too difficult to understand not only by a deaf person but also by an average citizen.

As a result of all these factors, people with hearing and sight dysfunctions have become socially excluded as far as the contacts with public administration are concerned.

Currently available devices that support the deaf and the blind in their everyday living are separate for each of these two groups. There are no integrated and, at the same time, publicly available solutions that would support people with each type of dysfunction, not only individual users.

II. Infomat-E project

The researchers of EMAG had recognized the numerous, varied and yet unsolved issues and launched a project aimed

The presented results are the results of the project No N R02 0059 06/2009

[1] National Census 2002.

at solving at least a part of problems of the handicapped. Within the project No N R02 0059 06/2009 EMAG began the development of an integrated set of hardware and software components which will allow efficient information transfer to people with sight and hearing dysfunctions. The objective of the project is to develop a prototype which will make it easier to deal with widely understood public issues. The solution will be based on providing information with the use of specially designed information kiosks. This article aims at presenting the results of analytical works related to the project along with the concept of the Infomat-E system development.

III. Analyses results

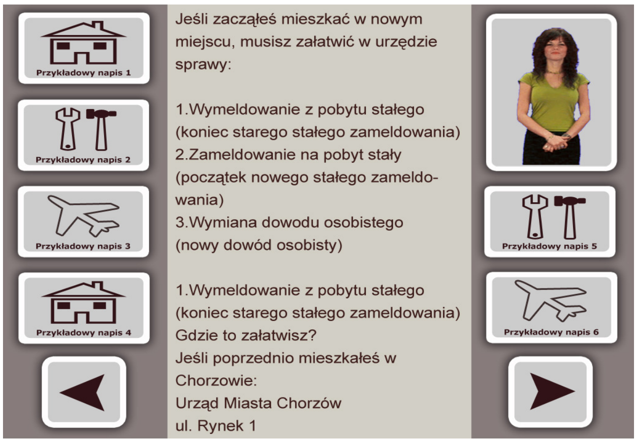
The Infomat-E project is developed in co-operation with two groups of handicapped people, according to the project assumptions. All analyses presented here, along with proposed technical solutions, were consulted with potential future users of the system, i.e. blind, poorly sighted, deaf, and hard-of-hearing people.

A. Requirements about appearance of the device

The Infomat-E project is based, to a large extent, on presenting information to the user. The information is presented in kiosks and the objective of the project is, among others, to make the presentation method and navigation through the system as easy as possible for people with sight and hearing dysfunctions. For this clearly defined objective there were requirements identified and analyzed within the project. The article features a part of these requirements with comments. For blind people it is important that the information should be property voiced and that an ergonomic control method should be applied (voice, well adapted keyboard). For poorly sighted people it is important that the information should be well visible. These issues are contained in WCAG (Web Content Accessibility Guidelines) [3]. This specification refers, largely, to the issues of web site content presentation. An information kiosk is a kind of a specific web site, thus the guidelines included in WCAG were a starting point for writing down the guidelines on how to present information in the kiosk. The layout of information sites was prepared. Then the sites were evaluated by poorly sighted people from the Polish Association of the Blind (PZN). It turned out that some WCAG guidelines were too lenient with respect to the colours or contrast applied in the presentation. A large part of screen layouts of the future Infomat-E presented to the poorly sighted were questioned due to poor readability, though all proposals complied with the WCAG requirements.

As a result of discussions between the project developers and PZN, some additional requirements, supplementary to those of WCAG, were prepared. The requirements about colours and general appearance of the site must help poorly sighted people pay attention to the most important information.

The sites should have high contrast applied to such elements as the text and diagrams. The background of the site should be uniform so as not to distract the poorly sighted person's attention. The site should be visible for the poorly sighted but, at the same time, should have an esthetic appearance for able-bodied people who will use it as well. Due to contrastive colours the site lacks esthetic qualities and, therefore, does not look professional. As a result of analytical works and consultations with the poorly sighted, it was decided that the best colours would be different shades of grey. Such a set of colours (Fig. 1) meets the requirements both about high contrast and esthetic requirements.



Jeśli zacząłeś mieszkać w nowym miejscu, musisz załatwić w urzędzie sprawy:

1.Wymeldowanie z pobytu stałego (koniec starego stałego zameldowania)
2.Zameldowanie na pobyt stały (początek nowego stałego zameldowania)
3.Wymiana dowodu osobistego (nowy dowód osobisty)

1.Wymeldowanie z pobytu stałego (koniec starego stałego zameldowania)
Gdzie to załatwisz?
Jeśli poprzednio mieszkałeś w Chorzowie:
Urząd Miasta Chorzów
ul. Rynek 1

Fig 1. Standard layout of a touch screen [source: own]

The site of the information kiosk should present the content with a contrast ratio of 7:1, according to the pattern described in WCAG, while the font size should be at least 20px[2] , which means that on 96dpi monitors the text will have a minimum height of 6 mm. After the analyses, the minimum recommended height was increased to 10 mm. As far as navigation buttons are concerned, they should be well visible against the background. This can be achieved by contrasting frames. The buttons should be placed in regular distances from one another, in flat rows. It is possible to use shadows around buttons, however each shadow should be symmetrically placed around the button.

In order that the function of the button could be better understood, it is possible to place an icon on it. The icon is also helpful for the deaf as it helps to understand the use of the given option – it creates certain associations. The icon on a button has to be in proper distance from the caption. The icon has to be outlined. Filling of the outlines should be avoided as this way the whole picture would be blurred and the pictogram would be seen as a shapeless blotch. Thus the used icons were quite big while the captions on buttons relatively small. There are certain arguments for such a choice. If big-size fonts were used – readable for poorly sighted people – the place in which to place the pictogram would be much smaller. It is possible to increase the size of buttons but this would cause problems with distributing larger num-

[2] px stands for screen pixel. All calculations of quantities referring to font sizes or other displayed elements have to take into account the real resolution of the monitor. It is necessary to remember that these could be different values for the X and Y axes. In this article the resolution of 96dpi was applied in calculations.

ber of buttons which give access to information contained in the Infomat-E system. Another argument to diminish the size of captions is the fact that one's associations are better developed on the basis of pictures rather than texts. Additionally, smaller captions on buttons will be readable for users who do not belong to two basic groups selected for the project. No matter whether the graphic layout is readable to the poorly sighted or not, those users will have additional support from those functions of Infomat-E which were designed for the blind. The whole of the system will be voice-enabled, including buttons.

B. *Guidelines about text presentation*

The method of text display in the kiosk should be compliant with a number of requirements. The text should be written in a sans-serif font, aligned to the left and should not contain words in italics. The recommendation to use a sans-serif font is the result of discussions with PZN and is different from the recommendations stipulated by WCAG. Sans-serif fonts are particularly clearly seen on raster scan displays, such as a computer monitor. Serif fonts and italics, well readable on printouts, are not displayed well on screens. The displayed text should contain enough blank space (line spacing, inter-paragraph spacing) to distinguish between particular information units. It is necessary to have two spaces after each sentence and a single line of the text should have no more than 80 characters. Interlinear spaces should have at least 2 px, while inter-paragraph spaces should be 1.5 times bigger. All texts displayed in the kiosk should have a uniform appearance, without unnecessary text formatting. The only exception are words that have to be specially distinguished, for example links to other sites. The letters in the words should not be placed too narrowly since they might merge into one another.

C. *Understanding the texts*

Understanding the texts is another important element which affects the kiosk accessibility by the handicapped. In this case the key factor is how information is understood by people who are deaf from birth because they particularly face difficulties while reading standard texts.

The deaf use the sign language every day. This language has a different structure and logic than the Polish language because the latter is based on linear language structures. Each sentence in Polish is a series of successive elements coming in a certain order which cannot appear all at the same time. In the sign language it is possible to use non-linear mechanisms, thus one can say a few things at the same time [2]. It happens so due to communication transferred along three channels at the same time (body language, voice-sound, words). As the deaf lack one of these channels, they reinforce other, i.e. the body language channel. Using body language is a natural thing here because this channel is responsible for the majority of communicated messages. Still, looking at the words transferring channel, in the case of the sign language, in long sentences the SVO[3] order dominates (the subject is before the verb and the object comes as the last one) while in short sentences – the SOV[4] order (the subject is before the object and the verb is the last element).

Another problem is the vocabulary range of the sign language. Currently in Poland the sign language contains, officially, about 2,500 words [4]. This concerns the documented and analyzed version of the sign language used in Poland, i.e. the Sign and Language System (SJM). A larger vocabulary range is the one of the Polish Sign Language (mother tongue of the deaf). However, this range is not contained in any dictionary and there are differences between regions where it is used. Independently on the version of the sign language, its vocabulary is much smaller than that of the Polish language. Therefore, if a text is to be understood by the deaf, it has to be simplified, to make a kind of translation, which is a very difficult task if one takes into account numerous differences in the vocabulary ranges between Polish and the sign language. It is important to note that many hearing people who use the Polish language every day have problems in understanding complicated texts, e.g. official texts, legal texts [5] [6]. Additionally, when official texts are simplified, it is forbidden, due to formal reasons, to translate certain precise expressions and the representatives of public administration are reluctant to allow such translations as some elements, such as the names of documents or departments, should not be modified.

Having in mind these two groups of people and other formal requirement, some recommendations were worked out on which the descriptions displayed in information kiosks should be based:

- The text should not contain long and complex sentences.
- Each sentence should have only one verb in second person singular and the SVO order which is the basic syntactical order of the Polish language and understood by the deaf.
- Passive voice must not be used.
- The texts should contain words only from the set of vocabulary of the sign language. If it is necessary to have in the description an expression incomprehensible to the deaf, the expression should be described by means of basic words.
- It is recommended to use expressions which show how a given activity should be performed with the use of concrete everyday things, e.g. the sentence "Submit an e-form" can be replaced by: "Save your data in the computer on the web site of our office".

D. *Navigation*

For blind people it is particularly important to have proper navigation guidelines which enable them to get to the desired place. After consultations with the blind it was decided that navigation guidelines should contain orientation points, such as: stairs, door, lift. One should not forget about putting a starting point into the guidelines either. It is not necessary to say how long the distance is, it is enough to describe the way and give the destination point or places where direc-

[3] SVO - Subject Verb Object

[4] SOV - Subject Object Verb

tions change. Navigation guidelines should not contain too many details as this can make it difficult for a blind person to remember or comprehend the description. Other users can make use of basic information about the place where a given issue is dealt with, i.e. room number and floor. Additionally, it is planned to have an interactive map that will picture the location of selected places in the building and how to get there.

IV. Concept of the solution

As a device, a multi-media kiosk is a specially prepared computer equipped with a touch screen, keyboard, camera and microphone.

As far as its application is concerned, the kiosk can be defined as a place where it is possible to get context information about things related to the institution in which the kiosk has been installed (office, shop, exhibition, museum, etc.).

People with different dysfunctions will have different problems when contacted with a typical information kiosk. The poorly sighted (or the blind) have problems with using touch screens. Sometimes there are keyboards on touch screens which make the latter useless for the blind. In the case of mechanical keyboards, there are often untypical keyboards used. The keyboard buttons have small key travel (weak tactile feel).

The kiosks also lack screen voicing in the form of the screen content being read aloud. It is possible to use the so called screen reader but, as the interaction with the kiosk is different than with the computer (no mouse), it is necessary to employ extra mechanisms. Using a speaker in the kiosks makes the users feel they have no privacy. People with hearing dysfunctions and deaf usually have difficulties in understanding written texts (the written language is, in fact, a foreign language to them). The best solution would be to present the content with the use of the sign language. Obviously, hard-of-hearing people find it difficult to hear sounds too, particularly when standard speakers are used.

To sum up, the requirements for the Infomat-E system, from the point of view of an information kiosk, are the following: proper voicing (both speech synthesis and analysis), presenting the content in the sign language, providing ergonomic operation of the device (proper receiver, keyboard). Other requirements regarding the information display are the following: suitable layout and content (avoiding too much information), simple texts (comprehensible texts), proper colours and sizes (contrast, brightness, colours).

While the technical concept of the solution was worked out, it was assumed that the solution should enable easy adaption of the existing kiosks. The kiosk structure should not divide the users into groups. There should be a common interface provided that would meet the requirements of all users. These objectives will be achieved through the use of proper software and a dedicated device – a manipulator.

The software will include several components which will facilitate access to particular pieces of information linked with one another so that their co-operation could bring a desirable effect.

The voicing will be achieved by providing speech synthesis in such a way to enable ergonomic listening (possibility of repeating already heard messages, reading sentences, reading words, explaining difficult words). Additionally, voice control will be available as a supplementary function to keyboard and touch screen based control.

To facilitate the system operation by the deaf, there will be an avatar presenting the information content in the sign language. This solution is developed in co-operation with the Silesian Technical University and the Progress company [7] [8]. It is composed of two elements: a module which is responsible for controlling the avatar's movements and a mechanism which translates sentences into the sign language (necessary simplifications, change of word order, etc.). This way it will be possible to present the content in the sign language automatically.

Apart from the software that will support handicapped users, there will be software to support the administration of the whole system. It will comprise; a tool supporting the verification of the content complexity (checking the words used in descriptions so that they should fall within a certain range, e.g. the range of the sign language), a tool verifying the used colours (measuring contrast according to the standards given by WCAG so that the colours should be suitable for the poorly sighted), an administration panel to manage the content and interactions with the kiosk.

Apart from the software, it is expected to have a special device added to the system – a manipulator (Fig. 2).

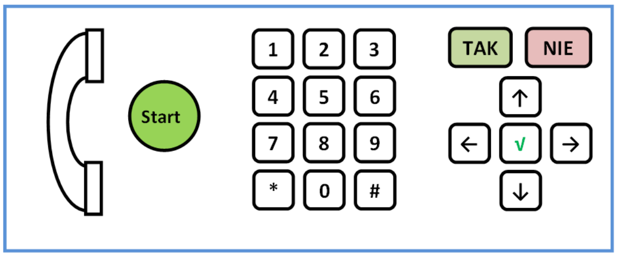


Fig 2. Sample layout of the manipulator panel [source: own]

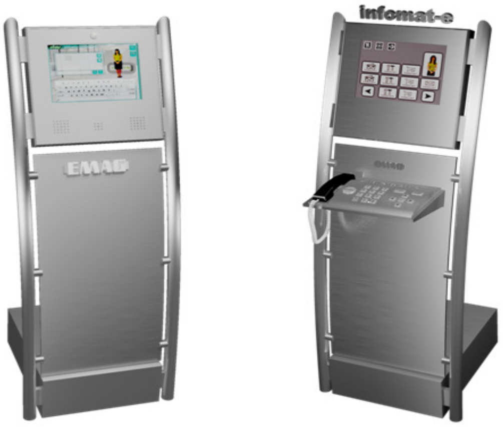


Fig 3. A typical information kiosk and a kiosk of the Infomat-E system [source: own]

The basic element of the manipulator will be a keyboard consisting of two parts: a numerical keyboard and a navigation keyboard (arrows). Both keyboard systems are very popular and commonly used (the numerical keyboard is used as a standard in phones, while the navigation keyboard – mobile phones). The START button will be specially distinguished. Its role will be to return to the initial state of the dialogue or to check whether the kiosk is operable. The keyboard will have clearly marked button travel while the buttons will have convex-print captions. There will be convex prints of symbols and Braille alphabet signs. The manipulator will also have a telephone receiver in order that the user could have some privacy. Additionally, the receiver will enable voice contact (listening and control) and will help people with hearing devices to use the system thanks to a built-in induction loop. There are plans to add a movement sensor to the manipulator so that a person approaching the kiosk could be detected. This will allow to implement certain reactions of the kiosk to such a situation. A Bluetooth module will allow to use the Infomat-E system through a mobile phone. It will be possible to send a note to a mobile phone, information about how to get to a certain place (map of the building), etc. What is more, it will be possible to connect the manipulator to already existing and functioning kiosks (Fig. 3). Thus there are two solutions worth mentioning here. The panel of the manipulator will be connected to the kiosk by means of special handles – this way if the manipulator needs to be adapted to a new type of kiosk, it will not be necessary to re-design it. The second quality which makes the system more universal is the use of a USB connection to connect the manipulator to any computer.

The concept of the Infomat-E system focuses on the part which is responsible for information display – the information kiosk. However, the kiosk is not the only element of the system. The system structure will make it possible to start the system on a single kiosk or on many kiosks operating within a network (Fig. 4).

It is planned that one database with texts will be used for many types of interfaces – the same content and different way of presentation on different devices. Thanks to one coherent management of such content, the quality of information service is increased because the presented data are coherent and updated (a change in one place is followed by changes in all types of interfaces).

V. Conclusion

The project enables to integrate the elements which have been scattered so far and which are results of separate research projects. The developed system will facilitate contacts with administration and other public institutions for a wide group of users, particularly those who have sight or hearing dysfunctions. The adopted concept of the system development allows easy information management within the system. The project contributes to all efforts against social exclusion and is an answer to social needs for such types of systems.

Additionally, the objective of the project is to develop a solution that would allow integration with any information kiosks offered by commercial companies on the market, provided that the kiosks comply with the technical requirements stipulated by the project. The presented technical concept enables to fulfill this objective.

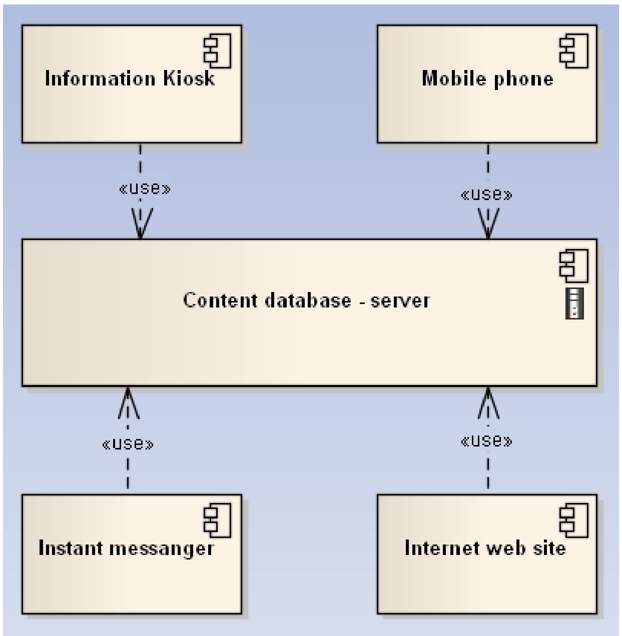


Fig 4. Infomat-E system architecture [source: own]

The results of conducted research and analyses will allow better identification of difficulties encountered by the blind, poorly sighted, deaf and hard-of-hearing. Higher awareness with respect to the existing barriers is sure to initiate another projects aimed at assisting the handicapped in their everyday living, similar to the Infomat-E project. Particularly the results of analytical works related to the presentation methods and content can be helpful in other projects which aim at providing access to information to people with sight or hearing dysfunctions.

References

[1] P. Ciecieląg, B. Lednicki, J. Moskalewicz, M. Piekarzewska, J. Sierosławski, M. Waligórska, A. Zajenkowska-Kozłowska, *Stan zdrowia ludności polski w 2004 r.*, Główny Urząd Statystyczny, Informacje i opracowania statystyczne, Warszawa 2006

[2] M. Mrozik, *“Powiedzieć wszystko na raz”*, Forum Akademickie, Lublin 2006

[3] *Web Content Accessibility Guidelines (WCAG)*, W3C Recommendation 11 December 2008

[4] J. K. Hendzel, *„Słownik polskiego języka miganego”*, Wydawnictwo „Rakiel”, Olsztyn 2000

[5] W. DuBay, *“The Principles of Readability”*, Impact Information Costa Mesa, California, 2004

[6] K. Wolff, „*Wyniki badań Biblioteki Narodowej nad stanem czytelnictwa w Polsce 2008*”, Pracownia Badań Czytelnictwa Instytutu Książki i Czytelnictwa Biblioteka Narodowa, 2009

[7] P. Szmal, N. Suszczanska, „*Tłumaczenie tekstów na język migowy w systemie TGT–1: zasady i realizacja*”, Speech and Language Technology 2002, eds.: G. Demenko, M. Karpiński, K. Jassem, vol. 6, 113-124.

[8] J. Francik, P. Fabian, “*Animating Sign Language in the Real Time*”, 20th IASTED International Multi-Conference Applied Informatics, Innsbruck, Austria, 2002, pp. 276-281.

Proceedings of the International Multiconference on Computer Science and Information Technology pp. 599–603
ISBN 978-83-60810-27-9
ISSN 1896-7094

Bidirectional voting and continuous voting concepts as possible use of Internet in democratic voting process

Jacek Wachowicz
Gdańsk University of Technology,
ul. G. Narutowicza 11/12,
80-233 Gdańsk Poland
Email:
jacek.wachowicz@zie.pg.gda.pl

***Abstract*—Democracies need elections for choosing their authorities and governments. This process has many factors that shape today's procedures. However, the Internet is a medium that may change the possibilities and elections. The main issue is concern on how changes may influence the whole democratic process. This paper shows two possible ideas – that of bidirectional voting and continuous voting, and considers possible reasons for introducing changes as well as the consequences. An introductory research in into this matter gives additional hints.**

I. Introduction

THE INTERNET changes almost everything. It introduced powerful digital goods markets, it empowered the role of information as a strategic good. In this paper the author shows the considerations on how it may affect an election process in order to make it more efficient. For discovering that, it is necessary to bring some thoughts on the politicians' motivations on elections - which seems to be somehow parallel to the process of raising children by its parents. Of course this should be said in context of the aims of the whole election process. Therefore this paper consists of a discussion on the voting process (including Internet voting), brought in chapter 2, followed by an analysis of the politicians' motivations. Then two solutions are proposed that may increase an election effectiveness – namely the bidirectional voting and continuous voting concepts. Those theoretical considerations should be followed by a research on voters' opinions, which is presented in chapter 6. Last chapter presents the author's conclusions.

II. Voting process and Internet voting

To propose any changes in electoral systems we need to think about their meaning, function and purpose. In Book II, Chapter 2 of his book 'The Spirit of Laws', Montesquieu states that in the case of elections in either a republic or a democracy, voters alternate between being the rulers of the country and being the subjects of the government. By the act of voting, the people operate in a sovereign (or ruling) capacity, acting as "masters" to select their government's "come." [6]

In more practical way we may read in Encyclopedia Britannica that regular elections serve to hold leaders accountable for their performance and permit an exchange of influence between the governors and the governed. The availability of alternatives is a necessary condition. [7]

We need to remember, that due to fact that people may vote against ruling government, for having effective elections their organization must safeguard the privacy of voting – understood as the impossibility to trace who voted how and as much as possible, the impossibility to influence on given votes. It is important, while providing election officials with an audit trail that can be used to conduct recounts of election results. [2] Moreover, we need to remember that the whole process needs to gain public acceptance and trust. So, one basic precondition for e-elections must be the feasibility of implementing the voting under such conditions that the principles underpinning the electoral system are not disregarded. Accordingly, the system must be at least as secure as corresponding traditional voting procedures. Another precondition is that the e-voting procedure must be simple and function smoothly for the voters. Its overall purpose is to enhance accessibility to voters. [10]

Therefore an electronic voting system via the Internet must fulfill the following basic requirements according to its trustworthiness and legitimacy [10]:

- Only people eligible to vote should be able to vote (identification).
- It should be possible to use one's vote only once .
- Ballots should be absolutely secret.
- It should not be possible for a vote cast to be changed by anyone else.
- The system should ensure correct tallying of votes at all levels (voting district, constituency and area).

We can count some opportunities to electronic voting [8]:

- Most countries believe that Internet voting will occur within the next decade.
- Internet voting options satisfy voter's desire for convenience.
- Internet voting can meet the voting needs of the physically disabled.
- Several countries are ready to try Internet voting for a small application immediately.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

- Several countries are contemplating voting system replacement and are frustrated with the limited number of options available.

But such system would face many barriers as well – like the difficulty of guaranteeing ballot secrecy with an absolutely certain guarantee. Another is a the question of the reliability of the system, i.e. that the system will in all situations function in the manner in which it is meant to function. Another disadvantage is the expense of development and operation. All in all, then, the primary considerations are security and reliability. [10]

Some others may be [8]:

- Lack of common voting system standards across nations.
- Time and difficulty of changing national election laws.
- Time and cost of certifying a voting system.
- Security and reliability of electronic voting.
- Equal access to Internet voting for all socioeconomic groups.
- Difficulty of training election judges on a new system.
- Political risk associated with trying a new voting system.
- Need for security and election experts.

III. Are Politicians Like Kids?

Now we can think about the motivations that drive politicians taking part in elections. We may understand their goals idealistically, ie. that they are trying to serve country and/or community. But serving people requires getting into power – in words; winning elections. Unfortunately, common observations make most of people to doubt such motivation, making many people believe that their most important aim is to get into power using the program as a tool rather than the goal itself. Of course, it shouldn't be generalized – especially that winning elections almost always requires many alliances with factions fractions that may sometimes have very distant views on selected topics.

However, one may try to find similarities between the election process and raising kids, when they grow. They both are observing responses from the surrounding world (politicians through pools and children through observation of their parents' and other persons' reactions). So by giving them a feedback information, giving hints to what is good and what is bad. They both (politicians and kids) try to behave in a way that maximizes their goals. In the case of politicians of course the aim is to get as many votes as possible.

Children interact with parents constantly – and parents may react in different ways. So let us now consider models of the process of raising kids. Usually we may distinguish two contradictory models and some located in between. We shall discuss only the extreme ones. The First one is a very strict model of raising kids (where children have a lot of restrictions and are punished when they do wrong, but ideally a lot of sensitivity and appreciation when they do good as well). Second one is so called 'stress-less' raising (where children are positively motivated and allowed many things that would be banned by strict parents – which means that they are rarely or never punished, ideally with as much positive motivation as possible and as little punishment as possible). It is Commonly is known, that in first case children will probably care more about others' people needs, but normally will be more backward behaving, they normally shall think about consequences before they do something. In the stress-less case children will be probably more open, active and sometimes even pushing the limits in search what they can. Having less fear they will experiment more willingly even without deeper considerations.

More deep considerations may be found in articles by Diana Baumrind [3], [4] who found what she considered to be the four basic elements that could help shape successful parenting.

Table 1:
Parenting styles [5]

		Strictness/supervision	
		High	**Low**
Acceptance/ Involvement	**High**	Authoritative	Indulgent
	Low	Authoritarian	Neglectful

Such descriptions would be reasonable as well for politicians' deeds. But we need to find two basic distinctions between kids and politicians.

First one is that raising kids is a continuous process – and elections happen (normally nowadays) once every 3-5 years (depending on country and subject of election). During the period in between elections politicians are not effectively rated - in terms of bouncing out-of the position.

The second difference is that in raising kids we may see two models (discussed earlier – strict with negative and positive incentives and stressless – with mostly positive incentives), whilst on elections basically we have only positive votes (which is positive incentive) and after elections politicians are normally safe (until next elections), which may be perceived just as lack of any incentives.

This may lead to conclusion, that lack of negative incentives pushes politicians to strategies of freely pushing limits in search what they can (regardless of true intentions) for getting into power (almost), which is reminiscent of the stressless way of raising kids. Of course in the case of politicians, we'd willingly see responsible, forward-looking persons serving country and regional needs. However achieving this may require (just like in a the process of raising children) adding some negative incentives as well as continuous feedback.

IV. Bidirectional voting

Negative voting (as of intentions) is said to occur when circumstances are unfavorable to the interests or preferences of constituents and evoke a stronger electoral response than comparable favorable circumstances evoke. [11]

The idea of bidirectional voting means giving voters possibility of having two votes – one positive (as usual) and one negative (new). Each voter would have ability to decide which votes to give to whom (and if any – voter would have a chance to give only one – either positive or negative vote). That idea is sometimes called 'negative voting' (in terms of votes), which seems to be miss-named to the author – while negative voting should be exactly opposite to the traditional way, which may be called 'positive voting', as voters have only positive votes. Thus negative voting should mean eliminating candidates (instead of choosing as it is now), which may lead to choosing people and parties without any ideas.

The idea of giving both positive and negative votes would bring a new possibility of expressing disagreement with politicians. One of possible ways of dealing with such votes, could be that negative votes would have to be subtracted from positive ones. This might enforce more responsibility on politicians, as their false or freaky ideas would decrease their election chances, especially in case if it would be required a quite straightforward requirement to gain more positive votes than negative ones (either or both in case of whole parties and in case of each candidate).

It has to be pointed that such a system would bring the possible risk of almost randomly chosen parties – for instance in the theoretical situations when

- a party 'A' would gain
  49% of positive votes and 48% negative votes,
- a party 'B' would gain
  47% of positive votes and 48% negative votes,
- a party 'C' would gain
  3% of positive votes and 1% of negative votes,
- a party 'D' would gain
  1% of positive votes and 3% of negative votes,

a party 'D' would gain 1% of positive votes and 3% of negative votes - the winners would be parties 'C' and 'A' with 2% and 1% of votes net, which recalculates to 67% and 33% and gives a majority to extremely small party, which gained only little votes. However, this problem might be reduced by keeping the normally existing thresholds of for instance 5% of all positive-only votes (which in example above would eliminate tiny parties 'C' and 'D') .

Furthermore it is worth mentioning, that in case of possible (not mandatory) giving positive and/or negative votes, a much more probable situation then mentioned above would be gaining by all parties more negative votes than positive ones. This would drive into the situation, when no-one could be elected, which would enforce the need for another election. However, this would give a possibility to show, that current politicians have lost connection with real-life (which is quite often regarded not to be very rare in politics). Such situation would require then organization of next elections in a short period of time, but would show, that radical changes in the politicians' programs are deeply expected and required.

V. Continuous voting

It is possible to figure out probable reasons why elections are hold every 3-5 years. Quite straightforward one of them seems to be of economical nature – elections preparation and conducting does cost quite a lot. So it wouldn't be sensible to organize it too often. Similarly, chosen politicians do have to have a reasonable time for bringing their programs into life (which normally is a huge task concerning a scale of a whole state). Moreover, politicians on the election year tend to promise and do almost only things that might be perceives as popular. On the other hand, organizing elections too rarely would limit possibilities of changing politicians in case of poor or ineffective leading state (or local) affairs.

Things seem to be different in case of an Internet voting. In this case it is possible to organize it far cheaper, which may give a chance for organizing votings much more often. But we need to think about possible reasons for more often votings – just as at the beginning of this chapter were given reasons for not doing it too often. That implies, that it shouldn't play role of simply more frequent elections as we know them nowadays. But, in chapter 4 (are politicians like kids?) we reached a conclusion, that a form of an feedback seems to be very important in controlling politicians. And this is the field, where continuous voting may seem to be very valuable. Furthermore, in the same chapter 4 we pointed, that a form of negative motivation usually helps in balancing behaviours, which may be very valuable for making politicians' work more for the society than nowadays it seems to be on average. For instance, voting for the worst politician (in terms of chosen in elections – like deputy, parliament member etc.) for having such deputy somehow punished may have a chance to be very positive in terms of quality of their work for the society. Such punishment should be serious one – probably most perceptible and most fair punishment for a deputy that had been chosen the worst one would be taking back such person's seat (mandate). Such continuous, serious threat would force taking into consideration voter's perception of their actions. This might constantly remind the reason why they were chosen.

To achieve same effect at parties level – such seat might stay empty until next elections, which would weaken party of the worst deputy.

On the other hand very important seems to be the positive motivation. So just like choosing the worst deputy as well the best one might be chosen. What might be a serious reward for the best deputy? Probably mandate (seat) in next elections (meaning without taking part in elections). Would it be unfair? Rather no, as this comes from people's voting, just like during elections.

Next we should consider how often such votings should be organized. Probably quite often, as it is known from the pedagogy and psychology, that the punishment (or reward) needs to be immediate. On the other hand, a threat of facing only very little votes given (which would make it meaningless) and on the other hand facing too many deputies left in negative voting (or safeguarded in positive voting) their seats shows that such votings shouldn't be organized too often as well. In author's opinion the right scale for this process would be potential elimination of 3 to 5% (maximum of about 10%) of deputies as well as rewarding the same quota. In such case, a feasible frequency for Poland would seem to be for instance once every 3 months, which would cause potential threat (and opportunity) for maximum 15 deputies during 4 years period. It is worth mentioning that bidirectional voting in such case would additionally give chance to people to say 'no one deserves punishment (reward)' by not giving negative (positive) votes, which is good, because in fact it shouldn't be obligatory to punish (or reward) someone every time.

Thinking about a punishment (reward) we need to point another problem – in case that there is only minimal number of people taking part in such voting – let's say only 10 persons gave their votes. Of course in such case it would not be fair to regard such a voting to be binding one. But this problem is easy to eliminate – it may be added a requirement that for instance at least 5% of voters has to vote for regarding such voting to be a valid one. Additionally for safeguarding controversial, but fair and effective deputies, who may receive a lot of both positive and negative votes – it may be required that for punishing (rewarding) of such controversial deputy would be necessary to reach difference of not less than 3% of all given votes between positive and negative votes.

VI. Research

The research was conducted on 68 students of Gdańsk University of Technology, in April 2010. It was devoted to search main reasons of voting and possible ways of making it more effective.

To understand the outcomes, we should realize that participation in elections in Poland is not compulsory, which results in frequencies below 50%. Moreover, voters are presented ballots with names of parties and candidates of these parties blocked together. Voters are choosing only one person, giving their vote to this person and its party at the same time. To enter the parliament or a local authority, a party needs to gain at least 5% of valid votes. Then, a number of mandates for each party is calculated and seats are allocated to candidates that won relatively most votes.

The first question of research was „which statement presents best your motivation for giving a vote to the candidate you voted for during last elections (and for going to elections at all)". The most important motivations were "I didn't support any of parties, so I voted choosing least evil option" (19,1%), "I didn't took part in elections because I saw no sense in voting (or as a private protest)" (17,6%), "I support both the party and the person I voted for" (17,6%), "I know and support the party and I chose a more or less random person from the party" (16,2%), "I didn't go for elections because there was no one worth voting for" (11,8%).

It seems to be worrisome, that three out of five most important motivations, chosen by a total of 48,5% of respondents, were negative (1st, 2nd and 5th), whilst voting is meant to be primarily positive choice process.

In next question respondents were asked "how long before elections have you decided on whom to vote". The most frequent answers were "a couple of days" (32,4%), "just before elections" (22,1%), "a couple of months" (17,6%), a couple of weeks" (13,2%).

These answers do not really seem to be constructive as well, as they show that voters don't have stable views – as a couple of days before elections about 54,1% of them didn't know whom to vote for. Situation is even worse a couple of weeks before elections, ranging about 67,3% of undecided voters.

Another question asked was an attitude for the bidirectional voting concept. It was formulated as a general question "If it would be possible to give a vote 'against' apart from vote 'for', would you:" followed by five statements:

- "use such possibility":
  'definitely yes' 54,4%, rather yes '32,4%',
  'rather no' 10,3%, 'definitely no' 1,5%
- "give both votes":
  'definitely yes' 38,2%, rather yes '45,6%',
  'rather no' 11,8%, 'definitely no' 2,9%
- "feel that it would be easier to make a decision":
  'definitely yes' 27,9%, rather yes '36,8%',
  'rather no' 29,4%, 'definitely no' 4,4%
- "feel that you can more completely show your views":
  'definitely yes' 50,0%, rather yes '41,2%',
  'rather no' 4,4%, 'definitely no' 2,9%
- "consider it to be fair":
  'definitely yes' 41,2%, rather yes '45,6%',
  'rather no' 7,4%, 'definitely no' 4,4%

This shows, that young, educated voters would welcome a new option of bidirectional voting - voting 'against' as a complimentary to voting 'for'. What is additionally important, among respondents who in the first question („which statement presents best your motivation for giving a vote to a candidate you voted for during last elections") pointed most frequent answer "I didn't support any of parties, so I voted choosing least evil option" 100% chosen that they would like to use possibility of giving negative vote. Moreover, of these who pointed answer "I didn't do for elections because I saw no sense in voting" 75% chosen that they would like to use possibility of giving negative vote. And among those, who pointed answer "I didn't go for elections because there was no one worth voting for" 62,5% chosen that they would like to use possibility of giving negative vote. This shows, that introducing bidirectional voting may help people to show their preferences in a better way, which should drive to potentially higher level of participation in elections.

Last question was asking about attitude towards voting through Internet in terms:

- "I think that voting through Internet should be available"
  'definitely yes' 61,8%, rather yes '23,5%',
  'rather no' 10,3%, 'definitely no' 4,4%
- "I would willingly give vote through Internet"
  'definitely yes' 58,8%, 'rather yes' 23,5%,
  'rather no' 14,7%, 'definitely no' 1,5%

This shows that young, educated people would welcome possibility of voting through Internet, which technologically would allow bidirectional voting, as it was described earlier.

VII. Conclusions

As we may see, Internet use in elections may help introduce new ideas on how to vote for having politicians to be more motivated for better taking care of common issues. The idea of bidirectional voting may help people to show their preferences in a better way, which may drive to a potentially higher level of participation in elections. On the other hand introduction of the continuous voting (which due to costs wouldn't be possible without Internet use) may introduce better control over politicians' deeds by continuous grading, which because of immediate reaction normally is most effective way of controlling of all recent deputy initiatives. This reflects that bringing these ideas into life may help election process to achieve a new, better quality and give a chance to improve democratic processes.

References

[1] "A Report on the Feasibility of Internet Voting", California Secretary of State Bill Jones, California Internet Voting Task Force, January 2000

[2] Baumrind, D., "Current patterns of parental authority". Developmental Psychology, 1971.

[3] Baumrind, D., "Parental disciplinary patterns and social competence in children". Youth and Society, 9, 1978.

[4] Chan T. W., Koo A., "Parenting Style and Youth Outcomes in the UK", European Sociological Review, Oxford University Press

[5] Election. Wikipedia definition. http://en.wikipedia.org/wiki/Election

[6] Election. Encyclopedia Britanicca definition. http://www.britannica.com/EBchecked/topic/182308/election

[7] Gritzalis D., "Secure Electronic Voting. New trends, new threats..." 7th Computer Security Incidents Response Teams Workshop, Syros, Greece, September 2002, Athens University of Economics and Business & Data Protection Commission of Greece, http://www.terena.org/activities/tf-csirt/meeting7/gritzalis-electronic-voting.pdf

[8] Herrnson P. S., Abbe O. G., Francia P. L., Bederson B. B., Lee B., Sherman R. M., Conrad F., Niemi R. G., Traugott M., "Early Appraisals of Electronic Voting", http://www.capc.umd.edu/rpts/md_evote_ContempVotingMach.pdf

[9] "Internet voting", Technology and Administration in Election Procedure, Final Report from the Election Technique 2000 Commission, Stockholm, Swedish Government Official Reports, SOU 2000:125 (Ministry of Justice) http://www.governments-online.org/documents/InternetVotingSweden.pdf,

[10] Morris P.F., Kenneth A.S., "Is negative voting an artifact?" http://www.stanford.edu/~mfiorina/Fiorina%20Web%20Files/NegVoting.pdf

[11] Williamson, J. , "Negative Voting in Presidential Elections" Paper presented at the annual meeting of the The Midwest Political Science Association, Palmer House Hilton, Chicago, Illinois Online, http://www.allacademic.com/meta/p138434_index.html

ISBN 978-83-60810-27-9
ISSN 1896-7094

The Double Jeopardy Phenomenon and the Electronic Distribution of Information

Urszula Świerczyńska-Kaczor
The Jan Kochanowski University
of Humanities and Sciences
Żeromskiego 5, 25-369 Kielce,
Poland
Email: swierczynska@ujk.edu.pl

Artur Borcuch
The Jan Kochanowski University
of Humanities and Sciences
Żeromskiego 5 25-369 Kielce,
Poland
Email:art123321@poczta.onet.pl

Paweł Kossecki
The Polish National Film,
Television and Theater School
ul. Targowa 61/63; 90-323 Łódź
Email:kossecki@poczta.onet.pl

***Abstract*—The aim of this paper is to attract attention to the double jeopardy phenomenon. Double jeopardy seems to very often go unnoticed by companies while they look for an explanation as to why their efforts to enhance the intensity of brand usage are unsuccessful. The clue is that the companies do not pay enough attention to raising the market share. Our discussion in this paper refers to informational websites. Our aim is not to form a final conclusion as to whether there is a double jeopardy phenomenon or not on this particular market. Instead, the conclusion is reached that although the double jeopardy pattern can be observed on the virtual market, the nature of virtual markets can oppose this phenomenon.**

I. Introduction

In the early 60s when Whilliam PcPhee examined the double jeopardy phenomenon (DJ), he might not have expected how universal his idea would be. Nowadays, double jeopardy is observed not only on the traditional market, but on the virtual market as well. Although on the virtual market the double jeopardy phenomenon is competing with the famous 'long tail' theory (the internet enhances the sale of small brands), the Internet distribution does not seem to protect 'small brands from losing twice' [1]. This means that brands on both the Internet and as traditional markets follow the double jeopardy phenomenon – small brands not only attract fewer customers, but these brands are less preferable.

Based on the previous research cited in literature we suspected that our analyses of the usage of the different websites would bring more evidence to suggest the breaking of marketers' 'double jeopardy myopia'. Marketers often do not like the idea of double jeopardy, as they tend to believe that marketing tactics can easily enhance the brand usage. We thought that our conclusion would be that "[...] long-term growth is seen to be a function of penetration not of increased repeat-purchase. This fits the Double Jeopardy pattern but not perhaps the common conception of brand loyalty among practitioners" [2]. But our analyses do not show clear picture of the double jeopardy phenomenon among different categories of websites. Our findings confirm that although the websites are visited by internet users according to the double jeopardy rule, there are many exceptions.

Although the double jeopardy phenomenon has been well-known for over half of century, few studies can be found in marketing literature. On the traditional market the consumers' behaviour during the process of purchasing their coffee, newspapers, soap or breakfast flakes is consistent with the double jeopardy phenomenon [5]. Furthermore, one of the latest publications is the work of C D A Graham [3] who presented a survey based on data gathered from 4,000 of UK households. The results confirmed the double jeopardy phenomenon and the fact that market equilibrium extends into the long term - "[...] exceptional, permanent, structural change in share (i.e. more than 6 points in as many years) does appear to have a common causality. It is strategic, achieved not through everyday manipulation of the promotional mix, but by a major change to brand architecture, or some discontinuous innovations" [4]. A. Elberse [6] pointed out the double jeopardy phenomenon on the virtual market. Her findings show that although as a medium the Internet enlarges the assortment of informational products (this means that according to long tail theory the tail is going to lengthen), the obscure products which form the end of the 'tail' still find few customers. The tail is going to be longer and flatter, not necessary bulked up. This fits with the double jeopardy phenomenon – less popular brands lose 'twice'. On the virtual market the double jeopardy phenomenon was also observed on Polish informational portals [7].

II. The analysis of Business and Informational Polish Portals

In order to examine how double jeopardy affects brands on the virtual market we conducted analyses of three different categories of websites: business and informational portals of Poland and additionally, in the next chapter - social networking sites in the United States.

First, we examined the two categories of Polish portals: business portals (category business, finance and law) and informational portals. The data were taken from the Megapanel PBI/Gemius survey from February 2010. For business portals the frequency of purchase refers to the frequency of visiting the website by the user. Therefore the variable 'views per user' reflects the process of 'buying' information from a particular portal. Trying to explain the double jeopardy phenomenon we use so called model 'w(1-b)'. This model takes into account three variables [8]:

- b_x represents the proportion of customers who buy at least once brand X during the analysed period
- w_x is the frequency of purchasing brand X during the analysed period

- the frequency of purchasing the brand X can be estimated according to the equation that $w_x = w_o/(1-b_x)$, in which w_o is the constant. W_o – is calculated as an average of the $w_x(1- b_x)$ for the all analysed brands.

If the double jeopardy phenomenon can be observed within the data about the reach, users and views for business and informational portals it would mean that the biggest portals (with furthest reach) should also have the highest numbers views. The empirical statistics seem to confirm this pattern. As we can see in Table I and Table II, the higher the reach of the portal (column 4), the higher the number of visits per 'average' internet users (column 5). The correlation between variable 'reach' and 'views per user' is strong:
- for business portals the correlation is 0,73 (R Spearman, p<,05)
- for informational portals the correlation is 0,75 (R Spearman, p<,05).

We can also apply the model of 'w(1-b)' to both analyses allowing us to calculate the expected numbers of visits on the basis of the reach of a particular portal. In this case, the model helps to answer the question: "Taking into account the reach of the portal, does the number of views fit to the reach?" Referring to the example, if the observed usage of a portal is similar to the number calculated using the 'w(1-b)' model, it means that the brand follows the pattern of double jeopardy. The problem for marketers starts when the usage of the brand is lower than calculated. This means, that the brand could be more intensely used, but some factors (unsuccessful campaign, negative PR) contributed to lowering the intensity of usage.

Analyses of Polish business and informational portals seem to confirm that 'small brands lose twice' – not only do fewer users see them, they are also used less intensively (the constant wo is 8.8 for business category and for informational portals wo constant is 16.7). However, we may observe that within both categories there are some brands which go 'against the double jeopardy phenomenon': 1) within the business category e.g. Group Wirtualna Polska– business, Group Bankier.pl or Group Interia.pl – business and 2) within the category informational portals e.g. Wirualna Polska – information, TVN – information or Group Polskie Radio.

III. The analysis of US social networks sites

Let us also analyze the correlation between market share of social networks sites and the activity of their users [Table III]. It should be noted that 'social software' has emerged as a driving force of Web 2.0. The term Web 2.0 was coined by Tim O'Reilly in 2005, to describe a sea of changes in web services and technologies. Overall, there is an increasing presence of social software applications that allow users to communicate, collaborate, and share their personal interests [9]. We examined the data from the five major social networks sites based in the United States: Facebook, MySpace, Tagged, Twitter, myYearbook [10]. If the double jeopardy phenomenon exists on this market, it would mean that users of the largest social network sites are more active than those on the smaller ones. The data only partly confirms this prediction. The changes in the market partly reflect the double jeopardy phenomenon – it means that when MySpace's [11] share of the market decreased,

Table I.
Polish business portals in February 2010. The implementation of the model 'w(1-b)' [14]

	1	2	3	4	5	6
		real users	views	reach (bx)	views per user – observed (views/real users – $w_{x)}$	estimated views per user – model w(1-b_x) (wo(1-bx)
1	Group Onet.pl	3 680 854	54 476 716	0,21	14,8	11,1
2	Group Money.pl	3 230 031	33 080 782	0,19	10,2	10,8
3	Group Infor	2 592 388	24 926 902	0,15	9,6	10,3
4	Group Wirtualna Polska	2 401 477	38 218 983	0,14	15,9	10,2
5	Group Bankier.pl	2 197 162	40 564 897	0,13	18,5	10,0
6	Group Gazeta.pl	1 773 374	17 713 166	0,10	10,0	9,8
7	Group Interia.pl	1 563 283	22 300 749	0,09	14,3	9,6
8	eGospodarka.pl -	1 448 256	8 139 142	0,08	5,6	9,6
9	forumprawne.org	839 639	-	0,05	-	9,2
10	Group Wolters Kluwer Polska -	759 682	4 034 215	0,04	5,3	9,2
11	Group o2.pl -	709 703	5 092 051	0,04	7,2	9,1
12	nf.pl	629 810	4 014 680	0,04	6,4	9,1
13	m2	515 555	2 333 250	0,03	4,5	9,0
14	Group Inwestycje.pl -	510 190	3 330 369	0,03	6,5	9,0

TABLE II.
POLISH INFORMATIONAL PORTALS IN FEBRUARY 2010. THE IMPLEMENTATION OF THE MODEL W(1-B) [15]

		real users	views	reach (b_x)	views per user – observed (views/real users – w_x)	estimated views per user – model w(1-b_x)
1	Group Onet.pl	6 044 938	226 955 389	0,35	37,5	25,5
2	Grupa Gazeta.pl	5 620 848	133 678 936	0,32	23,8	24,6
3	Group Wirtualna Polska	4 813 300	194 257 130	0,28	40,4	23,0
4	Group Interia.pl	3 170 747	69 062 585	0,18	21,8	20,4
5	Group Media Regionalne	2 934 016	78 030 484	0,17	26,6	20,0
6	Grupa Polskapresse - Informacje i publicystyka	2 897 029	35 010 910	0,17	12,1	20,0
7	Group TVN	2 067 830	65 740 032	0,12	31,8	18,9
8	Group Axel Springer	1 336 547	31 593 869	0,08	23,6	18,1
9	Group Presspublica	1 271 260	18 991 842	0,07	14,9	18,0
10	Group Infor - Informacje	1 237 180	22 712 598	0,07	18,4	17,9
11	Group TVP	1 099 680	11 961 268	0,06	10,9	17,8
12	Group o2.pl	1 074 275	17 495 750	0,06	16,3	17,8
13	se.pl	1 041 431	8 892 234	0,06	8,5	17,7
14	Group Google	948 809	-	0,05	-	17,6
15	trojmiasto.pl	707 141	-	0,04	-	17,4
16	Group Gery.pl	644 121	5 823 700	0,04	9,0	17,3
17	Group Polskie Radio	582 316	7 442 190	0,03	12,8	17,2
18	eioba.pl	505 441	-	0,03	-	17,2
19	pogodynka.pl	409 554	7 063 355	0,02	17,2	17,1
20	teleman.pl	401 147	4 665 509	0,02	11,6	17,1

the average time which users spent on this social networks site was also reduces. A similar correlation can be seen in an analysis of Facebook [12] – Facebook grew significantly which lengthened the time per user. Data regarding Twitter, however, runs contrary to the double jeopardy phenomenon: the average time spent on Twitter shortens as its market share rises.

TABLE III.
AVERAGE TIME SPEND ON THE TOP FIVE SOCIAL NETWORKING SITES (RANKED BY MARKET SHARE) AND THEIR MARKET SHARE AMONG US INTERNET USERS, SEPTEMBER 2008&2009 [16]

	September 2009		September 2008	
	market share	minutes:seconds	market share	minutes:seconds
Facebook	58,59	23:00	19,94	18:38
MySpace	30,26	25:56	66,84	29:37
Tagged	2,38	25:17	1,62	23:31
Twitter	1,84	15:52	0,15	36:27
myYearbook	1,05	18:07	1,76	26:12

IV. Looking for Explanation

The presented data partly confirms the double jeopardy phenomenon. The larger virtual portals (ranked by reach) are more intensively used (more views per user or longer time spend on the portal). However there are also smaller portals which do not lose 'twice' – they have similar level of usage as bigger ones. Why? We would like to point out a few possible explanations.

Firstly we wonder how far we can compare 'buying' by spending time on informational portals to buying other goods such as coffee, soaps. In a traditional or internet shop the customer can see different competitive brands and decide what to buy. Under image theory customers' behaviour in this situation is based on the pattern that "[...] most choices are actually choices *not* to do something, in other words, *rejections*." [13]. It means that it is very likely that 'big' brands (which are familiar for customer) simply 'win' the process of selection and this way the pattern of the double jeopardy is enhanced. In the process of choosing informational or business portals the situation is different. Although selected portal can be easily changed to another, because it does not directly require spending money, we should notice that the usage of websites is connected with time, convenience and psychological costs. So we may take into consideration that currency is in this case not money, but time (the 'budget' of time which a user poses and is willing to spend on websites is somehow limited) or convenience (the internet user can be attached to the portal, because they think: 'I'm familiar with navigating this web site', 'I have a mail box on this portal, which makes it easier to use business section on this site'). At the same time, it must be remembered that the internet user can easily read two or more different portals more or less simultaneously. The ease of using many different brands by a user at the same time (in this case portals) means that the double jeopardy pattern is somehow 'diluted'.

The case of social network sites seems somehow different. The value of a social network site increases when more users are within the social network. The effect of 'closeness' appears – the more users have their profiles on a social networks site, the more valuable the site seems to be. Therefore the effect of the double jeopardy phenomenon should (theoretically) be even more visible. Partly, we observed the effect – as Facebook enlarges its share of the market, the time spent visiting it grows. But we analyzed the data from very short period of time. This means that every change, even a temporary one, is reflected in the analyses presented and we are far from saying that the double jeopardy phenomenon has been proved.

To conclude, we observe the pattern of the double jeopardy phenomenon among users of websites. Smaller portals can lose 'twice' – meaning that if the portal attracts less customers, the customers are less willing to spend more time on it and visit less often. Therefore 'the more the better'- the more users the website pulls in, the more time they will spend on it and the more actively they will use it. However, according to our analyses, the reach of the portal is not everything. Smaller portals can enhance the level of activity of their users and break the double 'losing' (e.g. delivering unique utility and features, targeted to niche).

References

[1] A. Elberse (2008), *Should You Invest in the Long Tail?,* Harvard Business Review, July-August 2008, 88-96

[2] C D A Graham, *What's the Point of Marketing Anyway?* The Prevalence, Temporal Extent and Implications of Long-Term Market Share Equilibrium, Journal of Marketing Management 2009, Vol. 25, No 9-10, 867-874, 872

[3] C D A Graham, *What's the Point of Marketing Anyway?* The Prevalence, Temporal Extent and Implications of Long-Term Market Share Equilibrium, Journal of Marketing Management 2009, Vol. 25, No 9-10, 867-874

[4] C D A Graham, *What's the Point of Marketing Anyway?* The Prevalence, Temporal Extent and Implications of Long-Term Market Share Equilibrium, Journal of Marketing Management 2009, Vol. 25, No 9-10, 867-874, 870

[5] A. S. C. Ehrenberg, G. J. Goodhardt, T. P. Barwise, *O zjawisku podwójnego ryzyka raz jeszcze,* in: M. Lambkin, G. Foxall, F. van Raaij, B. Heilbrunn, Zachowanie konsumenta. Koncepcje i badania europejskie. PWN, Warszawa 2001, 105-124

[6] A. Elberse (2008), *Should You Invest in the Long Tail?,* Harvard Business Review, July-August 2008, 88-96

[7] U. Świerczyńska-Kaczor, P. Kossecki, *Who Wins 'Twice'?* An Analysis of Double Jeopardy Phenomenon within Polish Internet Informational Portals, Social Science Research Network: Electronic copy of this paper is available at: http://ssrn.com/abstract=967849

[8] A. S. C. Ehrenberg, G. J. Goodhardt, T. P. Barwise, *O zjawisku podwójnego ryzyka raz jeszcze,* in: M. Lambkin, G. Foxall, F. van Raaij, B. Heilbrunn, Zachowanie konsumenta. Koncepcje i badania europejskie. PWN, Warszawa 2001, 105-124

[9] L. Uden, A. Eardley, *The Usability of Social Software [w:] Handbook of Research on Social Interaction Technologies and Collaboration Software: Concepts and Trends,* red. T. Dumova, R. Fiordo, Information Science Reference, Hershey – New York 2009, s. 574.

[10] http://www.alexa.com/topsite - According to the alexa ranking of the top sites (6 may 2010) in the global Internet Facebook is in second position, Twitter is eleven, and MySpace is twenty two

[11] C. Mooney, Online Social Networking, Lucent Books, Detroit – New York – San Francisco – New Haven – Waterville – London 2009, p. 18 - In 2003 Tom Anderson and Chris DeWolfe launched MySpace in Santa Monica, California. As music fans, the pair designed the site as a place to promote local music acts. They also wanted to be able to connect with other fans and friends. On MySpace, users created a Web page with a personal profile. Then they invited other users to become their friends. Over the next two years, MySpace grew at a tremendous pace. The site's success brought attention from investors. Rupert Murdoch, famous for his media empire, wanted to buy MySpace. Murdoch had interests in television, film, newspapers, publishing, and the Internet. In 2005 Murdoch purchased MySpace for an amazing $580 million. By early 2008 MySpace had grown to a mind-blowing 110 million active users.

[12] C. Mooney, *Online Social Networking,* Lucent Books, Detroit – New York – San Francisco – New Haven – Waterville – London 2009, p. 19 - Facebook was one site that emerged as an alternative to MySpace. In February 2004 Harvard student Mark Zuckerberg launched Facebook. The site began as a closed network for college students. Closed networks only allow users to join if they meet certain criteria. In contrast, sites such as MySpace and Friendster were open social networking sites. Anyone could sign up for an account.

[13] K. Morrell, C. Jayawardhena, *Myopia and Choice: Framing, Screening and Shopping,* Journal of Marketing Managment, 2008, Vol. 24, No. 1-2, 135-152, 143

[14] http://www.wirtualnemedia.pl/artykul/najpopularniejsze-serwisy-tematyczne-w-polsce_1 [24.04.2010] (the data about the real users, reach and views)

[15] http://www.wirtualnemedia.pl/artykul/najpopularniejsze-serwisy-tematyczne-w-polsce_1 [24.04.2010] (the data about the real users, reach and views)

[16] eMarketer (10/28/2009 28 Oct. 2009), Facebook Grabs Soc Net Share http://www.emarketer.com/Article.aspx?R=1007351

International Symposium on E-Learning – Applications

E-Learning – Applications (EL-A) symposium is organized within a framework of the International Multiconference on Computer Science and Information Technology (IMCSIT), and focuses on development of e-learning technologies and their different application areas.

The EL-A will provide an international forum for experts from the academia, the R&D sector, and the industry to discuss and exchange current results and new ideas based on the ongoing research and experience.

We are seeking the submission of high-quality and original research papers that have not been previously published and are not under review for any another conference or journal. Submissions will be reviewed by at least two referees on the basis of the originality of the work, the validity of the results, chosen methodology, writing quality and the overall contribution to the field of e-learning.

Topics include but are not limited to:

- Reusable Learning Objects – good practices and issues
- Learning Management Systems and Learning Content Management System
- Authoring Tools for Online-based Education
- Design and Development of Online Courseware
- Digital rights management and access management
- Best Practices and experiences of Online-based Education
- Global Trends in e-Learning
- Open Source and Open Content
- Semantic web in e-Learning
- E-Learning 2.0
- E-Learning as a Social Activity
- Collaborative e-Learning – tools
- Communication Technology Applications in Online-based Education
- Use of Multimedia in e-Learning
- M-Learning
- Virtual Reality Applications in Online-based Education
- Use of Gaming in e-Learning
- Scientific Virtual Laboratories.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Simple Blog Searching Framework Based on Social Network Analysis

Iwona Dolińska
University of Economics and
Computer Science in Warsaw
Stokłosy 3, 02-787 Warsaw, Poland
Email: iwona.dolinska@wsei.pl

Abstract—**Blogs are very popular Internet communication tools. The process of knowledge sharing is a very important activity in the contemporary information era. Blogs are used for knowledge sharing on any subject all over the world. Knowledge gathered on blogs can be used in personal e-learning, which is a more informal and personal way of learning than the one offered by traditional e-learning courses. However, it is not easy to find valuable knowledge in the huge amount of invalid information. In this study the Simple Blog Searching framework is proposed to improve the blog searching process. The social network analysis methods of centrality measuring help to choose more easily the best results form the long list of hits, received from a blog search tool. To incorporate social network analysis methods, the blog searching have to be expanded with the blog links searching.**

I. Introduction

BLOGS are one of the most popular of the contemporary Internet communication tools. Blogs make an easy Internet publishing available to everybody. They are used for knowledge sharing. Some blogs are authored by very active and knowledgeable bloggers focused on specific knowledge domains [1]. Any kind of information on every subject can be found in the blogosphere.

Knowledge sharing is an activity through which knowledge (i.e. information, skills, or expertise) is exchanged among people, e.g. friends, or members of a family, a community or an organization. For knowledge sharing to happen, opportunities for interaction must be present. Blogs are very good tool for such kind of interaction.

Knowledge gathered on blogs can be used in personal e-learning, which is a very important aspect of present day life [2]. In the era of the knowledge economy many workers have to improve and extent their personal knowledge almost every day.

Everyone, who wants to learn anything can use knowledge collected on blogs [1], [2]. Blogs offer high searchability, because each post can be tagged with a category and can be retrieved easily by a simple search within that specific chosen category. But there is one fundamental problem with searching the blog content – the huge amount of hits, even using a specialized blog search tool. It can be very time consuming and frustrating for the learner to browse a long list of sources to find the relevant one.

To overcome this problem a simple search blog framework idea has been elaborated and will be presented in this article. The social network analysis (SNA) methods are used in this framework to improve search efficiency. SNA methods of centrality measuring allow to choose easily the best results form the long list of hits, received from a blog search tool.

II. The role of blogs in knowledge sharing

A. Knowledge sharing evolution

We can observe the rapid expansion of the process of information exchange and information sharing during the last few years. The industrial economy is replaced by the knowledge economy. Information technology continues to infiltrate all aspects of activities in a human society, such as business, schooling etc. Nowadays, more and more information builds the human knowledge. On the other hand, the human knowledge is build not only from information. Human knowledge is build also by sharing experiences and skills. People often learn from others. The knowledge has to be properly managed to remain being accessible and to be shared among people. For knowledge economy to function well, good knowledge workers are the foundation [1]. The good knowledge worker is a person possessing a decent understanding of knowledge searching and having the skills to convey knowledge sharing effectively.

Knowledge sharing is in some aspects similar to e-learning. Both are incorporating an Internet and multimedia technologies, both can use shared knowledge repositories, accessible by web tools and finally both of them are types of technology supported learning [3].

For knowledge sharing to happen, opportunities for interaction must be present. Traditionally, people communicate to share their learned lessons, experiences, insights and best practices through face-to-face meetings. Knowledge is usually shared by telling stories. However, these methods are practical only, when community is small and immobile. These methods are not effective for connecting the minds of people dispersed around the globe [1].

Nowadays only the Internet technologies are relevant for globally sharing of knowledge and experiences among people. Chat tools enable people to meet and discuss topics of their interest in a synchronized manner, i.e. simultaneously, when all engaged parties are on-line. Internet fora allow to discuss different subjects by writing text posts to the post list. All discussion members can read and answer the posts. Blogs provide much more possibilities and are the most con-

978-83-60810-27-9/09/$25.00 © 2010 IEEE

venient and the most advanced tool for knowledge sharing and distance learning.

B. Blogs supported knowledge sharing

Blogs allow to publish posts with rich content, tagged with categories [2]. In their posts authors can place links to another posts or another information gathered on the Web. Many blog providers support a blogroll – a list of links to other blogs or sites, used or recommended by the blog author. On the other hand, blog readers can place comments and reviews, often with links to their own blogs or another Web content. All these kinds of links are presented in Fig 1. In the other words, blogs offer bidirectional communication between authors and users on the scale, which is impossible to be achieved with earlier communication tools, i.e. e-mail, discussion lists, static personal websites or even Internet forums [3].

With all these kinds of links, blogs has formed a giant network, which is called blogosphere. Blogosphere is a collective term encompassing all existing blogs and their interconnections. It is often perceived that blogs exist together as a connected community (or as a collection of connected communities) or as a social network [5].

There are several purposes for writing blogs and corresponding types of blogs in blogosphere [7]. Blogs, which serve as diaries are called personal blogs. Blogs providing commentary and opinions are called issues blogs. And blogs, which articulate ideas through writing or serve as community forums are called topical blogs [7]. From the point of view of knowledge sharing the last type is more important than the others. The blogs with significant amount of knowledge collected on them are also called as knowledge blogs (or k-blogs) [8].

Blogs create a context for dialogues between bloggers and readers. Through conversations initiated by bloggers and engaged with by readers, blog platforms build a solid base of shared experiences and mutual relationships [2]. The interaction between bloggers and other users is also reflected by the role conversion of bloggers. Initially, bloggers receive information. During this process, bloggers are data consumers. Then, bloggers absorb the knowledge they have gained and publish new knowledge on their blogs. In this case, bloggers act as knowledge publishers [6].

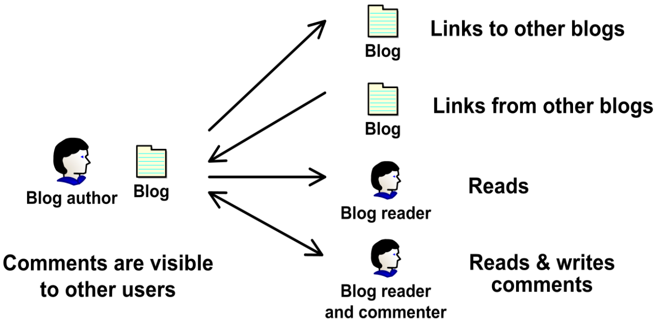


Fig 1. Bidirectional communication offered by blog, based on [4]

Computer-mediated discourse is shaped both by specific technology features and afforded by social practices of users. To better characterize blogging discourse, it is helpful to compare blogging briefly with other CMC (computer-mediated communication) fora. Blogging shares some features and practices with social networking applications within the technologies under the Web 2.0 umbrella, but it also has some distinct characteristics [8]. RSS (real simple syndication) and mashup applications, for example, allow users to summarize information items and access only updated information from multiple sources and to combine multiple sources of information into a single outlet. Bloggers, in contrast, usually develop their own content, even when referencing MSM news sources or other blogs or searching for content to create lists of links to interesting websites. Blog readers can use RSS and syndicate their favorite blogs in order to access only the latest posts [8].

III. Blogs as a knowledge source in personal e-learning

The traditional e-learning environment is asymmetric, with clear distinction between the roles of a teacher and a student [9]. Traditional learning theory emphasize mainly the teacher's point of view. The teacher is the person who prepares the curriculum, learning aids and points out learning path. The students' role is more passive in such situation – they learn from materials prepared earlier and accomplish exercises and quizzes. This traditional approach has been used for a long time in formal university e-learning courses. E-learning systems, which can be briefly categorized as learning management systems (LMS) or learning content management systems (LCMS), still concentrate on the knowledge delivery by lecturers only [9].

In the last years the need of more informal learning has been arisen. Many university programs require numerous informal learning activities, which are difficult to be achieved with LMS or LCMS systems. On the other hand, the Web 2.0 expansion brought new tools, like wikis, blogs, news readers, communication tools etc. They allow for new possibilities, like learning with people, controlling the learning resources, managing the student activities, and integrating the formal learning with informal learning [10].

The traditional asymmetric learning environment, with its clear distinction between the roles of instructors and students, is becoming more symmetric and based on communities of practice. Students will no longer passively consume learning materials but actively create and disseminate knowledge. Personal learning environments (PLEs) emphasize symmetric connections with a range of services in both formal (instructor-led) and informal (student-led) learning, work, and leisure. Rather than integrating tools within a single context, PLEs coordinate connections between users and a wide range of services offered by organizations and other individuals [11].

Personal learning environment can be defined from different points of view. For some, the PLE concept facilitates choice and control by students, allowing the selection and combination of informal and formal learning opportunities from a variety of sources. Others see the PLE as an extension of the portfolio, providing a learner-centered environment in which students can record achievements and plan and work towards new goals [10].

But in spite of these differences, personal e-learning environments are well adapted to the life-long learning, which is very important activity in the nowadays information world. Personal learning includes a progress of knowledge identification, knowledge acquisition, knowledge development and knowledge utilization [12]. The stages of identification and acquisition of knowledge can be supported by using knowledge gathered on blogs. But the most valuable and innovative knowledge is hard to be found, and it lies within distributed communities and networks.

IV. Searching in blogosphere

A. Blogosphere searching problems

Everyone, who wants to learn about any subject can use knowledge gathered on blogs [1], [2]. Blogs offer high searchability, because each post can be tagged with a category and can be retrieved easily by simply searching within that specific chosen category. But, like it was mentioned above, there is one fundamental problem with searching in blogs – the huge amount of hits (low query selectivity), due to tremendous dimension of the blogosphere. There are specialized blog search tools like Google Blog Search, Yahoo API Search, Blog Pulse (http://www.blogpulse.com/), BlogCatalog (http://www.blogcatalog.com/), and others in blogosphere. These tools allow users to search for data in blogs by tipping some keywords. But even using such specialized blog search tool does not make the search result list shorter. They all give extremely long lists of results. It can be very time consuming and frustrating for the student to break through such list of sources to find the most relevant ones. The most valuable knowledge is usually hard to be found.

The researchers have already worked out some solutions to solve the problem of searching in blogs. Blog searching has usually a specialized research purpose. The blog searching methods incorporate different techniques, depending on the domain of use. One part of these methods are based on blog ontology, another is based on clustering methods and finally some of them incorporate social network analysis methods.

B. The solutions developed in blog searching domain

The example of using blog ontology is described in [13]. Campos and Divino give a strict definition of the formal specification of structured Social Web data to express the information contained blog data sources [13]. The semantic structure provided in the blog ontology can be used by users and applications to state semantic tagging to blog resources. It can help then to retrieve relevant resources, to categorize and to organize content, and to navigate it meaningfully. The authors of this study propose also a framework for Blog Visualization System, which is based on the blog ontology. This tool can help users to track and analyze the spreading of memes through blogs. The term "meme" refers to a unit of cultural information (including deities, concepts, ideas, theories, opinions, beliefs, practices, habits, dances and moods) that is transmitted verbally or by demonstration from one mind to another mind [13]. So this is a specialized tool for visualization of meme spread path in the blogosphere.

Agarwal, Galan, Liu, and Subramanya present the blog clustering method in [14]. Blog site clustering helps better organize the information. This method also allows for optimizing the search engine by reducing the search space. So the search results are obtained faster [14]. Presented method of clustering incorporates a collective wisdom, gathered by bloggers on Blog Catalog directory, which allows bloggers to label the blogs under a given hierarchy. The collective wisdom is the wisdom generated by bloggers, when they tag and catalog their posts. Clustering method is applied to the labels rather than to blogs [14]. It means that the authors cluster blog categories. Such approach is time sensitive and adaptive to the current interests. But this method is difficult to use by an individual user.

A blog mining framework described in [7] is another very interesting example of blogosphere analysis. This framework consists of blog spider, blog parser, blog analyzer and visualizer [7]. Blog spider monitors and downloads content from multiple bloghosting sites. Blog parser extracts information from blogs, it is a specialized tool for blog content analysis. Blog analyzer extracts particular key phrases by using text mining technics. The blog analyzer is also capable of analyzing the network relationships among bloggers. And finally the blog visualizer presents content and network analysis [7]. Described framework allows to make complex analysis of the blogosphere on specified domain, like politics, business, cultural studies, and others [7].

C. Social network analysis in blogosphere searching

Many researches use social network analysis for the blogosphere exploration. But this method is used to find particular blogger groups rather than for blog searching.

Pikas studied the communities within the science blogs network [15]. She applied social network analysis to yield an understanding of the topology of the science blogosphere, basing on the blogs gathered on ScienceBlogs.com. The research started with an initial list of blogs, from which the blogroll links were gathered. Another list of links was selected from links in comments. Obtained lists of blogs were used to create a blog network and then social network analysis methods were applied to detect and describe network communities of scientists owning blogs [15].

Social network analysis is also used for finding friend groups in blogosphere, like it is described in [16]. S.-T. Kuan, B.-Y. Wu, and W.-J. Lee present a pre-processing algorithm based on n-clique extension for social network. After the pre-processing of social network the cliques can be directly found. These cliques can be formed into some representative friend groups [16].

V. Simple blog searching framework

A. The simple blog searching framework overview

Social network analysis, as it is mentioned, is often used to analyze blogosphere content, to describe relationship between bloggers or to find influential bloggers in the network. The fact that influential bloggers usually possess good knowledge blogs is the basis of the presented research. A blog of influential blogger, from the SNA point of view, has a high centrality value. As a simple centrality measure the in degree value can be used. In the blogosphere the in degree value denotes the number of links pointed at this blog. In the other words, the in degree value corresponds to the number of the links from other blogs to this blog. These links can be placed on blogroll or in posts. The fact that a blog has the high in degree value means, that many other bloggers regard this blog as a valuable one. Such blog can be useful in e-learning [17].

To find links pointed at the blog, the other blogs have to be searched and links from their blogrolls have to be collected. All these links form a social network, which can be visualized in the form of a graph and than can be analyzed with SNA methods. If knowledge blogs with the high in degree centrality are found in the blogosphere, these blogs could be used in e-learning.

The entire blogosphere can be represented as a directed graph, so any graph node (i.e. blog) can have not only the in degree measure, but also the out degree measure and the out degree centrality. The out degree value denotes the number of links from this blog to the other blogs. If a blog has a high out degree value, it means, that many links pointed out to another blogs are gathered on this blog. This fact is also incorporated in the described framework. High out degree value usually means, that the blogger, possessing his blog, has surfed a lot through the blogosphere and that he has gathered many interesting links and a lot of information on the blog. So such blog can be useful in e-learning as well [17].

For directed graph the all degree value for node can be calculated as a sum of in degree and out degree values. All degree centrality can be also used in blog searching process. It could be used as a additional criterion, confirming the value of a blog for e-learning.

In this article the simple blog searching (SBS) framework is proposed to enable users to make the process of searching in the blogosphere easier than with search tools described in the previous chapter. The scheme of this framework is presented in Fig 2. In this framework all the centrality measures, described above, are used to find the most valuable knowledge blogs.

B. *The simple blog searching framework modules*

The simple blog searching framework consists of four modules: BL Search Tool, SNA Analyzer, Network Visualizer and K-Blogs List. All of them are working in a pipe manner, i.e. the output of one module is the input for the next one.

The first module of SBS framework is BL Search Tool. BL means that this module is searching for the blogs and for links connecting these blogs. The links between blogs are needed for the next step, i.e. SNA analysis.

In the presented framework the blog searching process starts from finding the blog list with a blog searching tool. User has to define the selected category for this search. This category is specified by one or more blog tags or post tags. Additional criteria can include a time range or a blog language. After that, from the obtained blogs, all blogroll links are gathered and then passed to the next step of the analysis. To obtain a better result, i.e. better connected network, the blogroll link gathering can be repeated several times. The initial list of blogs for each repetition is a result list from the previous repetition. This means that newly found blogs (pointed by links originating from the initial blog list) are added to the intermediate result. The number of link search steps can be one of the parameters of the module. The BL Search Tool can be implemented as a specialized script.

The next SBS part, SNA Analyzer is the module, in which SNA methods are used to form the network graph and then to find important network nodes. First, SNA Analyzer analyzes the list of blogs and links connecting them, obtained from the first module. All the data is formed into the directed graph, representing obtained social network of blogs. Then SNA Analyzer finds the nodes with the high values of in degree centrality and the out degree centrality measure. The nodes with the high all degree centrality measure can be found next, as a sum of two previous centrality values. The nodes with the high centrality value can be selected in two manners. User can choose the nodes with a degree greater than the configured threshold value or he can choose *n* nodes with the highest degree value, where *n* is a configured number of nodes.

In the current version of the SBS framework, Pajek [18] works as SNA Analyzer and Network Visualizer. Implementation of these two steps is of course not restricted to this tool only. One can use another social network analysis tools, like Graphviz [19], R [20] or Ucinet [21].

The last SBS framework step is K-Blogs List, module presenting the final results of blog search to the user.

VI. Application of the SBS framework

A. *Data Gathering*

For the purpose of this research it is assumed that potential framework user wants to learn about e-learning methods and practice, so blogs about "e-learning" (Pol. e-learning) and "distance learning" (Pol. nauczanie na odległość) were searched for. Data gathering was performed in April and May of 2010. Data gathering started with an initial list of blogs acquired from a specialized blog search tool, described in section IV.A. The search was limited to blogs written in Polish only. Blog searching included two cate-

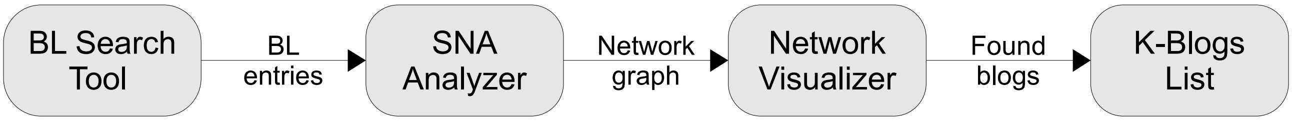


Fig 2. Simple Blog Searching framework scheme

gories, defined by tags “e-learning” and “nauczanie na odległość”.

The obtained list of blogs was then manually visited with the web browser (Firefox was used) to verify the value of the knowledge featured. Commercial blogs were left out. For the purpose of this research the small sample of Polish blogs about e-learning was finally chosen. After finishing the initial list creation, all blogs from this list were visited again to gather the blogroll links. And this step was repeated once again, as it was explained in section V.B. The first module of the SBS framework, the BL Search Tool, finished its work with the network file, containing the list of blogs and links connecting these blogs.

B. Data Analysis

The obtained blogs and links, i.e. BL entries, were then forwarded to the SNA Analyzer, and that to Network Visualizer. As it is mentioned above, Pajek was used in this example to analyze and visualize the network. The resulting network is presented in Fig 3. It turned out that there is a very small number of blogs about e-learning written in Polish in the blogosphere. The resulting network graph of the e-learning blogs network is weakly connected, so the density of the network is very small (0.05). But even such a simple sample is still sufficient for showing different aspects of using the SBS tool.

The first step of network analysis is finding the nodes with the highest in degree value. Fig 3 presents the result of this step (green point represents the blog with the highest in degree value, blue points represent the second places, red points represent the remaining blogs). Table 1 contains the five nodes with the highest in degree values. The first and second column of this table contains the data obtained from SNA analysis: node number (*NN*) and its in degree value, i. e. in degree centrality measure (*InD*). Following two columns contain blog name and blog address. The fifth column shows the effect of manual checking of the blog content to judge its quality for usage in classroom scenario. This inspection was done to verify the SNA analysis result. In the Table 1 the most valuable (with the highest in degree centrality value) is the node number 5. Manual checking shows it is a very well written blog and a good source of knowledge. In this paper author's opinion the first and the second blog could be quite easily used in classroom scenario. The three following blogs also contains useful information and their parts could be used as a supplementary lecture.

The second step of network analysis is finding the nodes with highest out degree value. Fig 4 shows the result of this step (green point represents the blog with the highest out degree value, blue points represent the second places, red points represent the remaining blogs). Table 2 contains the three nodes with the highest out degree values. The meaning of the table columns is the same as for Table 1, with exception for the second column, containing out degree values for nodes (*OutD*). The first blog in this category, blog number 7, has the highest out degree in the presented example, because its author prepared a rich blogroll. In this paper author's opinion all three blogs could be used in classroom scenario, but the second (number 1) in some topics only (a very good blog about cognitive science).

Both described steps of searching (for the highest in degree value and for the highest out degree value) give good results for established task of knowledge blogs finding. It means that a user can also take into account the result of the all degree value, which can be the next step of analysis. The result is presented in Fig 5 (point colors like in Fig 4). The best blogs in this classification are presented in Table 3. The result of this classification is the combination of the best results in first and second stages of analysis. It points out, that it is good criterion.

This research shows additionally that, from the point of view of presented method of blog searching, the connection making between blogs is a very important activity of the blog authors. The more connections from and to the blog, the more easily the potential student will find and use this blog.

The single nodes visible in Fig 3, are blogs with a very few information about e-learning or rarely updated blogs, but found by the search tool, because they have posts labeled with "e-learning" tag. The method of searching implemented in the SBS framework efficiently eliminates such nodes. They have zero in degree and out degree value. So they can be filtered out by such searching method.

Generally all blogs gathered in this research are valuable source of knowledge in the area of "e-learning".

VII. Conclusions

The process of knowledge sharing is a very important activity in the contemporary information era. This process can be effectively supported by good and useful technologies like blogs. The huge amount of knowledge is gathered on blogs, but the most valuable and innovative knowledge is hard to be found, and it lies within distributed communities and networks. This study presents a framework for simple blog searching, to make the process of knowledge gathering easier.

Simple Blog Search framework consists of four modules. SNA measures of in degree centrality and out degree centrality are used to help user in searching the blogosphere. The all degree centrality could be an additional searching criterion, confirming the blog usefulness. To use such a method the connections between blogs, constructing the network (i.e. hyper-links), must be used. To incorporate social network analysis methods, the common blog searching (i.e. searching with one or many tags) have to be extended with the blog links searching. In presented example links between blogs were gathered from blogrolls only. It is an easier way, but it can lead to weakly connected network, like shown in Fig 3. To create better connected network, links from comments and posts should be also collected. This is the subject for future work, because it demands more advanced scripts for links searching and collecting.

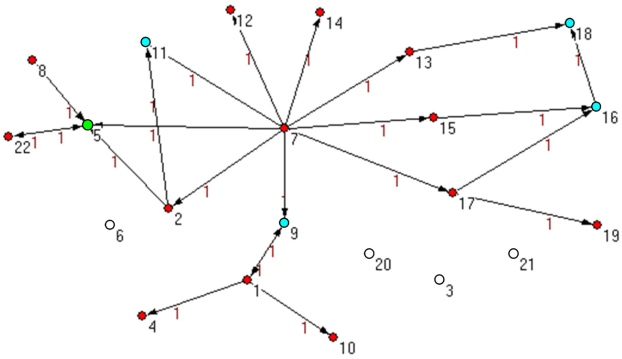


Fig 3. The acquired social graph of the e-learning blog network with in degree partition marked [17]

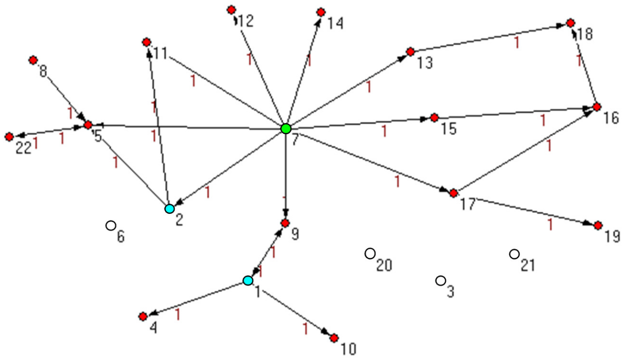


Fig 4. The out degree partitions of the e-learning blog network [17]

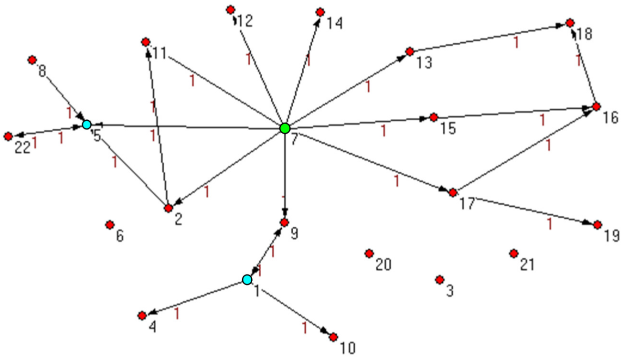


Fig 5. The all degree partitions of the e-learning blog network

Also the core of this method, i.e. centrality measuring manner (in degree and out degree centrality) will be replaced in the future research by more advanced measures of centrality and prestige measure calculation. To better confirm presented method of blog searching the English sites will be used in the future. It allows to extend the example sample and probably get more interesting result.

Table 1. The nodes with the highest in degree value

NN	InD	Blog name	Blog address	Blog usefulness
5	4	Elearning 2.0	http://elearning-20.blogspot.com/	good source
9	2	E-learning według Bartka	http://e-learning.blog.pl/	good source
11	2	Edukacja-online	http://edukacja-online.pl/	good parts, supplement
16	2	Edunews	http://www.edunews.pl/	good parts, supplement
18	2	e-mentor	http://www.e-mentor.edu.pl/	good parts, supplement

Table 2. The nodes with the highest out degree value

NN	Out D	Blog name	Blog address	Blog usefulness
7	9	Mentor Online	http://mentor.sceno.edu.pl/	good source
1	3	Mechanika umysłu	http://www.mechanikaumyslu.pl/	good source
2	2	eLearning dla opornych	http://elearning2.blox.pl/	good, basic level

Table 3. The nodes with the highest all degree value

Node number	All degree	Blog name	Blog address
7	9	Mentor Online	http://mentor.sceno.edu.pl/
5	5	Elearning 2.0	http://elearning-20.blogspot.com/
1	4	Mechanika umysłu	http://www.mechanikaumyslu.pl/

References

[1] A. A. Nasr, M. M. Ariffin, "Blogging as a means of knowledge sharing: Blog communities and informal learning in the blogosphere," *International Symposium on Information Technology*, vol. 2 2008, pp. 1–5.

[2] Y.-J. Chang, Y.-S. Chang, C.-H. Chen, "Assessing Peer Support and Usability of Blogging Technology," *Third International Conference on Convergence and Hybrid Information Technology,* 2008, vol. 1, pp. 184–189.

[3] I. Dolińska, "Blogs as a distance learning tool" in *Computer Aided Technologies in Science, Technique and Education*, A. Jastriebow, Ed. Radom 2009, pp. 281–284.

[4] I. Dolińska, "The role of blogs in e-learning and knowledge sharing," in *E-education advancement*, L. Banachowski, Ed. Wydawnictwo PJWSTK, Warsaw 2010, pp. 173–181.

[5] "Blogosphere" in Wikipedia, available: http://en.wikipedia.org/wiki/Blogosphere, accessed 05/09/10.

[6] X. Zhao, "Research on the Knowledge Transfer in Academic Blog," *Second International Symposium on Intelligent Information Technology Application*, 2008.

[7] M. Chau, P. Lam, B. Shiu, J. Xu, C. Jinwei, "A Blog Mining Framework," *IT Professional*, 2009, vol. 11, no. 1, pp. 36–41.

[8] E. Davidson, E. Vaast E, "Tech Talk: An Investigation of Blogging in Technology Innovation Discourse," *IEEE Transactions on Professional Communication*, March 2009.

[9] H.-Y. Chiu, S.-Z. Wen, C.-C. Sheng, "Apply Web 2.0 Tools to Constructive Collaboration Learning: A Case Study in MIS Course," *Fifth International Joint Conference on INC, IMS and IDC*, 2009, pp. 1638–1643.

[10] C. D. Milligan, P. Beauvoir, M. W. Johnson, P. Sharples,S. Wilson, and O. Liber, "Developing a Reference Model to Describe the

Personal Learning Environment," *Innovative Approaches for Learning and Knowledge Sharing*, Springer Berlin 2006, pp.506-511.

[11] M.M. Organero, C.D. Kloos, P.M. Merino, "Personalized Service-Oriented E-Learning Environments,“ *IEEE Internet Computing*, vol. 14 , no. 2, 2010, pp. 62–67.

[12] H. Li, X. Yang, S. Zao, "Research on Postgraduate's Personal Knowledge Management Based on Blog and RSS," *International Symposium on Knowledge Acquisition and Modeling*, 21-22 Dec. 2008, pp. 191−195.

[13] A. Campos, R. Dividino, "Blog Ontology (BloOn) & Blog Visualization System (BloViS)," *First International Workshop on Ontologies in Interactive Systems*, 2008, pp 83–88.

[14] N. Agarwal, M. Galan, H. Liu, S. Subramanya, "Clustering Blogs with Collective Wisdom," *Eighth International Conference on Web Engineering*, 2008, pp. 336–339.

[15] C. K. Pikas, "Detecting Communities in Science Blogs," *IEEE Fourth International Conference on eScience*, 7-12 Dec. 2008, pp. 95–102.

[16] S.-T. Kuan, B.-Y. Wu, W.-J. Lee, "Finding Friends Groups in Blogosphere," *22nd International Conference on Advanced Information Networking and Applications - Workshops,* 25-28 March 2008, pp. 1046–1050.

[17] I. Dolińska, "The application of SNA methods in thematic blog searching for the purpose of distance learning," *X Conference Virtual University- model, tools, practice*, Warsaw 2010, submitted for publication.

[18] "Pajek—Program for Analysis and Visualization of Large Networks," available: http://pajek.imfm.si/doku.php, accessed 05/14/10.

[19] "Graphviz - Graph Visualization Software," available: http://www.graphviz.org/, accessed 05/14/10.

[20] "The R Project for Statistical Computing," available: http://www.r-project.org/, accessed 05/14/10.

[21] "UCINET," http://www.analytictech.com/ucinet/, accessed 05/14/10.

6th Workshop on Large Scale Computations on Grids and 1st Workshop on Scalable Computing in Distributed Systems

The Large Scale Computing in Grids (LaSCoG) workshop originated in 2005, and when it was created we have stated in its preamble that:

"The emerging paradigm for execution of large-scale computations, whether they originate as scientific or engineering applications, or for supporting large data-intensive calculations, is to utilize multiple computers at sites distributed across the Internet. In particular, computational Grids are collections of distributed, possibly heterogeneous resources which can be used as ensembles to execute large-scale applications. While the vision of the global computational Grid is extremely appealing, there remains a lot of work on all levels to achieve it."

While, it can hardly be stated that the issues we have observed in 2005 have been satisfactorily addressed, a number of changes has happened that expanded the world of large-scale computing. Today we can observe emergence of a much more general paradigm for execution of large-scale applications, whether they originate from scientific or engineering areas, or they support large data-intensive calculations. These tasks utilize computational Grids, cloud-based systems and resource virtualization. Here, collections of distributed, possibly heterogeneous resources, are used as ensembles to execute large-scale applications.

This being the case, we have decided to keep the LaSCoG workshop tradition alive, but to co-locate it with a conference which will have an appropriately broader scope. This is how the Workshop on Scalable Computing in Distributed Systems (SCoDiS'10) emerged.

The LaSCoG-SCoDiS'10 pair of events shares a joint Program Committee and is envisioned as a forum to promote an exchange of ideas and results aimed at addressing sophisticated issues that arise in developing large-scale applications running on heterogeneous distributed systems.

Covered topics include (but are not limited to):

- Large-scale algorithms and applications
- Cloud computing
- Symbolic and numeric computations
- High performance computations for large scale simulations
- Large-scale distributed computations
- Agent-based computing
- Data models for large-scale applications
- Security issues for large-scale computations
- Science portals
- Data visualization
- Performance analysis, evaluation and prediction
- Programming models
- Peer-to-peer models and services for scalable Grids
- Collaborative science applications
- Business applications
- Data-intensive applications
- Operations on large-scale distributed databases
- On-demand computing
- Computation as a service
- Federation of compute capacity
- Virtualization supporting computations
- Self-adaptive computational / storage systems
- Volunteer computing

Program Committee

Rui Aguiar, Universidade de Aveiro, Portugal

Ishfaq Ahmad, UT Arlington, USA

David Anderson, UC Berkeley, USA

Mark Baker, University of Reading, United Kingdom

Xu Baomin, Beijing Jiaotong university, China

Andrej Brodnik, University of Primorska, Slovenia

Marian Bubak, AGH University of Technology, Poland and UvA Amsterdam, Netherlands

Hsu Ching-Hsien, Chung Hua University, Taiwan

Jose Cardoso Cunha, Universidade Nova de Lisboa, Portugal

Pasqua D'Ambra, ICAR-CNR, Italy

Frederic Desprez, INRIA, France

Beniamino Di Martino, Seconda Universita' di Napoli, Italy

Salvatore Filippone, Universita di Roma 'Tor Vergata', Italy

Ian Foster, Argonne National Lab & The University of Chicago, USA

Maria Ganzha, University of Gdansk and IBS PAN, Poland

Wolfgang Gentzsch, DEISA, Germany

Pawel Gepner, Intel, Poland

Minor Gordon, NEC High Performance Computing Europe, Germany

Dorian Gorgan, Technical University of Cluj-Napoca, Romania

Andrzej Goscinski, Australia, Australia

George Gravvanis, Democritus University of Thrace, Greece

Daniel Grosu, Wayne State University, USA

Pilar Herrero, Facultad de Informática – Universidad Politécnica de Madrid, Spain

Wei Jie, Thames Valley University, United Kingdom

Alexey Kalinov, Cadence Design Systems, Russian Federation

Aneta Karaivanova, Institute for Parallel Processing – BAS, Bulgaria

Helen Karatza, Aristotle University of Thessaloniki, Greece

Daniel S. Katz, University of Chicago & Argonne National Laboratory, USA

Jacek Kitowski, AGH University of Technology, Poland

Dieter Kranzlmueller, Ludwig-Maximilians-Universitaet Muenchen (LMU), Germany

Jan Kwiatkowski, Wroclaw University of Technology, Poland

Alexey Lastovetsky, University College Dublin, Ireland

Gregor von Laszewski, Indiana University, USA

Dong Woo Lee, OIKO Laboratory, Korea, Republic of

Alexander Legalov, Siberian Federal University, Russian Federation

Thomas Ludwig, University of Hamburg, Germany

Loris Marchal, CNRS, France

Tudruj Marek, Polish Academy of Sciences, Polish-Japanese Institute of Information Technology, Poland

John P. Morrison, University of Cork, Ireland

Thoai Nam, Ho Chi Minh City University of Technology, Vietnam

Henry Neeman, University of Oklahoma, USA

Richard Olejnik, University of Lille I, France

Can Ozturan, Bogazici University, Turkey

Michael Resch, HLRS, University of Stuttgart, Germany

Isaac D. Scherson, University of California, Irvine , USA

Erich Schikuta, University of Vienna, Austria

Bertil Schmidt, Nanyang Technolgocial University, Singapore

Bernhard Schott, Platform Computing, Germany

Wolfgang Schreiner, Johannes Kepler University, Austria

Simon See, Sun Microsystems Inc and Nanyang Technological University, Singapore

Hong Shen, University of Adelaide, Australia

Ha Yoon Song, Hongik University, Korea, Republic of

Frederic Suter, CNRS – IN2P3 Computing Center, France

Domenico Talia, University of Calabria & ICAR-CNR, Italy

Pavel Telegin, JSCC RAS, Russian Federation

Lang Van Tran, Vietnam Academy of Science and Technology, Vietnam

Denis Trystam, Grenoble University, France

Pavel Tvrdik, Czech Technical University in Prague, Czech Republic

Dick van Albada, University of Amsterdam, Netherlands

Stefan Wesner, High Performance Computing Centre Stuttgart, Germany

Roman Wyrzykowski, Technical University of Czestochowa, Poland

Chao-Tung Yang, Tunghai University, Taiwan

Albert Zomaya, The University of Sydney, Australia

Organizing Committee

Marcin Paprzycki (Chairman), WSM and IBS PAN, Poland

Dana Petcu, West University of Timisoara, Romania

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 621–628

ISBN 978-83-60810-27-9
ISSN 1896-7094

Exploratory Programming in the Virtual Laboratory

Eryk Ciepiela, Daniel Harężlak, Joanna Kocot,
Tomasz Bartyński, Marek Kasztelnik, Piotr Nowakowski, Tomasz Gubała,
Maciej Malawski, Marian Bubak
Institute of Computer Science, AGH,
Mickiewicza 30, 30-059 Krakow, Poland
ACC CYFRONET-AGH
Nawojki 11, 30-950 Krakow, Poland
Email: {malawski,bubak}@agh.edu.pl

***Abstract*—GridSpace 2 is a novel virtual laboratory framework enabling researchers to conduct virtual experiments on Grid-based resources and other HPC infrastructures. GridSpace 2 facilitates exploratory development of experiments by means of scripts which can be written in a number of popular languages, including Ruby, Python and Perl. The framework supplies a repository of gems enabling scripts to interface low-level resources such as PBS queues, EGEE computing elements, scientific applications and other types of Grid resources. Moreover, GridSpace 2 provides a Web 2.0-based Experiment Workbench supporting development and execution of virtual experiments by groups of collaborating scientists. We present an overview of the most important features of the Experiment Workbench, which is the main user interface of the Virtual laboratory, and discuss a sample experiment from the computational chemistry domain.**

I. Introduction

MODERN life sciences, particularly simulations in biochemistry, genetics and virology, impose significant requirements on underlying IT infrastructures. Such requirements can be loosely grouped into two general domains: demand for computational resources and demand for new software tools facilitating effective, productive and collaborative exploitation of such resources by a vast range of beneficiaries. While the key goal of supporting scientific experimentation with computerized infrastructures remains the provision of large-scale computational and data storage facilities [1], it is equally important to supply scientists with tools enabling them to collaboratively develop, share, execute, publish and reuse virtual experiments.

The presented Virtual Laboratory, first conceived as part of the ViroLab project [2] and currently being extended within the scope of the PL-Grid project, aims to respond to these requirements by supplying software which permits the execution of virtual experiments. Experiments can be written in popular scripting languages and executed on the distributed resources provided by HPC institutions participating in the project. The goal of the Virtual Laboratory is to a propose a model and facilities for exploratory, incremental scripting – already omnipresent in e-scientific research – and make it reusable and actionable for entire communities. Our framework bridges the gap between the oft-inaccessible high performance computing infrastructures and the end users (i.e. domain scientists), accustomed to running calculations and collating experimental data on their desktop computers. Rather than persuade the scientists to change their daily habits, we want to provide an environment which meshes seamlessly with their style of work, yet extends their experimentation and collaboration potential with the capabilities of high-performance computing clusters.

Our experience gathered in the course of developing the ViroLab Virtual Laboratory for virologists [2], [3], [4], the APPEA runtime environment for banking and media applications in the GREDIA project [5] as well as the GridSpace environment for running in-silico experiments, has been augmented with user requirement analysis conducted during the initial phase of the PL-Grid project, involving groups of scientists from various domains such as physics, chemistry and biology. The Virtual Laboratory presented in this paper should be considered as an evolution of the approach undertaken in the ViroLab project. The new virtual laboratory is focused on interactive and exploratory programming [6], together with a Web 2.0 interaction model.

This paper is organized as follows. In section II we compare our concept with other approaches. After describing our motivation in section III we introduce the main concepts of Virtual Laboratory in section IV while its architecture is discussed in section V. In section VI we introduce GridSpace platform, which is the base technology which implements the virtual laboratory. Section VII contains an overview of the most important features of the Experiment Workbench, which is the main interface of the Virtual laboratory. Further on, we provide a description of the steps which the user undertakes while working with the Virtual Laboratory (section VIII), together with use cases. The conclusions and future work can be found in section IX.

II. Related Work

Scientific workflow systems are important tools for development and execution of e-science applications [7]. Thanks to the well-defined workflow and dataflow models in systems such as Taverna [8] or Kepler [9] it is possible to graphically design applications which can then be executed on remote infrastructures. Scientific workflows can be subject of exploratory programming, as in the case of Wings project [6], as well as of collaborative sharing, such as in the case

978-83-60810-27-9/09/$25.00 © 2010 IEEE

of myExperiment social networking website [10]. The main drawbacks of workflow systems are related to the fact that contrary to programming languages, abstract workflow models are often insufficient for describing the required application flow.

Scripting environments are becoming increasingly important in scientific applications and modern petascale systems. An example of this evolution is Swift Script [11] – a dedicated language designed to describe large-scale computations, involving massive data processing. An important feature of Swift is its mapping between data files and programming language variables, which facilitates input/output processing. Scripting can also be used for debugging and instrumenting parallel applications [12]. Our approach is similar in the sense that we intend to use scripting to describe the high-level workflow of the application while retaining interactivity and enabling multiple interpreters to be combined together.

The need to integrate diverse applications, data sources and technologies emerges not only in scientific applications, but also in enterprise systems [13]. Enterprise Service Bus solutions such as ServiceMix or GlassFish, aim to facilitate such integration in the Web service context [14]. The BPEL workflow language can also be applied to scientific applications [15]. However, we believe that such enterprise workflow systems are too heavyweight for simple exploratory programming, where a scripting approach seems to be more appropriate.

The GridSpace environment, which is the foundation of the ViroLab Virtual Laboratory [4], has beed developed by our team to support complex applications running on e-infrastructures such as clusters, Grids and Internet-accessible Web services. One of the goals was to support heterogeneous middleware systems using the Grid Object abstraction layer [16] and facilitate access to distributed data sources. Additional important features include support for collaborative work including tools for application development, sharing, reuse and Web-based accces. Moreover, provenance and result management components provide semantic descriptions of data and enable users to view experiment execution histories. The limitations of GridSpace include its relatively high architectural complexity and the fact that only the Ruby scripting language is supported.

III. Motivation and Goals of the Virtual Laboratory

The main features of Virtual Laboratory result from the experience gained during the ViroLab project, the requirements of external users, as well as from discussions with potential users of the PL-Grid project. The main high-level objectives are as follows:

- To provide an environment which facilitates dealing with scientific application throughout its entire lifecycle (development, deployment, operation, maintenance);
- To reflect and support the day-to-day work of scientists who need to deal with software tools – workflows, procedures (including informal ones), scripts, etc. – enhanced by modern Web 2.0 tools;
- To addresses exploratory programming and a specific type of applications called experiments.
- To support collaborative work of teams of scientists in a Web 2.0 model.

The specific goals of Virtual Laboratory are based on the analysis of e-science applications and discussions held with their authors. The main contributions come from the fields of bioinformatics and computational chemistry. Requirements include:

- Support for different scripting languages – particularly Ruby, Python, Perl and awk;
- Provisioning of tools for publishing and reusing applications/experiments;
- Support for dynamic workflows – as each step of an experiment may depend on the results of previous steps, some workflows cannot be entirely predefined. Moreover, some experiment steps may involve batch jobs;
- Support for parameter-study research, conducted either on the level of a single tool or an entire workflow;
- Support for logging experiments and ensuring their reproducibility;
- Providing easy access to scientific software packages; (the suites most commonly used in PL-Grid include Gaussian, GAMESS, TurboMole and ADF1);
- Direct access to local PBS systems without an additional grid middleware layer;
- Support for creating and using format converters for different tools, as well as adapters for different data sources (not necessarily databases);
- Secure management of user credentials and other sensitive data.

In addition to these requirements, the virtual laboratory needs to satisfy several non-functional and more technical requirements, which are described in more detail in section VI.

IV. Virtual Laboratory Concepts

The key concept associated with the Virtual Laboratory is the experiment. As defined in [3], an experiment is a process that combines data with a set of activities (programs, services) which act on that data in order to produce experiment results. it is important to distinguish the experiment plan – a specific piece of software, written using scripting languages – from the experiment run (execution of the experiment). The key feature here is that the experiment may represent a complex workflow, going beyond simple, repeatable manual execution of installed programs.

The *experiment plan* combines steps realized by a range of software environments, platforms, tools, languages etc. It is developed, shared and reused collaboratively by ad-hoc research teams. The experiment is composed of collaboratively owned libraries and services used (called *gems*) and experiment parts (called *snippets*). Gems are used to represent either program libraries, such as BioPython [17], or applications such as

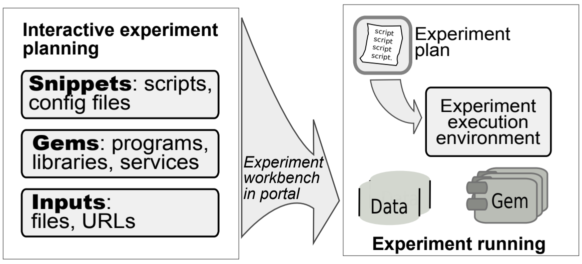

Interactive experiment planning
Snippets: scripts, config files
Gems: programs, libraries, services
Inputs: files, URLs
Experiment workbench in portal
script script script script
Experiment plan
Experiment execution environment
Data
Gem
Experiment running

Figure 1. The GridSpace 2 experiment concept

Gaussian [18] or external services, such as the ones from EBI [19]. Snippets refer to separate pieces of code, either in scripting languages, such as Python or Bash, or in other domain specific languages or input file formats, such as used by Gnuplot or Gaussian.

The key paradigm of the enhanced version of virtual laboratory, namely exploratory programming, involves experimentation – step-by-step programming where steps are not known in advance but rather defined on an ad-hoc basis, depending on the results of previous steps. The experiment may need to be re-enacted numerous times, with some ad-hoc customization introduced dynamically, once workflow execution has already commenced. Experiment execution cannot be fully automated and requires continuous supervision, validation or even intervention. This implies a dynamic nature of experiment plan – certain decisions need to be taken at runtime (e.g. code provided from input data). Nevertheless, experiment execution has to remain traceable, verifiable and repeatable.

Due to its focus on exploratory programming, the virtual laboratory is best suited for a try-evaluate-decide process, without losing the context of the experimentation, as opposed to applications with well-known implementations. The virtual laboratory aims at composing and automating higher-level, time-consuming workflows that combine existing software. This property contrasts with applications which perform "atomic" processing. Another interesting feature involves support for novel combinations of existing software modules, data sources and computational capabilities which may result in valuable utilities, as opposed to well-defined workflows which are already addressed by existing software.

V. Architecture of Virtual Laboratory

Fig. 2 presents the architectural overview of the Virtual Laboratory. It is divided into four layers: the Experiment Wokbench layer, available to the user as a set of Web applications, the Experiment Execution Layer which forms the runtime environment of the platform, the Gem Layer which includes all generic and application-specific libraries accessible to the experiments, and the Grid Fabric layer, with all the resources and middleware needed to access them.

The topmost layer is formed by a Web portal which constitutes an entry point for the whole Virtual Laboratory. This gives users access to the Virtual Laboratory from any workstation equipped with a web browser. This layer exposes a portal which is intended as a common, tool-rich workbench for all Virtual Laboratory researchers where they can perform their daily experimentation, collaborate, communicate and share resources (including reusable code). This layer is accessible to end-user browsers via the HTTPS protocol. File Manager is responsible for easy access to data files, Experiment Console allows editing and running experiment snippets, Credential Manager helps handle passwords, certificates and other secrets required by some parts of experiments, and Graphical Experiment Builder is intended to construct more complex experiments graphically.

Further down lies the Experiment Execution Layer where consecutive parts of experiments, provided by the users through the Portal, are evaluated in the context of a particular user account on an experiment host machine. A single experiment can invoke multiple interpreters, such as Python or Ruby. The key concept here is that an experiment can be developed in a piecewise fashion. Individual experiment parts are executed by an experiment interpreter which preserves state (namespace, runtime values) between evaluations. Consequently, experiment development and execution may overlap, and the activities of writing and executing code are intertwined. Such an approach admits introspective, interactive, explorative and dynamic experimentation recorded as experiment code. Moreover, interpreters are executed in separate processes, which provides better isolation and fault tolerance (i.e. a crash o a single interpreter does not influence the others). The Experiment Workbench Layer and the Experiment Execution Layer are in constant communication via the SSH protocol family. We also intend to allow direct user access to the Experiment Execution Layer though bare SSH and SCP.

The next layer consists of Gems which are libraries/modules/utilities invoked by experiments at runtime. The Gem Layer provides APIs for experiment developers, enabling programmatic access to underlying resources (Grid, clusters) and functionality exposed in the form of libraries or services. There are gems which are generic, such as the one which provides access to PBS batch system, or application specific ones, such as Gaussian.

The Grid Fabric Layer forms the lowest level of the infrastructure. It consists of grid resources available to Virtual Laboratory users, including clusters accessible through PBS, grids available through their dedicated middleware packages (e.g. gLite), external services (e.g. Web Services) and data sources (e.g. RDBMSs) which the users may explit of in the course of their research activities.

The entire Virtual Laboratory operates in a Single Sign On (SSO) mode, according to the accounts defined in the external authentication system and incorporating security policies involved in accessing Grid middleware. In the case of PL-Grid installation, it uses PL-Grid LDAP directory so that the virtual laboratory is automatically accessible to all registered PL-Grid users.

The four layers of the Virtual Laboratory are distributed and spread over physical computational nodes. The Experiment Workbench Layer operates on the so-called Portal Host. The

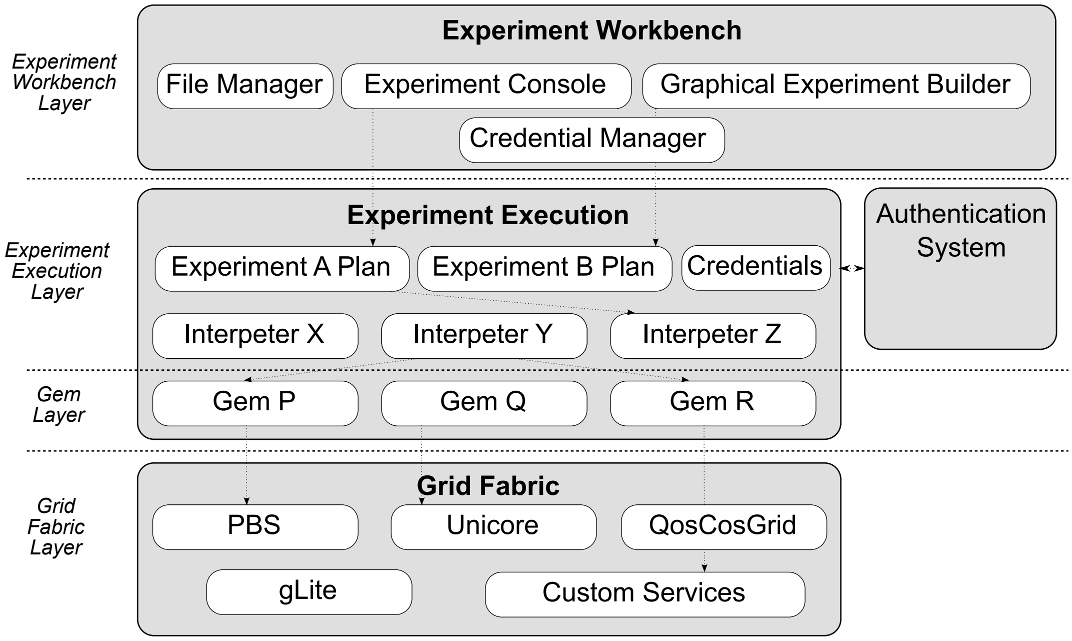


Figure 2. Architecture of the Virtual Laboratory including logical layers (captions on the left-hand side), main modules and their dependencies.

Experiment Execution Layer and the Gem Layer reside in one or more Experiment Hosts, which can be one of the front-end machines (user interfaces) of the clusters. The Grid Fabric Layer spans many distributed computational resources. The Portal Host and the Experiment Host may reside on distinct machines (in order to improve scalability); however they may also operate on a single host when system compactness is more important.

VI. GridSpace – Implementation

The presented Virtual Laboratory, is based on the core technology, called GridSpace, first conceived as part of the ViroLab project [2] and currently being extended within the scope of the PL-Grid. While still supporting the earlier version of the GridSpace framework, in GridSpace 2 we implement the new functionalities which focus on interactive and exploratory programming [6], together with a fully Web-based user interaction model.

In addition to the requirements specified in section III, there are some technical goals, which were not satisfied by earlier versions of the Virtual Laboratory. They include support for multiple scripting languages and more interactive and ad-hoc scripting capabilities. Non-functional requirements such as performance, maintability and ease of use have also been taken into account, motivated by the need to deliver production-quality software to be deployed on the PL-Grid national infrastructure.

It is important to note that GridSpace is a generic environment rather than a specific application. This means that it can be used to set up a specific instance of the Virtual Laboratory in support of a specific application domain. It can be applied whenever existing software modules, interpreters etc. need to be combined with other components. The learning curve involved in porting an application to the platform should be minimized.

Another feature of GridSpace is that it exploits Web 2.0 mechanisms by facilitating application development, operation and provisioning. This means that the entire experiment development and execution cycle is accessible from within a web browser, and moreover, experiments and their results can be shared among community members.

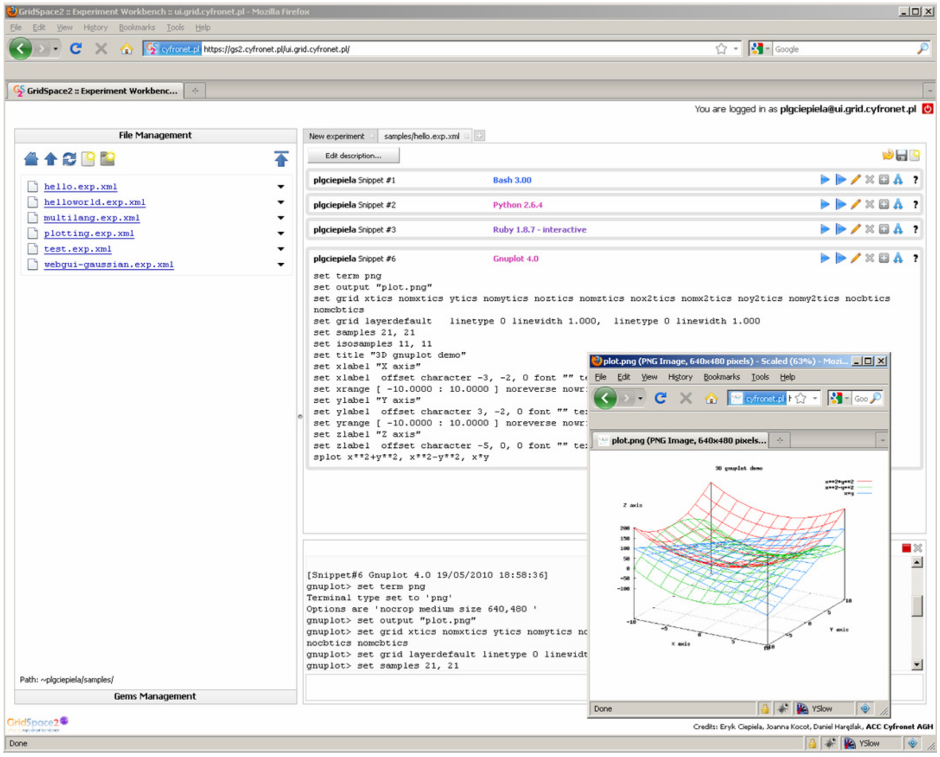
VII. Current Features of the Experiment Workbench

The main features of the current version of the Experiment Workbench include:

- File management
- Dedicated visualization openers
- Interactive interpretation of experiment script snippets
- Storing and sharing experiments using XML format
- Secure management of user credentials (passwords, certificates etc.)

The main exeriment workbench window is shown in Fig. 3. The file management tab is seen on the left, while the right-hand part of the application screen is taken up by the experiment console consisting of several experiment snippets. Above the console, in a separate window, a plugin for displaying graphical data is shown.

In addition to the basic features available directly in the Experiment Wokbench there are additional mechanisms which can be used in more advanced experiments. The first one is WebGUI, which allows experiments to expose dedicated Web interfaces. Another – semantic integration – enables the construction of domain-specific models and data exchange


GridSpace2 :: Experiment Workbench :: ui.grid.cyfronet.pl - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://gs2.cyfronet.pl/ui.grid.cyfronet.pl/
GridSpace2 :: Experiment Workbenc...
You are logged in as plgciepiela@ui.grid.cyfronet.pl
File Management
hello.exp.xml
helloworld.exp.xml
multilang.exp.xml
plotting.exp.xml
test.exp.xml
webgui-gaussian.exp.xml
Path: ~plgciepiela/samples/
Gems Management
New experiment
samples/hello.exp.xml
Edit description...
plgciepiela Snippet #1 Bash 3.00
plgciepiela Snippet #2 Python 2.6.4
plgciepiela Snippet #3 Ruby 1.8.7 - interactive
plgciepiela Snippet #6 Gnuplot 4.0
plot.png (PNG Image, 640x480 pixels) - Scaled (63%) - Mozi...
3D gnuplot demo
x**2+y**2
x**2-y**2
x*y
Z axis
Y axis
X axis
Done
Credits: Eryk Ciepiela, Joanna Kocot, Daniel Harężlak, ACC Cyfronet AGH

```
set term png
set output "plot.png"
set grid xtics nomxtics ytics nomytics noztics nomztics nox2tics nomx2tics noy2tics nomy2tics nocbtics
nomcbtics
set grid layerdefault   linetype 0 linewidth 1.000,  linetype 0 linewidth 1.000
set samples 21, 21
set isosamples 11, 11
set title "3D gnuplot demo"
set xlabel "X axis"
set xlabel  offset character -3, -2, 0 font "" te
set xrange [ -10.0000 : 10.0000 ] noreverse nowr
set ylabel "Y axis"
set ylabel  offset character 3, -2, 0 font "" te
set yrange [ -10.0000 : 10.0000 ] noreverse nowr
set zlabel "Z axis"
set zlabel  offset character -5, 0, 0 font "" te
splot x**2+y**2, x**2-y**2, x*y
```

```
[Snippet#6 Gnuplot 4.0 19/05/2010 18:58:36]
gnuplot> set term png
Terminal type set to 'png'
Options are 'nocrop medium size 640,480 '
gnuplot> set output "plot.png"
gnuplot> set grid xtics nomxtics ytics nomytics nc
nocbtics nomcbtics
gnuplot> set grid layerdefault linetype 0 linewidt
gnuplot> set samples 21, 21
```

Figure 3. Experiment Workbench screen, including experiment snippets and a sample result plot.

formats to facilitate colaboration and data sharing between experiments and users.

A. File management

At the present stage of development, the GridSpace 2 platform provides basic but stable functionallity which presents a sound basis for further expansion. The most fundamental feature is to enable users to manage their files on an experiment host through an HTTPS interface, enabling basic file system operations as well as uploading, downloading and accessing files via URLs. Data elements and experiment files are assigned globally unique URLs, which facilitate access, annotations and linking resources in a weblike manner. The URLs are given as: https://experiment.workbench.name/experiment.host.name/files/username/path/to/a/file. The Experiment Workbench handles these URLs by accessing the remote experiment host in the scope of the authenticated session. An SSH connection is established with the host, using the provided login/password pair. Following authentication, the workbench performs all file system operations via SCP/SFTP on behalf of the end user. Therefore, authorization relies on file access policies of the operating system residing on the experiment host.

B. Visualization openers

Files served by the Experiment Workbech can be downloaded to the end-user's desktop or viewed and edited within a web browser using so-called openers. The mechanism follows and extends the idea URL-accessible files by introducing the HTTPS GET param "opener", yielding URLs such as https://experiment.workbench.name/experiment.host.name/files/username/path/to/a/file?opener=openerName. The Experiment Workbench handles such requests by serving a page with an embedded opener applet instead of the file itself. The opener applet is configured on the fly, using the input URL, so that it is able to get and put a file using the above mentioned HTTPS interface. As the opener applet launches within the web browser, the authenticated session context (implemented using cookies) covers its function as well.

C. Interactive interpretation of experiment script snippets

The Experiment Workbench implements the exploratory programming paradigm through interactive interpretation of experiment parts called snippets. A snippets is an atomic planning and execution unit of an experiment. Exploratory development involves experiment planning and execution, both of which start at the same time and proceed in parallel. During the exploration process snippets can be incremetally added to the experiment plan, which, in turn, can be incrementally executed in a snippet-by-snippet manner. If the experiment plan calls for a change in a snippet which has alrady been evaluated in a run, the experiment needs to be reexecuted or rolled back (the latter functionality is, however, a challanging issue which will be addressed in the scope of further research).

D. Storing and sharing of experiments using the XML format

The experiment plan is modeled as a sequence of snippets. Each snippet is associated with an interpreter required for its execution. Interpreter specification includes a command, a set of environment variables and a prompt character sequence which is required in order to enable identification of script line evaluation completeness. Interpreters can be intaractive (capable of evaluating snippets line by line) or batch-oriented (the whole snippet is sent to the interpreter followed by an EOT character, upon which the environment waits for output). Any executable capable of operating in one of the above modes may be configured as a GridSpace 2 interpreter. The idea of an experiment file is that it presents a complete and standalone artifact sufficient for performing the experiment run. As long as it is contained in a single file, it can be associated with a global URL, which makes the experiment an addressable, shareable and linkable resource within the Web 2.0 space. Along with snippet code and interpreter specifications, the experiment XML file contains metadata including the name of the experiment, its description, its creation date, author names and comments. In addition, each snippet may specify a list of so-called secrets, i.e. data elements (such as passwords) which represent user credentials. This data should not be stored in experiment files but instead retrieved from the local user's wallet each time a given experiment is run. The Workbench provides a user-friendly way to manage secrets and use them in experiments while the GridSpace platform facilitates secure storage of credentials and other sensitive data.

A sample experiment XML file in a simplified notation is shown in Fig. 4. It includes a metadata tag, where user comments can be added during experiment evolution. An interpreter definition is presented in line 26 where it is possible to specify the command to initiate an interactive session using PBS. In lines 31–36 of this "hello world" example we can see the sample code in the specified languages.

```
1  experiment:
2    metadata:
3      expname: Hello Experiment
4      author: plgciepiela
5      description:
6         Demo example that prints out hello messages.
7      comments:
8        comment:
9          author: plgciepiela
10         payload: Very simple example, just demo.
11   interpreters:
12     interpreter:
13        cmd="bash --noprofile --norc"
14        interactive="true"
15        name="Bash 3.00"
16        prompt1="$
17        " prompt2="&gt; "
18      envvar: name="PS1" value="$ "
19     interpreter:
20        cmd="/software/local/bin/python"
21        interactive="true"
22        name="Python 2.6.4"
23        prompt1="&gt;&gt;&gt; "
24        prompt2="... "
25     interpreter:
26        cmd="qsub -I -q plgrid -S /bin/bash -v PS1=$"
27        interactive="true"
28        name="PBS - Bash 3.00"
29        prompt1="$"
30        prompt2="&gt; "
31   snippet: id="1" interpreterName="Bash 3.00"
32     code: echo "Hello, Bash"
33   snippet: id="2" interpreterName="Python 2.6.4"
34     code: print("Hello, Python")
35   snippet id="3" interpreterName="PBS - Bash 3.00"
36     code: echo "Hello, Bash via PBS"
```

Figure 4. Sample "Hello world" experiment XML file (tags are represented as bold text for clarity).

E. WebGUI

In order to enrich user experience a WebGUI integration tool is available. It provides well-defined frames for incorporating external web applications in the experiment execution flow. The tool enables experiment creators to plan interactions with the end user and either retrieve additional data or present intermediate experiment results. For integration with external web applications simple JSON communication is used, which makes the process of creating new or modifying existing web applications straightforward. For less demanding experiment creators who do not wish to create or reuse external applications, a generic web UI implementation is available. Through a simple JSON-based definition, available from the experiment code, a graphical interface can be spawned and presented during experiment execution. This implementation allows for building web forms using standard controls (e.g. text fields, text areas, radio boxes, etc.) In addition, a rich text editor control is available.

F. Semantic integration

Among the goals of the Virtual Laboratory, as stated in Section III, is the provision of a generic technology, supporting scientists from specific application domains. Since each of the *in-silico* experiments supported by our platform comes from a specific field of science, the need for domain-specific data models is clear. The semantic integration concept [20] is a method of building application-specific data models and cross-combining them with protocols and tools developed for the application environment. In other words, semantic integration helps scientific developers support structures and taxonomic characteristic for a given field of science via generic storage mechanisms and generic information exchange protocols.

The incorporation of semantic integration in the presented Virtual Laboratory provides a means of storing data and metadata for bioinformatics applications (such as protein pocket finding), as well as for computational chemistry applications running Gaussian and GAMESS. In the former case it is used to store and publish several gigabytes of data produced in high-throughput computations involving numerous proteins. In the later case it helps store and exchange metadata for the output files generated by various chemistry packages. Since the Virtual Laboratory application pool is still being extended, we can expect that the semantic integration solution will eventually cater to other scientific domains as well.

VIII. Working with GridSpace and Examples of Usage

Virtual Laboratory defines a specific model of interacting with applications. The main procedure for preparing an experiment is as follows:

1) The user identifies a procedure (process, workflow) that involves manual use of a number of software pieces which could benefit from (semi-)automation, making them easy to use for the user and for the research team.
2) The user writes this procedure in a stepwise fashion where each step is decided upon by viewing the outcome of previous steps.
3) At each step the user takes advantage of one of a number of programming languages, platforms or programs which are the most suitable for the current purpose.
4) Following each step the user may open the retrieved files with the available tools (e.g. display a graph, visualize molecules, show text content etc.)
5) The user can retrace his/her steps in order to find the best path to the solution.
6) The user can save the current sequence of steps (i.e. the experiment) and open it later for further development.
7) Having discovered the right sequence of steps, the user can save the experiment again and specify the group of users who will be allowed to run or further modify that experiment.
8) The user can send a link to the experiment web application to his/her group.
9) The link leads to a page where, following successful login, other users can run the web application or open it in an experiment editor.
10) The user can enrich experiments with custom graphical user interfaces which collect input and display results.

As an example of use we can present an application from the chemistry domain involving the study of aqueous aminoacid solutions. The analysis process is a workflow which involves multiple steps realized using many tools, languages and libraries. First, Packmol [21] is used to perform molecular dynamics simulations for animoacid aggregation in the presence of water. The resulting solution is visualized with Jmol [22] and can be manually checked prior to further processing, i.e. computing a spectrum using the Gaussian [18] tool. In order to extract spectrum-related information from the Gaussian output file we need to use the CCLIB [23] library written in Python. Finally, spectrum information can be visualized as a plot using GnuPlot.

In addition to the above, many other programming languages and tools can be configured in the Experiment Workbench and thus made available for experiment developers.

Our sample installation of the Experiment Workbench supports a set of interpreters and tools including Bash 3.00, Python 2.6.4, Perl 5.8.5, Ruby 1.8.7, Packmol, Gaussian 09 and Gnuplot 4.0. All interpreters can be launched directly on the experiment host or through the Torque Portable Batch System.

IX. Conclusions and Future Work

In this paper we have presented the main requirements, concepts and current status of the Virtual Laboratory based on GridSpace platform. Although based on experience gained in the course of the ViroLab project, GridSpace 2 constitutes a novel framework. Its main advantage is support for exploratory programming, where each experiment consists of snippets programmed interactively in multiple programming languages using a web console. The Experiment Workbench allows interactive experiment development, file manegement and experiment sharing. The Virtual Laboratory also supports web-based graphical user interfaces and semantic integration.

GridSpace 2 has been made available for the users of the Pl-Grid project for beta testing. Preliminary feedback from bioinformatics and computational chemistry applicaiton domains shows promising results. The final release and integration with the PL-Grid infrastructure is planned for the end of 2010. The continuously updated beta installation of the Virtual Laboratory has been made available to the PL-Grid users and is accessible at the Virtual Laboratory website [24]. More information about the GridSpace 2 technology, including demos and presentations can be found at [25].

Future work will focus on enhancing the usability and security features. One of the planned enhancements involves development of a graphical tool for constructing experiments with hierarchical snippet trees. Another will provide handling of multiple security credentials to facilitate access to heterogeneous middleware systems and data sources. We are also adding support for more application-specific gems, interpreters and visualization tools to extend the range of supported application domains.

Acknowledgments

The research presented in this paper has been partially supported by the European Union within the European Regional Development Fund program no. POIG.02.03.00-00-007/08-00 as part of the PL-Grid project (www.plgrid.pl) and ACC Cyfronet AGH grant 500-08. Maciej Malawski acknowledges support from the UDA–POKL.04.01.01-00-367/08-00 AGH grant.

References

[1] U. Schwiegelshohn, R. M. Badia, M. Bubak, M. Danelutto, S. Dustdar, F. Gagliardi, A. Geiger, L. Hluchy, D. Kranzlmueller, E. Laure, T. Priol, A. Reinefeld, M. Resch, A. Reuter, O. Rienhoff, T. Rueter, P. Sloot, D. Talia, K. Ullmann, R. Yahyapour, and G. von Voigt, "Perspectives on grid computing," *Future Generation Computer Systems*, vol. In Press, Corrected Proof, pp. 1104–1115, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/B6V06-5046M26-9/2/48b1a0c4be91df6f89554f74e94ae792

[2] ViroLab team at CYFRONET, "The ViroLab Virtual Laboratory Website," 2009, http://virolab.cyfronet.pl.

[3] M. Bubak *et al.*, "Virtual laboratory for development and execution of biomedical collaborative applications," in *Proceedings of the 21st IEEE CBMS, June 17-19, 2008, Jyväskylä, Finland.* IEEE Computer Society, 2008, pp. 373–378.

[4] M. Bubak, M. Malawski, T. Gubala, M. Kasztelnik, P. Nowakowski, D. Harezlak, T. Bartynski, J. Kocot, E. Ciepiela, W. Funika, D. Krol, B. Balis, M. Assel, and A. T. Ramos, "Virtual laboratory for collaborative applications," in *Handbook of Research on Computational GridTechnologies for Life Sciences, Biomedicine and Healthcare*, M. Cannataro, Ed. IGI Global, 2009, ch. XXVII, pp. 531–551.

[5] P. Nowakowski, D. Harezlak, and M. Bubak, "A new approach to development and execution of interactive applications on the grid," in *8th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2008), 19-22 May 2008, Lyon, France.* IEEE Computer Society, 2008, pp. 681–686.

[6] Y. Gil, V. Ratnakar, E. Deelman, G. Mehta, and J. Kim, "Wings for pegasus: Creating large-scale scientific applications using semantic representations of computational workflows," in *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*. AAAI Press, 2007, pp. 1767–1774.

[7] Z. Zhao, A. Belloum, and M. Bubak, "Special section on workflow systems and applications in e-science," *Future Generation Comp. Syst.*, vol. 25, no. 5, pp. 525–527, 2009.

[8] D. Hull, K. Wolstencroft, R. Stevens, C. Goble, M. R. Pocock, P. Li, and T. Oinn, "Taverna: a tool for building and running workflows of services," *Nucl. Acids Res.*, vol. 34, no. suppl_2, pp. W729–732, July 2006. [Online]. Available: http://dx.doi.org/10.1093/nar/gkl320

[9] B. Ludäscher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. B. Jones, E. A. Lee, J. Tao, and Y. Zhao, "Scientific workflow management and the kepler system," *Concurrency and Computation: Practice and Experience*, vol. 18, no. 10, pp. 1039–1065, 2006.

[10] D. De Roure, C. Goble, and R. Stevens, "The design and realisation of the myexperiment virtual research environment for social sharing of workflows," *Future Generation Computer Systems*, vol. 25, no. 5, pp. 561–567, May 2009. [Online]. Available: http://eprints.ecs.soton.ac.uk/15709/

[11] M. Wilde, I. T. Foster, K. Iskra, P. H. Beckman, Z. Zhang, A. Espinosa, M. Hategan, B. Clifford, and I. Raicu, "Parallel scripting for applications at the petascale and beyond," *IEEE Computer*, vol. 42, no. 11, pp. 50–60, 2009.

[12] F. Gioachin and L. V. Kale, "Dynamic high-level scripting in parallel applications," in *Parallel and Distributed Processing Symposium, International*. Los Alamitos, CA, USA: IEEE Computer Society, 2009, pp. 1–11.

[13] G. Hohpe and B. Woolf, *Enterprise Integration Patterns : Designing, Building, and Deploying Messaging Solutions*. Addison-Wesley, 2004.

[14] S. Weerawarana, F. Curbera, F. Leymann, T. Storey, and D. F. Ferguson, *Web Services Platform Architecture : SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More*. Prentice Hall PTR, March 2005.

[15] W. Tan, P. Missier, R. Madduri, and I. Foster, "Building scientific workflow with taverna and bpel: A comparative study in cagrid," in *Service-Oriented Computing âĂŞ ICSOC 2008 Workshops*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 118–129. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01247-1_11

[16] M. Malawski, T. Bartyński, and M. Bubak, "Invocation of operations from script-based grid applications," *Future Gener. Comput. Syst.*, vol. 26, no. 1, pp. 138–146, 2010.

[17] P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. L. de Hoon, "Biopython: freely available python tools for computational molecular biology and bioinformatics," *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, June 2009. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/btp163

[18] Gaussian, Inc., "Gaussian," 2010, http://www.gaussian.com.

[19] H. McWilliam, F. Valentin, M. Goujon, W. Li, M. Narayanasamy, J. Martin, T. Miyar, and R. Lopez, "Web services at the european bioinformatics institute-2009." *Nucleic acids research*, vol. 37, no. Web Server issue, pp. W6–10, July 2009. [Online]. Available: http://dx.doi.org/10.1093/nar/gkp302

[20] T. Gubala, M. Bubak, and P. M. Sloot, "Semantic integration of collaborative research environments," in *Handbook of Research on Computational Grid Technologies for Life Sciences, Biomedicine and Healthcare*, M. Cannataro, Ed. IGI Global, 2009, ch. XXVI, pp. 514–530.

[21] L. Martínez, R. Andrade, E. G. Birgin, and J. M. Martínez, "Packmol: a package for building initial configurations for molecular dynamics simulations." *Journal of computational chemistry*, vol. 30, no. 13, pp. 2157–2164, October 2009. [Online]. Available: http://dx.doi.org/10.1002/jcc.21224

[22] A. Herraez, "Jmol: an open-source Java viewer for chemical structures in 3d," 2010, http://www.jmol.org/.

[23] N. M. O'Boyle, A. L. Tenderholt, and K. M. Langner, "cclib: a library for package-independent computational chemistry algorithms." *Journal of computational chemistry*, vol. 29, no. 5, pp. 839–845, April 2008. [Online]. Available: http://dx.doi.org/10.1002/jcc.20823

[24] Virtual Laboratory Team at CYFRONET, "The PL-Grid Virtual Laboratory Website," 2010, http://wl.plgrid.pl.

[25] DICE Team at CYFRONET, "GridSpace 2 Website," 2010, http://gs2.cyfronet.pl.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Modelling, Optimization and Execution of Workflow Applications with Data Distribution, Service Selection and Budget Constraints in BeesyCluster

Pawel Czarnul
Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, Poland
Email: pczarnul@eti.pg.gda.pl, `http://fox.eti.pg.gda.pl/~pczarnul`

***Abstract*—The paper proposes a model which allows integration of services published by independent providers into scientific or business workflows. Optimization algorithms are proposed for both distribution of input data for parallel processing and service selection within the workflow. Furthermore, the author has implemented a workflow editor and execution engine on a platform called BeesyCluster which allows easy and fast publishing and integration of scientific and business services. Several tests have been implemented and run in BeesyCluster using services for a practical digital photography workflow with and without budget constraints. Two alternative goals are considered: minimization of the execution time with a budget constraint or a linear combination of cost and time.**

I. Introduction

SERVICE integration has become the main focus in distributed environments. Firstly, Service Oriented Architecture (SOA) promoting loosely coupled services interacting with each other, secondly Open Grid Service Architecture in the context of grid computing laid fundamental concepts for modern distributed applications. In both cases, workflow management systems using Web Services and grid middlewares allow composition of individual services into complex workflows. In the classic approaches, tasks are either mapped to available resources or available services are assigned to tasks so that QoS metrics for particular tasks or for the whole workflow are optimized. This can be minimization of the execution time with a budget constraint on the costs of resources or services used. Furthermore, many of the practical workflows are data-intensive and require data partitioning for parallel processing. From this point of view the author proposes a model with data partitioning for parallel execution and service selection, both important for practical workflows.

From the infrastructure point of view, it is desirable for a service provider to publish scientific or business services easily and instantly, define prices and other QoS parameters as well as be able to manage the service. For the client, it is important to be able to integrate such services from various providers. This requires selection of best services considering imposed QoS constraints and the goal followed by execution of selected services. The author has implemented these in BeesyCluster which satisfies all of these requirements for scientific and business environments.

II. Related Work and Motivations

Firstly, *task scheduling* considers mapping of workflow tasks to available resources so that workflow execution time is minimized [1], [2]. In the classic problem, task execution times are known in advance, neither data flows nor other metrics such as costs are usually considered. Some works take into account execution costs of particular resources [3]. Paper [4] considers resource requirements for tasks of the workflow.

Secondly, in *scheduling in utility grids* [5] or *workflow scheduling in grids* [6] for each task t_i we distinguish a set of services S_i out of which only one service is to be chosen to execute t_i. Other attributes such as service costs are considered [7]. As considered by [8] and [5] the goal is to find the best assignment of $t_i \rightarrow (s_k, t_{ik}^{st})$ where s_k is a service able to execute task t_i and t_{ik}^{st} is the starting time of execution of task t_i on service s_k. Execution of t_i and t_j on one s_k must not overlap and the workflow execution time should be minimized while keeping the cost of selected services below a predefined minimum. In the context of typical business interactions, the *QoS service selection/workflow composition* problem is stated. Compared to the scheduling problem, many more quality attributes are considered without dependencies between or overlapping of services executing different tasks. The goal is to select proper services so that a function of QoS metrics such as: execution time, cost, availability [9], [10], [11], accessibility [10], fidelity [12] or conformance [10], security [10], reputation [9] is minimized possibly also with additional conditions imposed on some of them [11], [13]. For instance, the cost of using the services must not exceed the given budget.

There exist several workflow management systems for grid computing. Paper [2] describes and compares Gridbus, Kepler [14], Pegasus [15], Triana [16], P-GRADE [17], Directed Acyclic Graph Manager (DAGMan), ICENI, GridFlow, GrADS, Askalon, UNICORE, Taverna, GridAnt. These grid-oriented systems mainly use middlewares such as Globus Toolkit, Grid Application Toolkit or other resource management systems etc. for running jobs. More business oriented environments focusing on execution of BPEL or semantic service discovery and compositon such as Meteor-S [18] usually do not offer support for HPC environments although [19] presents how to use BPEL for grid environments.

Such workflow management systems support data parallelism and various operations on the data [20], [21]. In Gridbus, input parameters can be defined as parameter values, files or data streams [20]. In Process Networks in Kepler the workflow is driven by data availability and can be used for parallel processing on distributed systems [22]. Since the service may receive data as it runs and possibly from various predecessors, data processing can be defined in many ways on the input data. In the one-to-one composition pattern successive portions of input data from two input sources are processed pairwise [21]. Paper [21] demonstrates that MOTEUR can handle one-to-one and all-to-all patterns as well as workflow, data and service parallelism and is compared to Taverna, Kepler and Triana which lack some of these features. [23] optimizes workflow makespan also considering the possibility of dynamic deployment of services on various resources to save large data communication costs.

From the point of view of the model, the contribution of this work is a combined model (Section III) with both data distribution for parallel computing by parallel tasks and selection of services to optimize various QoS goals and meet QoS constraints. The consumer constructs a workflow i.e. a complex scenario composed out of simple tasks. It is assumed that the workflow graph can contain both sections with parallel tasks for among which input data is distributed for parallel processing. In this case each task has a single service assigned to it. Also, the graph may contain tasks with alternative services out of which one needs to be chosen to perform the given task. As an example, the workflow shown in Figure 1 allows parallel processing of several input digital RAW images by several filters. These can be executed by services created from free applications such as `dcraw`, `RAWtherapee` or possibly offered by companies who offer own paid image processing software. A comparison of RAW converters is available at `http://www.photozone.de/conclusion-on-going`. It finally produces a Web album out of images previously processed in parallel. The problem is to find both: data distribution – partition input data for parallel processing by parallel paths to speed up workflow execution and service selection – a service to execute each task – for each task t_i there may be one or more services (s_{ij}) capable of executing the task at price per unit of data c_{ij} (one out of two web album generation services $s_{19\ 0}$, $s_{19\ 1}$ can be chosen for task t_{19}). The goal is to minimize one of the following: $v_{MIN\_T\_C\_BOUND}$ – minimization of the workflow execution time with a constraint that the total cost of selected services does not exceed the given budget or, v_{MIN_TC} – minimization of a linear combination of workflow execution time and the total cost of services. Services may be offered by independent providers for potentially various prices and each service is installed on a particular cluster or server of a predefined speed. Incompatibility between formats used by various providers is considered as additional cost in the model. If there is a budget constraint set then more tasks will be sent along cheaper paths and more expensive paths will not be used. What is important, the model proposed by the author and presented in Section III considers alternative approaches as to how data is passed between tasks. As discussed in Section III-A, either the next task starts when all data has been processed by preceding tasks or processing data in streams for overlapping communication and computations is proposed. In [24] the author has presented how to choose services at runtime for highly changeable environments rather then before workflow execution considered in this work.

From the infrastructure point of view, the contribution is the author's implementation of the model in BeesyCluster (Section VI) taking advantage of features of the latter. Secondly, the paper demonstrates usefulness of the solution for a real world workflow application (Section VII). BeesyCluster is the middleware for the workflow system described in Section VI in this work. BeesyCluster differs from other middlewares as it accesses services published by users through user accounts via SSH. This means that contrary to many other systems adding new clusters or registering new user accounts is a matter of seconds as requires just corresponding entries in the database regarding access. Similarly, each user can become a provider by publishing their parallel or sequential applications installed on various clusters as services in a matter of seconds in the file manager in BeesyCluster. This allows publication of scientific services from HPC clusters as BeesyCluster hides low level details like queueing systems. Also, businesses can publish services from their own servers attached to BeesyCluster. This also brings pricing for not only business but also scientific services and allows composition of both types.

III. Model of the Workflow Scheduling with Data Distribution

Consequently, the author proposes a model with data distribution in which a directed graph $G(V, E)$ represents a workflow where nodes V correspond to tasks while edges E denote task dependencies. There is at least one starting node with initial data and one termination node which terminates computations. The model allows: a sequence – for connected tasks, a service connected to the successor will be executed only after the service selected for the predecessor has completed (t_1 and t_2 in Figure 1), fork – services associated with tasks following the forked task can potentially be executed in parallel (if run on separate processors – e.g. t_1, t_4, ..., t_{16} in Figure 1), join – the service selected for the task to which other tasks are connected will be executed only after each of the predecessors has finished (in Figure 1 t_{19} will execute only after t_3, t_6, ..., t_{18} have finished).

The model distinguishes the following: a set of services $S_i = \{s_{i0}, s_{i1}, ..., s_{i(|S_i|-1)}\}$ out of which only one must be selected to execute task t_i, c_{ij} – cost of processing a unit of data by service s_{ij}, P_{ij} – the provider of service s_{ij}, N_{ij} – the node service s_{ij} runs on while sp_n the speed of node n, d_{ij}^{in} and d_{ij}^{out} are the size of the input data and the size of the data produced by service s_{ij} bound with $d_{ij}^{out} = f_{t_i}(d_{ij}^{in})$. d_i denotes the size of data processed by task t_i. d_{ijkl} denotes the size of data to be sent from service s_{ij} to service s_{kl}. f_{t_i} denotes how the size of output data for task t_i depends on the size of input data. Then the required

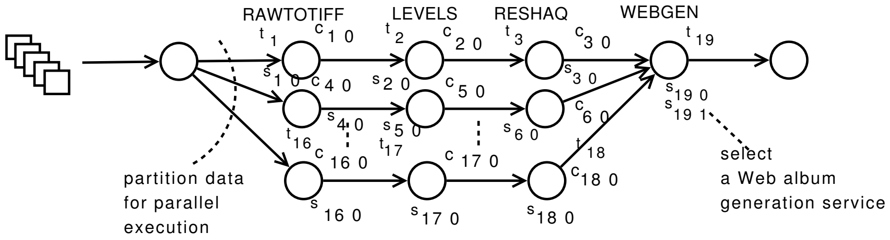


Figure 1: A Parallel Digital Photo Workflow with Budget Constraints

constraints on these variables would include: $d_i = \sum_j d_{ij}^{in}$, $d_{kl}^{in} = \sum_{i,j:(v_i,v_k)\in E} d_{ijkl}$, $d_{ij}^{out} = f_{t_i}(d_{ij}^{in})$. $d_{ij}^{in} > 0$ for only one selected service s_{ij} for task t_i. Output data from a task can be sent and/or partitioned into input files of successors. In particular, the following alternatives are possible: All Fork (AF) task i.e. t_{ij}: $\forall_{k:t_k\in Succ(t_i)} d_{ij}^{out} = \sum_l d_{ijkl}$ or Distribute Fork (DF) task i.e. t_{ij}: $d_{ij}^{out} = \sum_{l,k:(v_i,v_k)\in E} d_{ijkl}$. Then assuming $t_{ijkl}^{comm}(d_{ijkl}) = t_{N_{ij}N_{kl}}^{startup} + \frac{d_{ijkl}}{bandwidth_{N_{ij}N_{kl}}}$ – communication time of data of size d_{ijkl} sent from service s_{ij} and s_{kl}, $t_{ij}^{exec}(d_{ij}^{in})$ – the execution time of service s_{ij}, t_{ijkl}^{tr} – additional time for data conversion between output/input formats if connected services are offered by various providers, $t_i^{st} : i \in |V|$ – the time at which service s_{ij} chosen to execute t_i starts processing it, we have $\forall_{i,k:(v_i,v_k)\in E}\ t_k^{st} \geq t_i^{st} + \sum_j t_{ij}^{exec} + \sum_{j,l} t_{ijkl}^{comm} + \sum_{j,l} t_{ijkl}^{tr}$. t_{ij}^{exec} will be larger than 0 only for one j. Similarly, for the given i and k t_{ijkl}^{comm} and t_{ijkl}^{tr} will be larger than 0 only for one pair of l and k since only one service per node i and one per node k will be selected. Additionally, $t^{workflow} = t_{termination}\ \ not\exists_q(v_{termination}, v_q) \in E$

We consider two alternative minimization goals where $t^{workflow}$ is the time when the last service finishes:

1) min $v_{MIN\_T\_C\_BOUND} = t^{workflow}$ while adding a constraint on the total cost i.e. $\sum d_{ij}^{in} c_{ij} < B$ where B is the budget (problem MIN_T_C_BOUND).
2) min $v_{MIN\_TC} = \alpha t^{workflow} + \sum d_{ij}^{in} c_{ij}$ (problem MIN_TC), $\alpha > 0$.

A. Data Processing

For data processing and communication between tasks, three possibilities are considered:

1) synchronized: as assumed above, before execution of the given task starts, all data processed by previous tasks must be sent and ready,
2) overlapping: instead of $t_i \rightarrow t_j$ we can introduce parallel paths $t_i \rightarrow t_{j'}$, $t_i \rightarrow t_{j''}$, ..., $t_i \rightarrow t_{j'^n}$ such that every $t_{j'^l}$ is in fact the same task as t_j and has the same services assigned to it as t_j and a constraint that the same service is chosen for every one of them. Instead of passing whole data d_i to t_j, $\frac{d_i}{n}$ can be passed to every following task independently. This means that services for $t_{j'^l}$ will be executed faster. Secondly, if copying to every $t_{j'^l}$ is delayed compared to $t_{j'^{l-1}}$ this results in overlapping communication and computations.
3) streaming: after receiving a unit of data, each task processes it and passes to a following task(s) and does so for every unit.

IV. Algorithms

The author has proposed and implemented three different algorithms to solve the problem:

1) genetic algorithm (GA) – a chromosome encodes both the assignment of services to tasks and data distribution among tasks. The algorithm does not impose linearity on the constraints but it comes at the cost of slower convergence compared to MGALP described next.
2) mixed genetic algorithm and linear programming (MGALP) – where the genetic algorithm is used to determine the assignment of services to tasks and fast linear programming [25] is used to determine optimal data distribution for the given schedule; a chromosome encodes just the assignment of services to tasks,
3) mixed integer linear programming (MILP) – the author adopts MILP traditionally used for solving the service selection problem and extends it for scheduling and solving data flows. The algorithm solves the problem optimally but at a high computational cost for large problem sizes. Similarly to [9], [18], [26], integer variables denote which service is selected for the particular task while real or integer variables sizes of data sent between tasks.

GA and MGALP are much faster for larger graphs at the cost of lower quality results. Due to space limitations, only GA is presented in more detail. For large workflows, the algorithm may take more steps to arrive at an acceptable data distribution or even such that make constraint $\sum d_{ij}^{inp} c_{ij} < B$ satisfied for goal MIN_T_C_BOUND. To make the starting solution better, for each of the initial chromosomes, linear programming was used to help find better data flows and following iterations used only crossover and mutation operators.

Each chromosome consists of the representation for the schedule shown in Figure 2.

1) the first $|V|$ numbers contain indexes of selected services for each task i.e. $0 \leq e_i < |S_i|$ $0 \leq i < |V|$. Initial values are chosen by a random selection of services for each task i.e.: $e_i = random()\ mod\ S_i$;
2) the last $|V|$ numbers contain ordering of the $|V|$ tasks. Namely, assuming tasks t_k, t_l are executed on one processor (i.e. services on one processor are chosen to

$$\underbrace{e_0 \cdots e_{|V|-1}}_{e_i \text{ – service for } t_i} \quad \underbrace{e_{|V|} \quad \cdots \quad e_{|2V|-1}}_{e_i \text{ – task id}}$$

Figure 2: GA's Representation

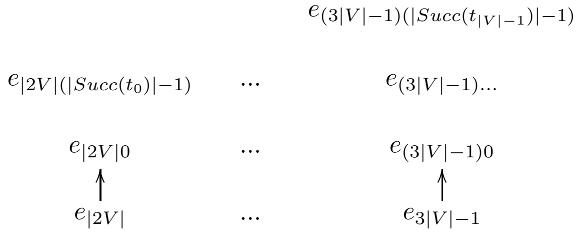


Figure 3: GA's Representation for Flow Distribution

execute them), t_l is executed after t_k iff $e_i = k$ $e_j = l$: $i < j$ $i \geq |V|$. Initial values are chosen randomly out of following tasks starting from the initial task. In the next step its successors are added to the pool of the tasks from which in the next step one will be chosen randomly and the procedure is repeated.

Additionally each chromosome contains a distribution of d_{ij}^{out} into data of the following tasks. For DF tasks data from t_i is simply copied to each task in $Succ(t_i)$. For AF tasks the structure shown in Figure 3 is used. For task t_i, since data is divided into all following tasks, there is an array of $|Succ(t_i)|$ $e_{(2|V|+i)j}$ elements for this task such that $\sum_{j=0}^{|Succ(t_i)|-1} e_{(2|V|+i)j} = 1$ and d_{ij}^{out} $e_{(2|V|+i)y} = \sum_{t_k : t_k \text{ is } y\text{-th successor of } t_i \text{ out of tasks in } Succ(t_i)} d_{ijkl}$ which denotes how output data for task t_i is split into input data of following tasks.

Crossover for service selection works as follows – a pivot (task) is selected randomly and services for tasks smaller than the pivot are taken from the first chromosome and for larger from the second chromosome. The order is taken from one of the two chromosomes with probabilities 0.35 and a new one is generated randomly with probability 0.3 (Figure 4). The algorithm for data distribution is shown in Figure 5. Function `SetRndDistr` assigns the flow to only one successor with probability 0.5 or divides it into more following tasks with smaller probabilities. This ensures that the algorithm can both direct the flow to a small and a large number of following tasks.

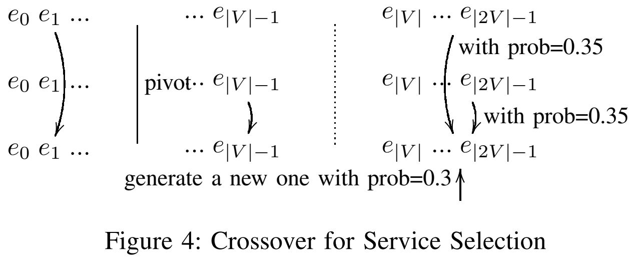

$e_0\ e_1 \cdots$
$\cdots e_{|V|-1}$
$e_{|V|} \cdots e_{|2V|-1}$
with prob=0.35
$e_0\ e_1 \cdots$
pivot$\cdots$ $e_{|V|-1}$
$e_{|V|} \cdots e_{|2V|-1}$
with prob=0.35
$e_0\ e_1 \cdots$
$\cdots e_{|V|-1}$
$e_{|V|} \cdots e_{|2V|-1}$
generate a new one with prob=0.3

Figure 4: Crossover for Service Selection

Mutation for service selection is applied as follows – a new service is selected for a task with the probability of 0.1 and a new order is generated with the probability of 0.02. Function `SetRndDistr` is invoked for a new distribution with probability 0.05.

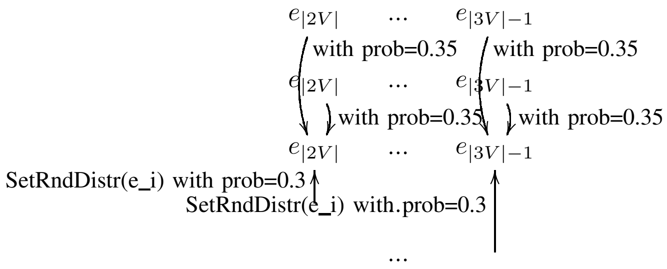

$e_{|2V|}$ $\cdots$ $e_{|3V|-1}$
with prob=0.35
with prob=0.35
$e_{|2V|}$ $\cdots$ $e_{|3V|-1}$
with prob=0.35
with prob=0.35
$e_{|2V|}$ $\cdots$ $e_{|3V|-1}$
SetRndDistr(e_i) with prob=0.3
SetRndDistr(e_i) with.prob=0.3
$\cdots$

Figure 5: Crossover for Data Distribution

This version can be adjusted to use only integer data sizes by rounding d_{ijkl} values which result from $e_{|2V|}$ to $e_{3|V|-1}$ distributions to integer numbers such that constraints III are satisfied and successive evaluation of the chromosome using integers.

V. Optimal vs Heuristic Algorithms

The author has run the three implemented algorithms for workflow graphs with varying numbers of nodes (tasks) and services assigned to each task. The main goal is to assess workflow sizes for which the MILP algorithm is able to return an optimal solution within a reasonable time limit of 20 seconds. For the given number of nodes, out of all possible edges between the nodes, 10% are chosen randomly. Then all nodes without predecessors are connected to the initial node and all without successors connected to the terminating node. For this particular test service costs are random from the $[0, 40]$ range, service execution coefficients $e_{ij} : f_{eij}(d) = e_{ij}d$ are random from the $[0, 4]$ range, the time coefficient $\alpha = 100$.

MILP is able to solve only small problems with a small number of tasks and services optimally. Since e.g. solving for MIN_T_C_BOUND is NP-hard, heuristic algorithms need to be used for large workflows. Since it is not possible to compare to an optimal algorithm, for MGALP and GA, we measure the improvement the algorithm is able to make over an initial value (for goal MIN_T_C_BOUND or MIN_TC) for the first chromosome of the population $v_{initial\ i}$. The improvement is computed as follows:

$$impr[\%] = \frac{100}{sc} \sum_{i=1}^{sc} \frac{v_{initial\ i} - v_{final\ i}}{v_{initial\ i}} \qquad (1)$$

where sc denotes the number of simulations. It must be noted that both for GA and MGALP the initial chromosomes are set using random schedules and data distribution computed using linear programming. This allows GA to have a reasonable start for tuning using only crossover and mutation. $v_{initial\ i}$ is always obtained by taking the value from the initial chromosome of the population. For MIN_T_C_BOUND it may not meet the cost bound though. If this is the case, $v_{initial\ i}$ is obtained using linear programming minimizing the cost and taking the workflow time obtained for the minimum cost (if the minimum cost is lower than the bound set). In the experiments B is set to $bestcost + \frac{worstcost - bestcost}{4}$ where $bestcost$ is the minimum possible workflow cost without considering the workflow time and $worstcost$ is the cost corresponding to minimization of only workflow execution time.

Figures 6 and 7 present which algorithm is best for a workflow with a particular number of tasks and services marked on the two axes respectively. Each algorithm is given 20 seconds and results are compared. Each measured number is an average from 50 runs for a particular input data set. MILP is best for small graphs as it generates optimal results. For larger graphs, either MGALP or GA are preferred. Colors denote the ratio of improvement over the initial solution of GA divided by the improvement for MGALP. The range of input data for which MILP completes in 20 seconds (marked as OPT) and thus is best is limited (white color). Light areas correspond to workflow sizes for which GA gives better results than MGALP.

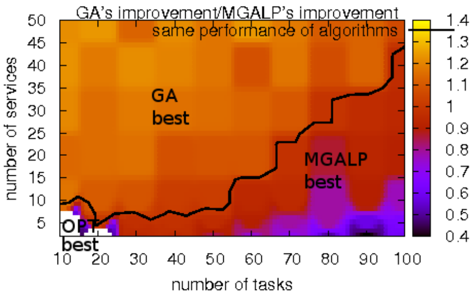


Figure 6: Comparison of OPT, MGALP, GA within 20 seconds vs Task Count and Average Service Count MIN_TC

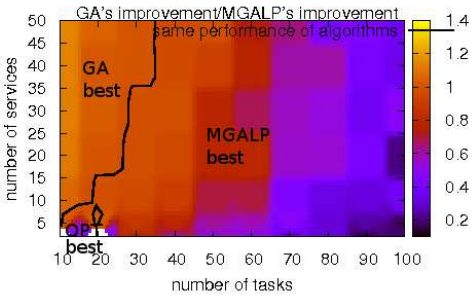


Figure 7: Comparison of OPT, MGALP, GA within 20 seconds vs Task Count and Average Service Count MIN_T_C_BOUND

VI. Modelling, Scheduling and Execution of Workflow Applications in BeesyCluster

The author has created an environment for modelling, scheduling and execution of workflow applications implementing the proposed model and algorithm. It is embedded in BeesyCluster. The latter is a JEE-based front-end and middleware which allows publishing and consuming services offered by various providers and consumers from locations managed by them. BeesyCluster, designed and co-developed by the author, was deployed at Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology for integration of clusters and laboratories. It is used for research and teaching high performance computing.

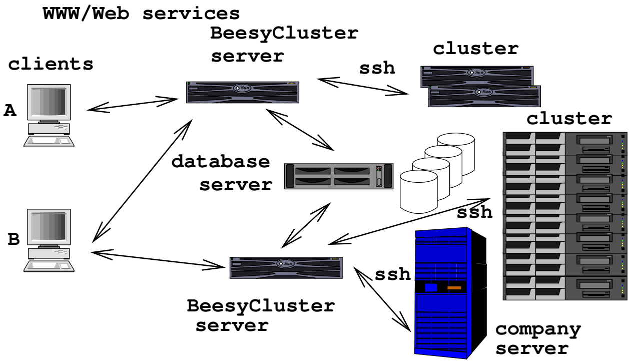


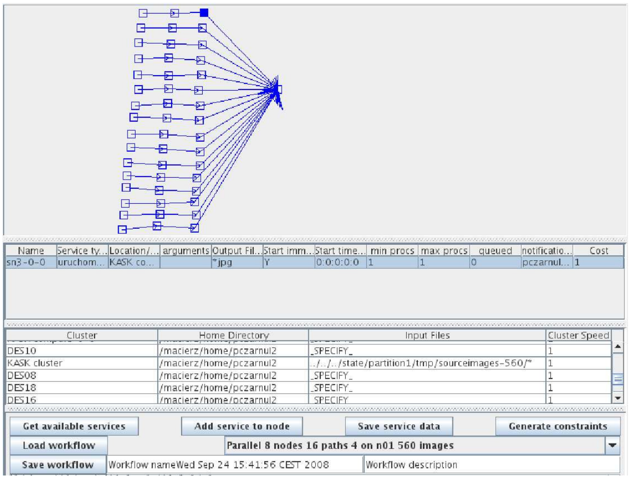
Figure 8: BeesyCluster Architecture

BeesyCluster allows the registered user to access accounts on various clusters or servers via WWW/Web Services (Figure 8). [27] presents details and performance of the Web Service interface. The system accesses these accounts via SSH. It allows to manage resources on these accounts allowing copying, editing files, management of directories and archives in a versioning system, compilation, queueing and even running applications interactively in a graphical mode via the browser.

Advantages of this complete environment with workflow support are as follows:

- the user can publish a sequential or a parallel (e.g. MPI) application from his/her user account as a service This brings business features to the scientific environment,
- companies can publish commercial applications as services from their servers,
- such services can be incorporated into workflows as described by the model,
- if the service is deployed on a cluster with a queueing system (e.g. PBS, LSF), the latter will be used transparently to the user,
- support for a complete workflow creation and execution cycle i.e.: workflow editing using a GUI, workflow optimization (service selection and data distribution), workflow execution in the real distributed environment.

The author has implemented a workflow editor in BeesyCluster shown in Figure 9 implementing the model proposed in Section III. It allows to define a workflow graph including tasks and dependencies, assign services to tasks, define input data which will be partitioned for parallel execution as well as one of the optimization goals. Data sizes including the size



Name	Service ty...	Location/...	arguments	Output Fil...	Start imm...	Start time...	min procs	max procs	queued	notificatio...	Cost
sn3-0-0	uruchom...	KASK co...		*jpg	Y	0:0:0:0:0	1	1	0	pczarnul...	1

Cluster	Home Directory	Input Files	Cluster Speed
DES10	/macierz/home/pczarnul2	_SPECIFY_	1
KASK cluster	/macierz/home/pczarnul2	../../../state/partition1/tmp/sourceimages-560/*	1
DES08	/macierz/home/pczarnul2	_SPECIFY_	1
DES18	/macierz/home/pczarnul2	_SPECIFY_	1
DES16	/macierz/home/pczarnul2	_SPECIFY_	1

Parallel 8 nodes 16 paths 4 on n01 560 images

Workflow nameWed Sep 24 15:41:56 CEST 2008

Workflow description

Figure 9: Workflow Editor in BeesyCluster

of input data and data flowing between services correspond to the numbers of files flowing between the services. Data distribution policies mentioned in Section III-A are possible. To run a workflow, BeesyCluster launches one of the algorithms implemented by the author to determine services for tasks and data distribution and executes the selected services handling data transfers between clusters and servers on which the services are installed. [24] and [28] present implementation details of the workflow execution engine which is also used in this work. However, [24] focuses on just-in-time service selection while [28] on incorporation of a checkpoint/restart mechanism for execution of scientific workflows.

VII. Compute and Data Intensive Digital Photography Workflow

The workflow shown in Figure 1 was modelled and tested in BeesyCluster. It shows a scenario in which there are three filters applied on input RAW images to generate a final web album to be published in the photographer's portfolio. Optimization problems include: partitioning a set of input images among services installed on nodes of various speeds, a trade-off between the budget for the execution and the execution time (some services are more expensive than others).

1) Testbed Environment and Services: The environment consists of 16 nodes installed at the Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology. The speeds of all the nodes used are identical for this application. One BeesyCluster server located at `https://lab527.eti.pg.gda.pl:10030/ek/AS_LogIn` was used.

The following services were used to execute the tasks (Figure 10): RAWTOTIFF – conversion from a RAW format to TIFF: each service is performed by script `dcraw -T $1` using the well known `dcraw` (`http://cybercom.net/~dcoffin/dcraw/`) converter, LEVELS – level adjustments so that whites, blacks and middle tones are mapped correctly: each service is performed by script `convert $1 -normalize $1` using the `ImageMagick` (`http://www.imagemagick.org/script/index.php`) tool, RESHAQ – resizing, sharpening and saving as a 97% JPEG image for further generation of a web album: each service is performed by script `convert $1 -resize 600x400 -sharpen 1x1.2 -quality 97% $1.jpg` using the `ImageMagick` tool, WEBGEN – generation of a Web album: two services are installed on only one node (are alternative) and performed by applications `album` (`http://marginalhacks.com/Hacks/album/`) and `jigl` (`http://xome.net/projects/jigl/`) respectively.

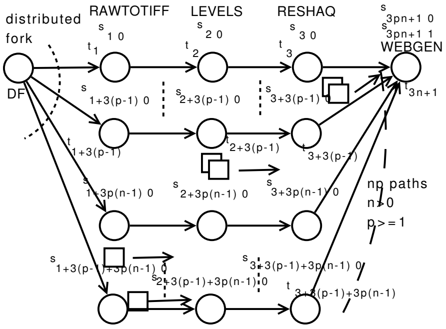


Figure 10: A Parallel Digital Photo Workflow with Several Paths per Node

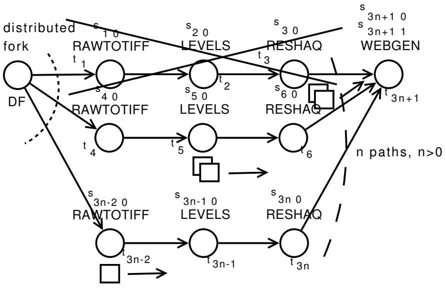


Figure 11: A Parallel Digital Photo Workflow with Budget Constraints

2) Workflow Configurations and Results: The following configurations were tested (optimization goals are noted): **MIN_TC: parallel** (Figure 10) – each filter is cloned into up up to $16p$ ($p \geq 1$) functionally equivalent t_{i+3k} $k \in \{0, 1, ..., 16p-1\}$ $i \in \{1, 2, 3\}$ tasks (Figure 10 also shows nodes the services are installed on). If $p = 1$ then each node waits for its batch of data to start processing. If $p > 1$ there are p services for each filter on a given node and it is possible to start processing faster as a smaller batch of input data can reach the service faster. Execution of services on one node is serialized. Each batch is processed independently allowing computations on the given node to start faster at the cost of running and managing more threads responsible for more paths. Since the cost of each service was set to the same value in this configuration the algorithm effectively minimizes the total execution time. Figures 12 and 13 present execution times and speed-ups obtained for 40, 160 and 560 input images

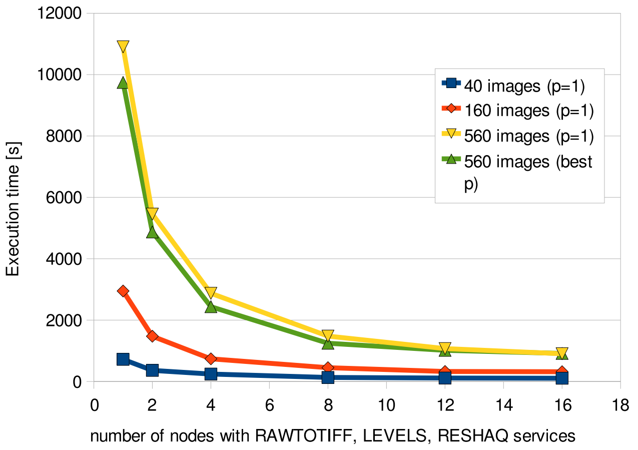


Figure 12: Digital Photography Workflow: Execution Times [s]

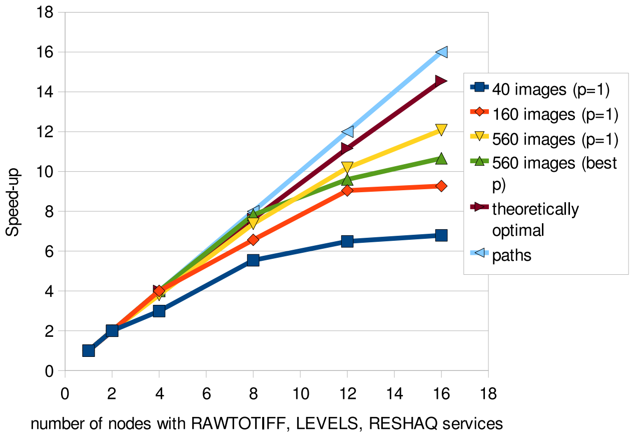


Figure 13: Digital Photography Workflow: Speed-up

Table I: Services and Cost per Processor Second for Testbed Clusters

			cost per second	
Cluster	services	node count	day	night
1	$s_{1\ 0}$ to $s_{24\ 0}$	8	20	10
2	$s_{25\ 0}$ to $s_{36\ 0}$	4	10	20
3	$s_{37\ 0}$ to $s_{48\ 0}$	4	15	15

(500MB, 2GB, 7GB of input data) respectively and from 1 to 16 nodes thus 1 to 16 paths.

As expected better speed-ups are obtained for a larger number of input images. For 560 images there is a comparison between $p = 1$ and best p for the given configuration. In the latter case, the time is lower as services can start processing earlier. BeesyCluster first sends a smaller batch to each service on a different node, then proceeds with following batches effectively overlapping communication and computations. It can be noted that the gain is much smaller for a larger number of nodes because the communication/transfer times between the tasks are much smaller for these configurations. Secondly, the node with the initial data may be a bottleneck for a large number of following tasks.

MIN_T_C_BOUND: parallel with budget constraints on the selection of services (Figure 11) – in this example the goal is to obtain the shortest possible execution time assuming the total cost of selected services does not exceed the given threshold. The literature proposes [29], [30] several pricing methods for services. Following the simplest strategy of charging per processor second and assuming the computational time can be divided between clusters we obtain the same cost irrespective on the number of processors used [31]. However, if costs per processor can vary, we are faced with a non-trivial cost-performance trade-off. Similarly to [30] the author has distinguished services installed on three logical clusters with 8, 4 and 4 nodes with prices for a processor second as shown in Table I. The prices vary depending on the time of day. Two configurations were tested: clusters 1 and 2 (total of 12 nodes), clusters 1, 2 and 3 (total of 16 nodes) both during the day and the night. For each configuration the total allowed budget (for the parallel paths) is varied from the minimum cost allowing full parallelization to through 0.9 to 0.8 of this cost. Figures 14 and 15 show that if the budget is limited then the more expensive paths are not used which results in higher execution times. For day simulations, services from cluster 1 will be ommitted first, for night simulations services from cluster 2.

VIII. Summary and Future Work

The paper formulated a problem on how input data should be distributed among parallel tasks and how services should be selected for other workflow tasks so that a QoS goal is optimized while possibly other QoS constraints are met. Both the model, algorithms and an execution engine were implemented in BeesyCluster allowing to consume distributed services from various providers. As an example, a data and compute intensive digital photo workflow was executed in this environment. It has been demonstrated that the solution achieves good speed-ups and is able to select services in such a way that the execution time is minimized and the total cost of selected services does not exceed the budget. Clearly, the environment can be used for other workflow applications which can be easily constructed from distributed services deployed in BeesyCluster.

Further work will include testing more complex workflows in more distributed environments with services installed on distant clusters.

Acknowledgment

Research sponsored by research grant N N516 383534 “Strategies for management of information services in distributed environments”.

References

[1] J. Blythe, S. Jain, E. Deelman, Y. Gil, K. Vahi, A. Mandal, and K. Kennedy, “Task scheduling strategies for workflow-based applications in grids,” in *CCGrid 2005. IEEE International Symposium on Cluster Computing and the Grid*, vol. 2, May 2005, pp. 759–767.

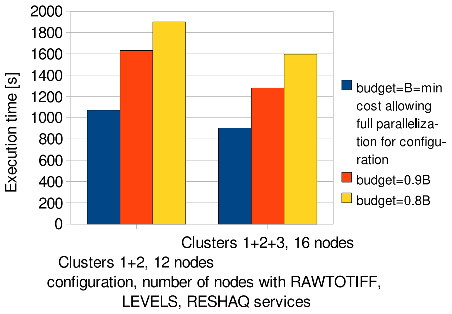


Figure 14: Digital Photography Workflow: Execution Time [s] under Cost Constraints: Day

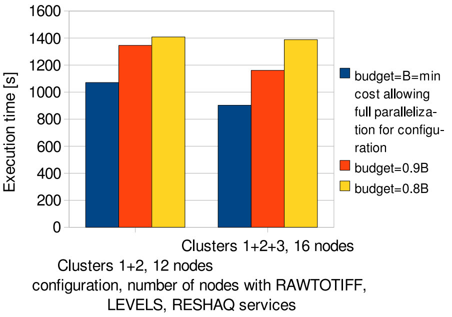


Figure 15: Digital Photography Workflow: Execution Time [s] under Cost Constraints: Night

[2] J. Yu and R. Buyya, “A taxonomy of workflow management systems for grid computing,” *Journal of Grid Computing*, vol. 3, no. 3-4, pp. 171–200, September 2005. [Online]. Available: http://dx.doi.org/10.1007/s10723-005-9010-8

[3] R. Sakellariou, H. Zhao, E. Tsiakkouri, and M. Dikaiakos, “Scheduling workflows with budget constraints,” in *Integrated Research in GRID Computing*, ser. CoreGRID, S. Gorlatch and M. Danelutto, Eds. Springer-Verlag, 2007, pp. 189–202. [Online]. Available: http://www.cs.man.ac.uk/ rizos/papers/coregrid2005a.pdf

[4] D. M. Quan and D. F. Hsu, “Mapping Heavy Communication Grid-Based Workflows Onto Grid Resources Within an SLA Context Using Metaheuristics,” *International Journal of High Performance Computing Applications*, vol. 22, no. 3, pp. 330–346, 2008. [Online]. Available: http://hpc.sagepub.com/cgi/content/abstract/22/3/330

[5] J. Yu and R. Buyya, “Scheduling scientific workflow applications with deadline and budget constraints using genetic algorithms,” *Scientific Programming Journal*, 2006, iOS Press, Amsterdam.

[6] Yingchun, X. Li, and C. Sun, “Cost-effective heuristics for workflow scheduling in grid computing economy,” in *GCC ’07: Proceedings of the Sixth International Conference on Grid and Cooperative Computing*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 322–329.

[7] J. Yu and R. Buyya, “A budget constrained scheduling of workflow applications on utility grids using genetic algorithms,” in *Workshop on Workflows in Support of Large-Scale Science, Proceedings of the 15th IEEE International Symposium on High Performance Distributed Computing (HPDC 2006)*, Paris, France, June 2006.

[8] J. Yu, R. Buyya, and C.-K. Tham, “Cost-based scheduling of workflow applications on utility grids,” in *Proceedings of the 1st IEEE International Conference on e-Science and Grid Computing (e-Science 2005), IEEE CS Press*, Melbourne, Australia, December 2005.

[9] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Sheng, “Quality driven web services composition,” in *Proceedings of WWW 2003*, Budapest, Hungary, May 2003.

[10] C. Patel, K. Supekar, and Y. Lee, “A QoS Oriented Framework for Adaptive Management of Web Service based Workflows,” in *Proceedings of the 14th International Database and Expert Systems Applications Conference (DEXA 2003)*, ser. LNCS, Prague, Czech Republic, September 2003, pp. 826–835.

[11] G. Canfora, M. D. Penta, R. Esposito, and M. Villani, “A Lightweight Approach for QoS-Aware Service Composition,” iCSOC 2004 forum paper, IBM Technical Report Draft.

[12] J. Cardoso, A. Sheth, and J. Miller, “Workflow quality of service,” LSDIS Lab, Department of Computer Science, University of Georgia, Athens, GA 30602, USA, Tech. Rep., March 2002.

[13] G. Canfora, M. D. Penta, R. Esposito, and M. Villani, “Qos-aware replanning of composite web services,” in *Procs. of 2005 IEEE International Conference on Web Services*, vol. 1. Res. Centre on Software Technol., Sannio Univ., Italy, July 2005, pp. 121–129.

[14] B. Ludascher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger-Frank, M. Jones, E. Lee, J. Tao, and Y. Zhao, “Scientific Workflow Management and the Kepler System,” *Concurrency and Computation: Practice & Experience, Special Issue on Scientific Workflows*, 2005.

[15] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, S. Patil, M.-H. Su, K. Vahi, and M. Livny, “Pegasus : Mapping Scientific Workflows onto the Grid,” in *Across Grids Conference*, Nicosia, Cyprus, 2004, `http://pegasus.isi.edu`.

[16] S. Majithia, M. S. Shields, I. J. Taylor, , and I. Wang, “Triana: A Graphical Web Service Composition and Execution Toolkit,” in *IEEE International Conference on Web Services (ICWS’04)*. IEEE Computer Society, 2004, pp. 512–524.

[17] *Parallel Grid Runtime and Application Development Environment, User’s Manual, ver. 8.4.2*, Laboratory of Parallel and Distributed Systems, MTA SZTAKI, Hungary.

[18] R. Aggarwal, K. Verma, J. Miller, and W. Milnor, “Constraint driven web service composition in meteor-s,” in *Proceedings of IEEE International Conference on Services Computing (SCC’04)*, 2004, pp. 23–30.

[19] R.-Y. Ma, Y.-W. Wu, X.-X. Meng, S.-J. Liu, and L. Pan, “Grid-enabled workflow management system based on bpel,” *Int. J. High Perform. Comput. Appl.*, vol. 22, no. 3, pp. 238–249, 2008.

[20] Gridbus Project, “Workflow language (xwfl2.0),” `gridbus.cs.mu.oz.au/workflow/2.0beta/docs/xwfl2.pdf`.

[21] T. Glatard, J. Montagnat, D. Lingrand, and X. Pennec, “Flexible and Efficient Workflow Deployment of Data-Intensive Applications On Grids With MOTEUR,” *International Journal of High Performance Computing Applications*, vol. 22, no. 3, pp. 347–360, 2008. [Online]. Available: http://hpc.sagepub.com/cgi/content/abstract/22/3/347

[22] “Kepler user manual,” May 2008.

[23] Z. Du, M. Wang, Y. Chen, Y. Ye, and X. Chai, “The triangular pyramid scheduling model and algorithm for pdes in grid,” *Simulation Modelling Practice and Theory*, vol. 17, no. 10, pp. 1678 – 1689, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/B6X3C-4WYJSNC-2/2/ae37494233580d168fdaadd2ec4e5b76

[24] P. Czarnul, “A JEE-based Modelling and Execution Environment for Workflow Applications with Just-in-time Service Selection,” in *proceedings of Grid and Pervasive Computing*, Geneva, Switzerland, May 2009.

[25] M. M. Syslo, N. Deo, and J. S. Kowalik, *Discrete Optimization Algorithms*. Prentice-Hall, 1983.

[26] R. Aggarwal, K. Verma, J. Miller, and W. Milnor, “Dynamic web service composition in meteor-s,” LSDIS Lab, Computer Science Dept., UGA, Technical Report, May 2004.

[27] P. Czarnul, M. Bajor, M. Fraczak, A. Banaszczyk, M. Fiszer, and K. Ramczykowska, “Remote task submission and publishing in beesycluster : Security and efficiency of web service interface,” in *Proc. of PPAM 2005*, Springer-Verlag, Ed., vol. LNCS 3911, Poland, Sept. 2005.

[28] P. Czarnul, “Integration of compute-intensive tasks into scientific workflows in beesycluster,” in *Computational Science – ICCS 2006*, ser. LNCS, vol. 3993. Springer, 2006, pp. 944–947.

[29] A. Caracas and J. Altmann, “A pricing information service for grid computing,” in *MGC ’07: Proceedings of the 5th international workshop on Middleware for grid computing*. New York: ACM, 2007, pp. 1–6.

[30] R. Buyya, D. Abramson, and J. Giddy, “A case for economy grid architecture for service oriented grid computing,” in *IPDPS ’01: Proceedings of the 10th Heterogeneous Computing Workshop*. Washington, DC, USA: IEEE Computer Society, 2001.

[31] E. Afgan and P. Bangalore, “Computation cost in grid computing environments,” in *ESC ’07: Proceedings of the First International Workshop on The Economics of Software and Computation*. Washington, DC, USA: IEEE Computer Society, 2007, p. 9.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Multi-level Parallelization with Parallel Computational Services in BeesyCluster

Pawel Czarnul
Faculty of Electronics, Telecommunications and Informatics
Gdansk University of Technology
Poland
Email: pczarnul@eti.pg.gda.pl
http://fox.eti.pg.gda.pl/∼pczarnul

***Abstract*—The paper presents a concept, implementation and real examples of dynamic parallelization of computations using services derived from MPI applications deployed in the BeesyCluster environment. The load balancing algorithm invokes distributed services to solve subproblems of the original problem. Services may be installed on various clusters or servers by their providers and made available through the BeesyCluster middleware. It is possible to search for services and select them dynamically during parallelization to match the desired function the service should perform with descriptions of services. Dynamic discovery of services is useful when providers publish new services. Costs of services may be incorporated into the selection decision. A real example of integration of a given function using distributed services has been implemented, run on several different clusters without or with external load and optimized to hide communication latency.**

I. Introduction

PARALLEL computing has come a long way from dedicated parallel applications to grid computing allowing to couple clusters to perform distributed computations. Traditionally, parallel computing has been performed either on shared memory machines or within dedicated multi-node clusters. For the former, threads or processes may be spawned and communicate using shared memory which is supported using e.g. Pthreads [1], Java threads [2], OpenMP [3]. For clusters, MPI [4] and PVM [5] allow development and running parallel programs using the message passing paradigm for communication. For fast communication between processes on one node MPI can use shared memory. MPI implementations may offer various support for threads. Furthermore, computational codes may be wrapped into services, discovered and integrated for distributed computations at a higher level.

Section II investigates existing approaches for parallelization among clusters and proposes a framework for multi-level parallelization through dynamic discovery of services in BeesyCluster that can make use of specific features of the latter. Section III presents a concept and an algorithm for parallelization of computations using BeesyCluster services while Section IV presents experimental results including the ability of the solution to hide latency of Web Service calls. Finally, Section V presents a summary and future work.

II. Related Work and Motivations

Grid systems like CrossGrid [6] or Clusterix [7] use grid middleware such as Globus Toolkit [8] to allow the user to manage, access and share various resources. Grid middlewares allow controlled resource sharing and offer uniform interfaces to access various resources in various administrative domains. In turn, parallelization among clusters has become easier especially for the master-slave or divide-and-conquer paradigms as previously used within clusters [9], [10]. Such parallelization is possible either by running grid-enabled MPI applications or composing distributed services.

Grid-enabled versions of MPI such as MPICH-G2 [11], PACX-MPI [12], BC-MPI [13] allow one application to run on more than one cluster using TCP for inter-cluster and usually faster Infiniband or Myrinet within each cluster. MPICH-G2 uses Globus for job control as can PACX-MPI and LAM/MPI [14] to couple remote clusters. BC-MPI can use BeesyCluster [15] for this purpose [13].

There exist several frameworks that allow management of distributed processing and computations. The MW toolkit [16] allows development of a master-worker style application that works in the distributed environment of Condor. [17] studies load distribution methods for a master-slave scheme on a multi-cluster architecture. Tuning of master-slave applications on grids is analyzed in [18]. Nimrod/G [19] allows resource management and scheduling in a global computational grid. It introduces a concept of computational economy that allows the user to specify the cost and the deadline for the work or negotiate for resources to do the job.

Grid or Web Services can be used as a front-end to HPC resources [20]. This allows parallelization using services. [21] and [22] contain an interesting comparison of performance of a computationally intensive application run on MPI and using Web Services as workers rather than front-ends to HPC resources. The latter showed higher execution times but has advantages in case of firewalls and better interoperability in heterogeneous environments. A methodology to migrate from MPI to Web Service base applications is shown in [22]. [23] implements an Application Management System into an MPI application and its ability to balance load in a computational grid is demonstrated for an application which partitions initial data for parallel execution on a three site grid. [24] demonstrates an approach to find and use grid services for parallel solving of matrix multiplication – although sequential in the workers. The framework allows to specify priority QoS parameters like accuracy, speed, bandwidth.

In this paper, using BeesyCluster and combining features of the previous works, the author proposes a dynamic master-slave parallelization approach able to discover and invoke computational services spawning MPI applications on HPC resources. The algorithm finds and invokes ready-to-use possibly geographically distributed computational services derived from MPI applications published by possibly independent providers from various clusters/servers incorporating possibly changing costs of such services. The solution focuses on efficient parallelization of computations among the services with hiding communication latency and parallelization on clusters using MPI. Although the master-slave paradigm has been widely used in parallel programming and distributed processing on grids as discussed above, the contribution of this work is that the proposed scheme benefits from particular features offered by the BeesyCluster middleware:

- parallel and sequential applications created by users can be published instantly as services within BeesyCluster and made available as Web Services accessible from outside BeesyCluster. The publisher can grant rights to other BeesyCluster users to run the service and defines QoS parameters such as cost the client needs to pay to invoke the service. When a BeesyCluster client invokes the service, the cost is subtracted from their account. Our latest work [25] shows how thousands of existing Linux applications distributed in packages such as deb or rpm can be published in BeesyCluster and thus be available to be used in the proposed load balancing scheme.
- searching for services that might then be used in the load balancing scheme. Currently, each service deployed in BeesyCluster is described in the database as well as in a UDDI registry based on JUDDI and used by BeesyCluster. The latter may be queried using the UDDI API. We have also implemented [25] an intelligent search mechanism based on the similarity of the service description and user query that uses Apache Lucene. For instance, a user might request services for numerical integration and those with best matching descriptions are returned.
- incorporation of possibly changing costs when discovering and selecting services.
- access to the services is restricted only to BeesyCluster users to whom it has been granted. Also, services offered by *various* providers can be integrated in one load balancing scheme.
- in case of parallel applications run on clusters with queuing systems such as PBS, LSF, LoadLeveler etc., BeesyCluster can handle queuing transparently to the client.

BeesyCluster (Figure 1) is a middleware and front-end to several clusters and allows users to access their system accounts on clusters or servers using a web browser or web services [26]. BeesyCluster features an easy-to-use WWW interface with a file manager with drag and drop in the browser, ability to launch graphical or text sessions with remote clusters through a browser. Among others, users can publish own MPI applications from their system accounts as services (Figure 2) and assign a price for invoking the service and access rights to particular users or groups. The service can then be invoked on the cluster the provider has installed it through BeesyCluster via a web browser (Figure 3) or Web Services [26]. BeesyCluster uses SSH to access clusters (Figure 1). Moreover, such services can be combined into workflows [15]. Since the standard MPI and PVM allow parallelization within a cluster, an MPI or PVM application may be published as a service (also studied in [27] instantly with a few clicks (Figure 2, [26]) and used as a part of a framework for balancing work between clusters by passing distinct data sets to particular services. BeesyCluster co-developed by the author hides the underlying queuing (PBS, LSF etc.) system on the cluster the application was installed on.

III. Concept and Algorithm for Parallelization of Computations using BeesyCluster Services

For composing services, usually a workflow graph $G(V,E)$ is defined [28] where V is a set of vertexes denoting tasks while E is a set of edges denoting dependencies between the tasks. Out of a set of services for each task one is to be chosen so that certain QoS goals are optimized. As an example, this could be minimization of the workflow execution time with an upper bound on the total cost of selected services.

In this paper, we focus on multi-level parallelization of computations in a dynamic master-slave fashion using services published by independent providers from potentially geographically distributed clusters. Each service is a parallel MPI application.

The parallelization framework is implemented on the client side and thus does not depend on grid middleware and allows access to several BeesyCluster servers handling local computer centers in different administrative domains. The main steps of the strategy are shown in Figure 4:

1) Providers publish parallel MPI applications as BeesyCluster services (Figure 2). In case the cluster on which the application runs uses a queuing system, its usage is handled by BeesyCluster and hidden to the user. Providers specify costs which will need to be paid by the client when invoking particular services. Each BeesyCluster user has a virtual purse, can earn on own services used by others. A description of each BeesyCluster service is stored in BeesyCluster's own UDDI registry based on JUDDI. Thus the balancing thread can query for services using SOAP.
2) The client queries BeesyCluster for services capable of performing the given function [25] also taking into account the costs of available services. The client receives a list of capable services. The recently developed

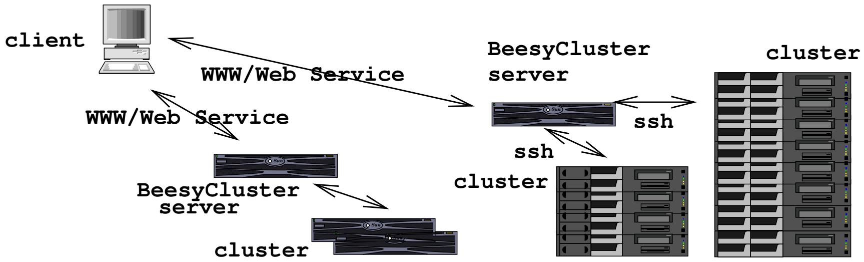
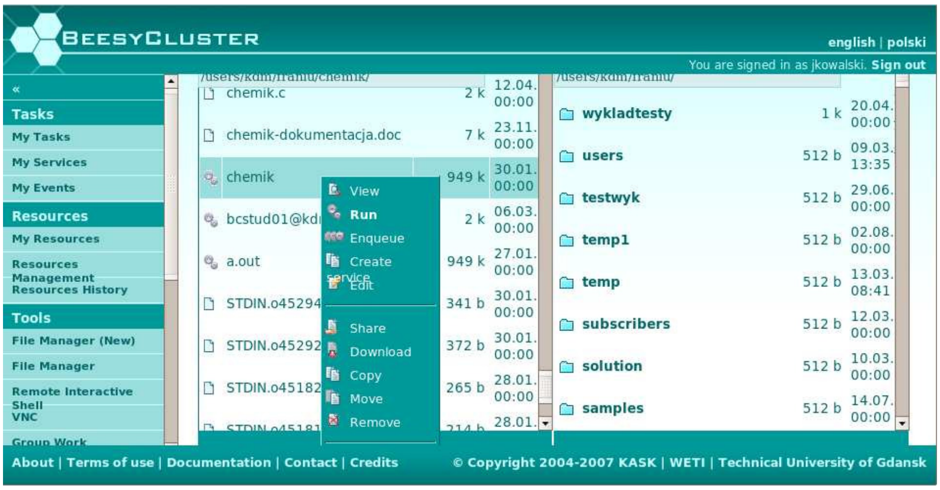


Figure 1: Architecture of BeesyCluster



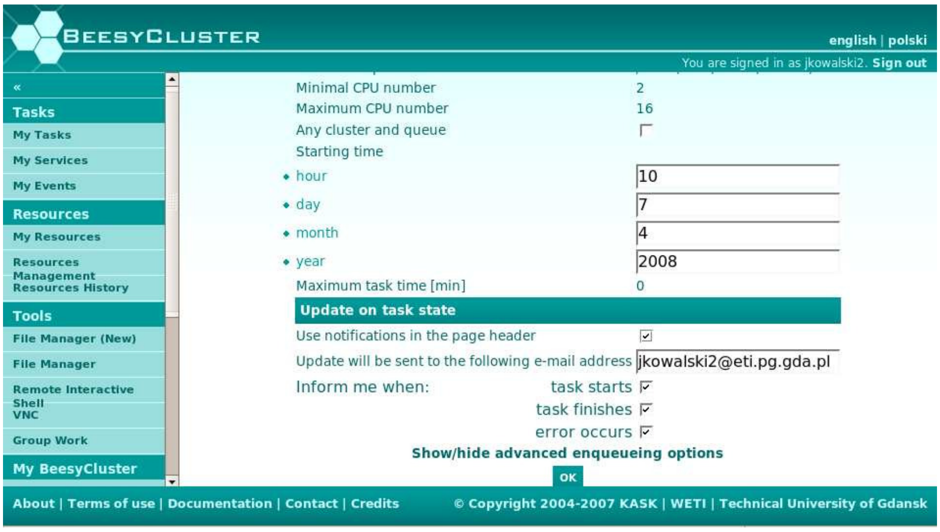
/users/kdm/franiu/chemik/

chemik.c	2 k	12.04. 00:00
chemik-dokumentacja.doc	7 k	23.11. 00:00
chemik	949 k	30.01. 00:00
bcstud01@kd	2 k	06.03. 00:00
a.out	949 k	27.01. 00:00
STDIN.o45294	341 b	30.01. 00:00
STDIN.o45292	372 b	30.01. 00:00
STDIN.o45182	265 b	28.01. 00:00
STDIN.o45181	214 b	28.01.

/users/kdm/franiu/

wykladtesty	1 k	20.04. 00:00
users	512 b	09.03. 13:35
testwyk	512 b	29.06. 00:00
temp1	512 b	02.08. 00:00
temp	512 b	13.03. 08:41
subscribers	512 b	12.03. 00:00
solution	512 b	10.03. 00:00
samples	512 b	14.07. 00:00

Figure 2: Publishing a Parallel MPI Application as a Service in BeesyCluster



Minimal CPU number 2
Maximum CPU number 16
Any cluster and queue ☐
Starting time
- hour 10
- day 7
- month 4
- year 2008

Maximum task time [min] 0

Update on task state

Use notifications in the page header ☑
Update will be sent to the following e-mail address jkowalski2@eti.pg.gda.pl
Inform me when: task starts ☑
task finishes ☑
error occurs ☑

Figure 3: Running a Service Published by Another User and Derived from an MPI Application (WWW)

module in BeesyCluster allows for advanced search and matching of service descriptions to the user textual query [25] using the Apache Lucene engine. For the desired function expressed in the natural language (e.g. numerical integration etc.), BeesyCluster returns a list of services with numerical matching scores. Furthermore, the aforementioned balancing algorithm can choose services with greatest $clusterspeed/(1+cost)$ ratios.

3) The balancing thread (Listing 1) divides the initial data into a predefined number of parts set by the programmer.

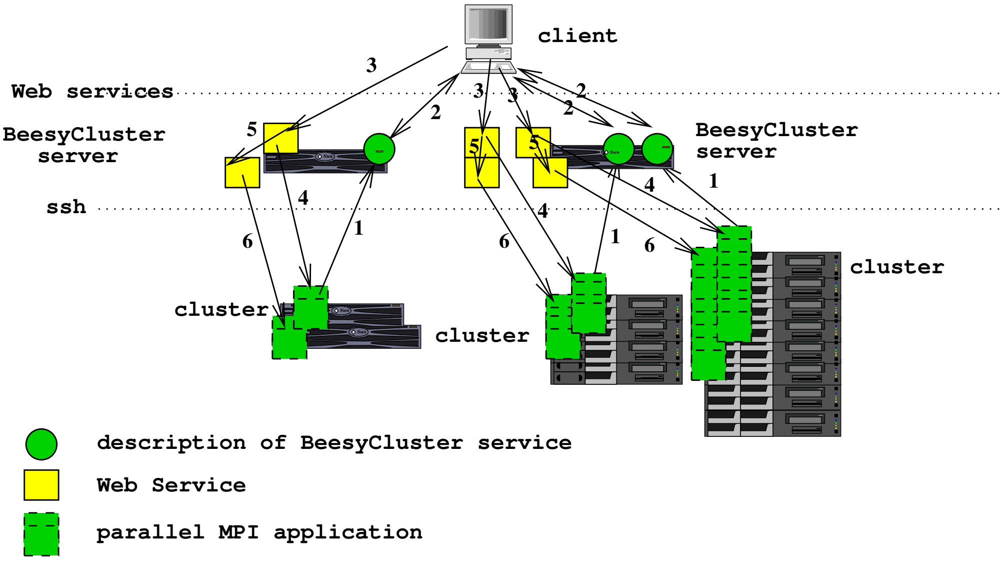


Figure 4: Interactions in the Solution

The thread also checks the speed of each cluster or server on which the services are installed. The balancing thread launches a new thread (Listing 2) for each part which then calls a Web Service in BeesyCluster.

4) The Web Service implementation within BeesyCluster starts an MPI application on a cluster remotely using ssh.
5) The framework launches a second instance of the Web Service for each service which is delayed compared to the first one to hide the communication latency upon termination of the first one.
6) As for the first one, the Web Service in BeesyCluster invokes the MPI application.
7) Then after the first application has terminated, another one is launched (Listing 1) while the one launched as second is still running. This way the cluster is kept busy. The framework proceeds until all data has been processed.

In the integration example considered in this paper the data part corresponds to a range to be integrated.

The algorithm can handle irregular problems taking different amounts of time on data sets of the same size. Examples of such applications include adaptive quadrature integration or $\alpha\beta$ searches [9].

IV. Experimental Results

A. Testbed Application and Environment

As an example, a parallel program for integration of a given function on the given range was used. Initially, the balancing thread divides the range into a predefined number of subranges. Then it launches services on clusters each of which represents a parallel MPI application and passes a single range $[a, b]$ and $accuracy$ which denotes width of polygons used to compute the integral.

The testbed environment used one BeesyCluster server installed at Faculty of Electronics, Telecommunications and Informatics which accesses in particular two clusters:

1) `holk` – a cluster with 288 dual core Itanium 2 processors connected with Infiniband, 2304 GB of RAM, 5.8 TB of disk space,
2) `KASK's cluster n01.eti.pg.gda.pl` – a cluster with 10 Dell PowerEdge 1850 nodes each with 2 dual core Intel Xeon processors, Infiniband interconnect, 40 GB of RAM.

For the tests, the client-BeesyCluster connection was at 2Mb/s while the BeesyCluster-cluster(s) connections were internal university links at 10Mb/s. Two nodes of `holk` and 9 nodes of `KASK`'s cluster were used as 11 virtual clusters on which the client called MPI integration applications as Web Services in BeesyCluster.

B. Hiding latency of the Web Service Calls

The goal of this test was to assess how running two parallel MPI simulations concurrently on one cluster, spawning one with a slight delay can mitigate communication latency. If only one application is run on a cluster (MPI-SA case), after it has terminated, the cluster is idle until a new one is started. If two are run, the second started slightly after the first one (MPI-2A case) the former can keep processors busy after the latter has terminated and a new application is being launched on this cluster. It happens at the cost of context switching between processes. Two nodes (`h001` and `h002`, each node with 2 dual core Itanium2 processors) of cluster `holk` were used. Both MPI-SA and MPI-2A ran with 4 processes per node. The range [1,1000000] with $accuracy$=3e-5 was used. Table I presents execution times for the aforementioned configurations. The variable is the number of ranges to integrate.

```
Vector pending; // with pending WorkRequestResult (cluster + data)
Vector finished; // with finished WorkRequestResult (cluster + data + result)

public void BalancingLoop() {
   // find/discover a certain number of capable services in BeesyCluster based on the
   // required functionality, costs and available budget

   // generate data
   int data_count=COUNT;
   type[] data;
   generate_data_and_store_in_data();

   // prepare data and start processing
   // there are two threads launching each service twice
   for(k=0;k<2;k++)
     for(i=0,j=0;j<services.length;j++) {
       WorkRequestResult wrr=new WorkRequestResult(j,data);
       synchronized(pending) { pending.add(wrr); } // add a request to a queue
       new Thread(this).start(); i++; // spawn a thread to get this data to process
     }
   WorkRequestResult wrr; int finishedcount=0; result=0;

   for(;(i<data_count) || (finishedcount<data_count);i++) { // main loop
     try {
         synchronized(finished) {
           if (finished.size()==0) finished.wait(); // notified by
         } // a notify() from method run()
     } catch (Exception e) { ... }

     // read data from the finished queue and then spawn another thread
     synchronized(finished) {
         wrr=(WorkRequestResult)finished.elementAt(0);
         finished.removeElementAt(0); }
     result=result.add(wrr.result); finishedcount++; // 'add' the result depending on
     // the problem
     if (i<data_count) {
         // now launch another thread for running the same service unless conditions have changed
         // - in the latter case discover next best service
         wrr.data=new data;
         synchronized(pending) { pending.add(wrr); } // add to the pending tasks
         new Thread(this).start(); } // spawn a thread to get this data to process
   } System.out.println("Result="+result);
}
```

Listing 1: Main Load Balancing Loop

```
public void run() {
  WorkRequestResult wrr;
  synchronized(pending) {     // first get a pending request
    wrr=(WorkRequestResult)pending.elementAt(0);
    pending.removeElementAt(0); }
  // process it on the cluster and update the result
  wrr.result=CallService(wrr.service,wrr.data);
  synchronized(finished) { // insert the result
    finished.add(wrr); } //  into the finished queue
  // now wake up the main thread
  synchronized(finished) { finished.notify(); }
}
```

Listing 2: Thread Calling Services

Table I: Hiding Latency of Web Service Calls: Execution Time [s]

ranges	2	4	8	20	30	50
MPI-SA	64	68	76	86	117	156
MPI-2A		64	70	82	85	111

It is clear that launching two MPI applications (with 4 processes) each on a half of the initial range gives best possible results as there are just two calls (one to each cluster) and returning results. However, we obtain the same results for 4 ranges (2 calls to each cluster) run with concurrent applications on each node (MPI-2A). Here, the communication latency is practically hidden by the other applications running on the clusters. Moreover, increasing the number of ranges gives MPI-2A an edge over MPI-SA.

The larger the number of ranges the more perfect balance can be achieved especially if we consider potentially various speeds of clusters or additional load which may appear on some clusters.

For the following tests, the best execution times including hiding communication latency using this technique are considered.

C. Execution Time under External Load

Table II: Execution Time under External Load [s], 11 clusters

ranges	18	36	72	144	288
execution time [s]	174	150	147	167	218

In this case, all 11 clusters were used while node compute-0-0 of KASK's cluster was loaded with computationally intensive processes launched by another user. Table II shows how the execution time of the integration application changes with the number of ranges. A small number of ranges means that the integration processes (on reasonably large subranges) on compute-0-0 were considerably prolonged. On the other hand, a very large number of ranges resulted in a large overhead. 72 ranges turned out to be the best trade-off between the range counts tested. The range [1,1000000] with *accuracy*=3e-5 was used.

D. Scalability Results

For the first test, a single homogeneous cluster (KASK's cluster was partitioned into up to 9 virtual clusters each composed of 4 processing cores. Execution times and speed-ups of the load balancing solution were measured and shown in Figures 5 and 6 given various numbers of ranges from the number of nodes up to 16 times the number of nodes used for the run. The range [1,1000000] with *accuracy*=3e-5 was used.

Similarly, the second test involved a heterogeneous environment with two nodes of cluster holk as 2 virtual clusters and up to 9 virtual clusters defined on the KASK cluster as in the previous experiment. Each processor of the first cluster is approximately 7 times faster than of the KASK cluster for the integration example. The balancing algorithm merges ranges for the faster cluster trying to obtain similar execution times for each requested range for each cluster. Figures 7 and 8 present the execution time and speed-up for integration of ranges [0,2e6], [0,5e6], [0,1e7] with *accuracy*=3e-5. All presented examples confirm good speed-ups (compared to a single fastest processor in the configuration) given the overhead of the client-BeesyCluster Web Service calls, BeesyCluster-cluster SSH communication and management of ranges.

V. Summary and Future Work

The paper presented an idea and an implementation of a framework for parallelization of computations in a dynamic master-slave fashion using distributed services installed in BeesyCluster and derived from MPI applications. The framework can use services published by various users and discover services at runtime for a desired application using service parameters such as cost. The algorithm is able to use the services to solve the problem on subsets of the initial data set and then merge into final results. An example of integration of a given function was presented and run on several modern clusters using services installed on them. Tests with and without external load for up to 11 clusters with varying speeds confirmed good scalability of the approach.

Future work includes testing more applications in the proposed scheme as well as tests of searching of services using ontologies.

Acknowledgments

Calculations were carried out at the Academic Computer Center in Gdansk, Poland. The work is sponsored by research grant MNiSW N516 383534.

References

[1] B. Wilkinson and M. Allen, *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers.* Prentice Hall, 1999.

[2] R. Buyya, Ed., *High Performance Cluster Computing, Programming and Applications.* Prentice Hall, 1999.

[3] "OpenMP: Simple, Portable, Scalable SMP Programming," http://www.openmp.org/.

[4] W. Gropp and E. Lusk, *User's Guide for mpich,a Portable Implementation of MPIVersion 1.2.2*, 2001, http://www-unix.mcs.anl.gov/mpi/mpich/.

[5] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Mancheck, and V. Sunderam, *PVM Parallel Virtual Machine. A Users Guide and Tutorial for Networked Parallel Computing.* MIT Press, Cambridge, 1994, http://www.epm.ornl.gov/pvm/.

[6] Official Crossgrid Information Portal, http://www.crossgrid.org/main.html, supported by Grant No. IST-2001-32243 of the European Commission.

[7] CLUSTERIX, *The National Linux Cluster*, http://clusterix.pcz.pl.

[8] B. Sotomayor, "The Globus Toolkit 4 Programmer's Tutorial," November 2005, http://www.casa-sotomayor.net/gt4-tutorial/.

[9] P. Czarnul, "Programming, Tuning and Automatic Parallelization of Irregular Divide-and-Conquer Applications in DAMPVM/DAC," *International Journal of High Performance Computing Applications*, vol. 17, no. 1, pp. 77–93, Spring 2003.

[10] R.D.Blumofe, C.F.Joerg, B.C.Kuszmaul, C.E.Leiserson, K.H.Randall, and Y.Zhou, "Cilk: An efficient multithreaded runtime system," in *Proceedings of the 5th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, July 1995, pp. 207–216.

[11] N. Karonis, B. Toonen, and I. Foster, "MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface," *Journal of Parallel and Distributed Computing (JPDC)*, vol. 63, no. 5, pp. 551–563, May 2003.

[12] R. Keller and M. M?ller, "The Grid-Computing library PACX-MPI: Extending MPI for Computational Grids," www.hlrs.de/organization/amt/projects/pacx-mpi/.

[13] P. Czarnul, "BC-MPI: Running an MPI Application on Multiple Clusters with BeesyCluster Connectivity," in *Proc. of PPAM 2007*, Springer-Verlag, Ed., vol. LNCS 4967, Poland, 2007.

[14] LAM/MPI Parallel Computing, http://www.lam-mpi.org/.

[15] P. Czarnul, "Integration of Compute-Intensive Tasks into Scientific Workflows in BeesyCluster," in *Proceedings of ICCS 2006 Conference,*. University of Reading, UK: Springer Verlag, May 2006, lecture Notes in Computer Science, LNCS 3993.

[16] E. Heymann, M. A. Senar, E. Luque, and M. Livny, "Adaptive scheduling for master-worker applications on the computational grid," in *GRID '00: Proceedings of the First IEEE/ACM International Workshop on Grid Computing.* London, UK: Springer-Verlag, 2000, pp. 214–227.

[17] A. E. D. Giusti, M. R. Naiouf, L. C. D. Giusti, and F. Chichizola, "Dynamic load balancing in parallel processing on non-homogeneous clusters," *Journal of Computer Science & Technology*, vol. 5, no. 4, December 2005.

[18] G. F. de Carvalho Costa, "Automatic dynamic tuning of parallel/distributed applications on computational grids," Ph.D. dissertation, Universitat Autonoma de Barcelona, May 2009.

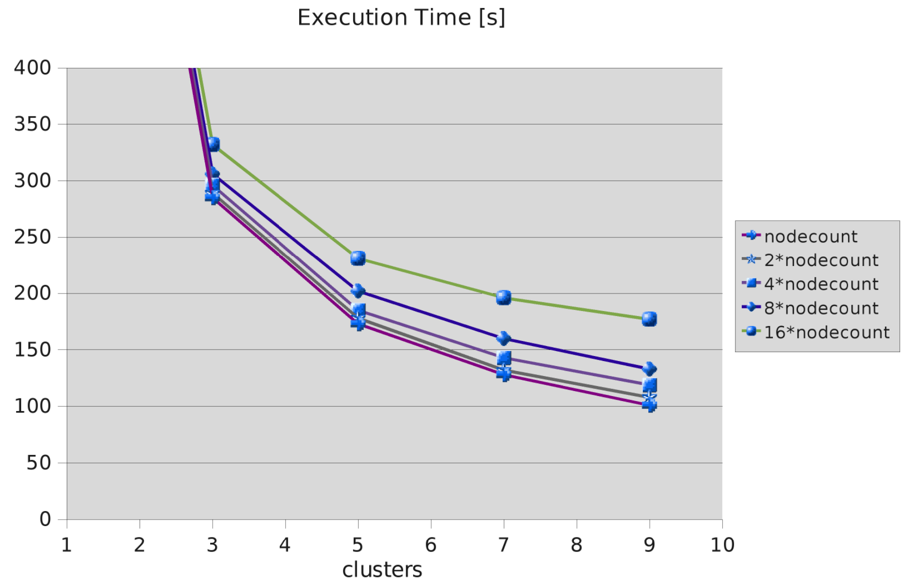


Figure 5: Execution Time in a Homogeneous Environment

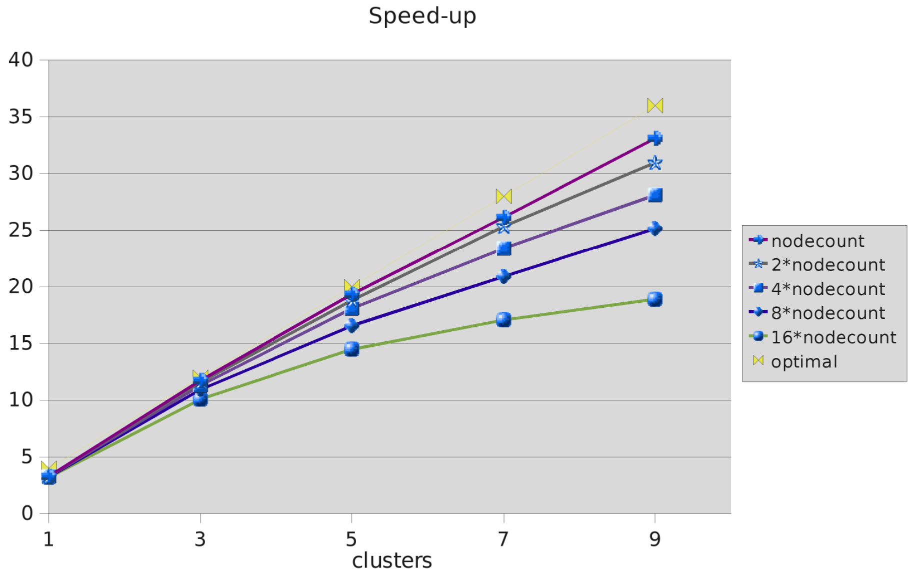


Figure 6: Speed-up in a Homogeneous Environment

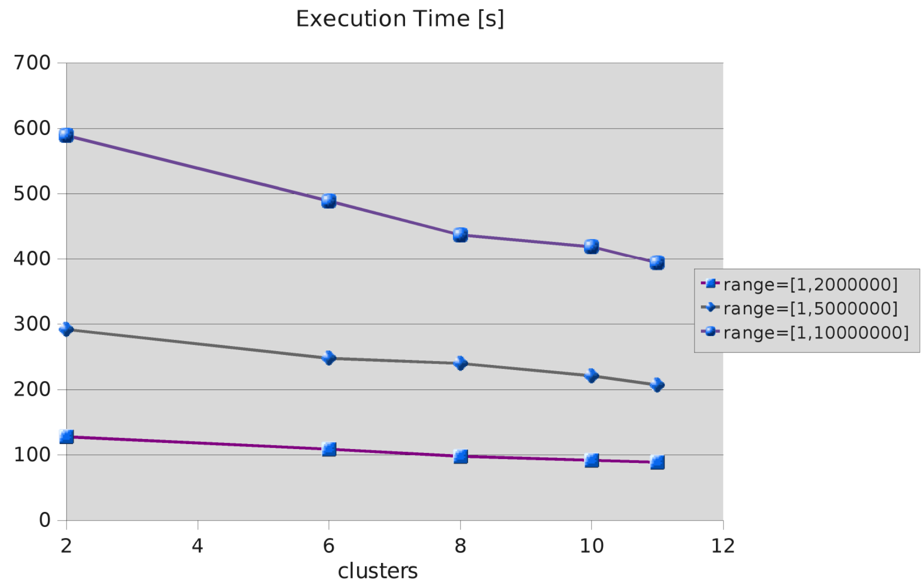


Figure 7: Execution Time in a Heterogeneous Environment

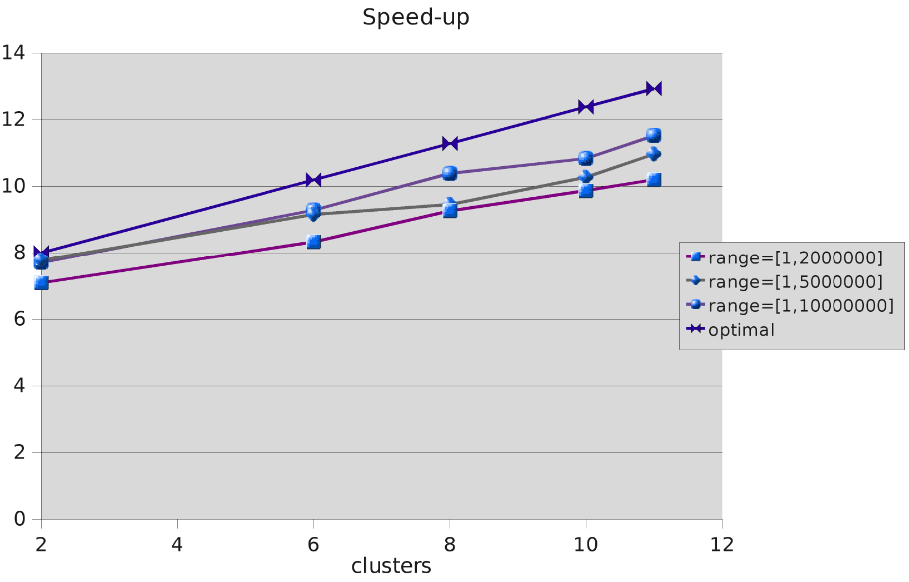


Figure 8: Speed-up in a Heterogeneous Environment

[19] R. Buyya, D. Abramson, and J. Giddy, "Nimrod/g: An architecture for a resource management and scheduling system in a global computational grid," in *Proceedings of the 4th International Conference on High Performance Computing in Asia-Pacific Region (HPC ASIA 2000)*.IEEE Computer Society Press, 2000, pp. 283–289.

[20] L. Dimitriou, "Distributed parallel computing with web services. a pivotal role on the back end," SOA World Magazine, vol. 5, issue 2, February 2005, `http://soa.sys-con.com/read/48036.htm`.

[21] D. Puppin, N. Tonellotto, and D. Laforenza, "Using web services to run distributed numerical applications," in *PVM/MPI*, ser. Lecture Notes in Computer Science, D. Kranzlmüller, P. Kacsuk, and J. Dongarra, Eds., vol. 3241. Springer, 2004, pp. 207–214.

[22] D. Puppin, N. Tonellotto, and D. Laforenza, "How to run scientific applications over web services," in *ICPP Workshops*. IEEE Computer Society, 2005, pp. 29–33.

[23] C. Boeres, A. P. Nascimento, V. E. F. Rebello, and A. C. Sena, "Efficient hierarchical self-scheduling for mpi applications executing in computational grids," in *MGC '05: Proceedings of the 3rd international workshop on Middleware for grid computing*. New York, NY, USA: ACM, 2005, pp. 1–6.

[24] M. Wurz and H. Schuldt, "Dynamic parallelization of grid-enabled web services." in *EGC*, ser. Lecture Notes in Computer Science, P. M. A. Sloot, A. G. Hoekstra, T. Priol, A. Reinefeld, and M. Bubak, Eds., vol. 3470. Springer, 2005, pp. 173–183.

[25] P. Czarnul and J. Kurylowicz, "Automatic conversion of legacy applications into services in beesycluster," in *Proceedings of 2nd International IEEE Conference on Information Technology ICIT'2010*, Gdansk, Poland, June 2010, in press.

[26] P. Czarnul, M. Bajor, M. Fraczak, A. Banaszczyk, M. Fiszer, and K. Ramczykowska, "Remote Task Submission and Publishing in BeesyCluster : Security and Efficiency of Web Service Interface," in *Proc. of PPAM 2005*, Springer-Verlag, Ed., vol. LNCS 3911, Poland, Sept. 2005.

[27] E. Floros and Y. Cotronis, "Exposing mpi applications as grid services," in *Euro-Par*, ser. Lecture Notes in Computer Science, M. Danelutto, M. Vanneschi, and D. Laforenza, Eds., vol. 3149. Springer, 2004, pp. 436–443.

[28] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Sheng, "Quality driven web services composition," in *Proceedings of WWW 2003*, Budapest, Hungary, May 2003.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Managing Large Datasets with iRODS—a Performance Analysis

Denis Hünich, Ralph Müller-Pfefferkorn
Center for Information Services and High Performance Computing (ZIH)
Technische Universität, Dresden

***Abstract*—The integrated Rule Orientated Data System (iRODS)[3] is a Grid data management system that organizes distributed data and their metadata. A Rule Engine allows a flexible definition of data storage, data access and data processing. This paper presents scenarios implemented in a benchmark tool to measure the performance of an iRODS environment as well as results of measurements with large datasets. The scenarios concentrate on data transfers, metadata transfers and stress tests. The user has the possibility to influence the scenarios to adapt them to his own use case. The results show the possibility to find bottlenecks and potential to optimize the settings of an iRODS environment.**

I. Introduction

TODAY and even more in future scientific research generates and will generate enormous amounts of data. To handle these data they are often stored in distributed locations within a Grid data management systems like the integrated Rule Orientated Data System (iRODS)[3]. iRODS allows a fast and easy access on files and their provenance. As it is used more and more in production environments and with rapidly increasing data sets it is of importance to be able to analyze and optimize the performance of an iRODS installation.

This paper describes a tool and scenarios which enable the measurement of the performance of data and metadata transfers as well as internal parameters of an iRODS system. First measurement with millions of files stored in iRODS show already the potential to optimize the performance. It is based on the benchmark tool BenchIT[1]. The aim of the development was to allow the users to easily optimize an iRODS environment. This paper is structured as follows: section 2 presents related work, section 3 gives an overview of iRODS and BenchIT, section 4 describes the investigated scenarios, section 5 shows the results of measurements and section 6 concludes this paper.

II. Related Work

There exist a variety of tools that define benchmarks and provide performance tests of I/O, e.g. IOZone [5] or IOR [4]. But these focus primarily on POSIX files systems. iRODS is not a POSIX accessible system but provides its own clients to access the storage.

iRODS is a relatively new product (version 1 released in 2008) and there are only a few performance evaluation results published, most of them focus on simple tests. Iida[7] measured the performance of transfers between two iRODS systems. Furthermore, he performed scaling tests for iCAT, concurrent tests for iCAT and a comparison of iRODS and SRB. Baquero[8] tested the performance of iRODS and Hadoop[9] and compared the results. On the iRODS website results of an ”Ingestion Test”[10] and a ”General Query Stress Test”[11] are shown. This are stress tests of the iCAT server. But a systematic performance analysis of iRODS was not done yet nor exist scenarios or a tool for users. Thus, the scenarios and the tool presented in this paper gives the user the possibility to test an iRODS environment and compare it with others.

III. Overview

A. iRODS

iRODS, developed by the Data Intensive Cyber Environment (DICE) group, organizes distributed data up to a range of petabytes. It is released with an Open-Source-Licence. iRODS stores data on heterogeneous storage systems (so-called ”iRODS server”) and the related metadata in a database on a special iRODS server named ”iCAT enabled server” (figure 1). Metadata are for example information on stored data, their locations or user defined information. A set of metadata is created and stored automatically by iRODS for every stored data - e.g. file, location or user. Additional information can be stored as user defined metadata.

An iRODS server has a Rule Engine which interprets rules. A Rule contains Micro-Services, arranged in a workflow, which process special tasks like replicating data or calculating check sums. Microservices can be provided by anyone. Additionally, a rule defines conditions to execute the Micro-Services and has a backup strategy if an error occurs. The definition of rules allows the users to configure the iRODS environment according to their requirements and is also one reason why performance testing is so important for iRODS. iRODS provides a number of clients - e.g. command line clients (the iCommands) or the iRODS explorer - as well as programming interfaces to transfer files to and from an iRODS server for example.

B. BenchIT

BenchIT[1] is a benchmark tool to analyze and optimize computer systems. It was developed at the Center for Information Services and High Performance Computing (ZIH) of the Technische Universität Dresden. BenchIT provides a set of small algorithms (kernels) to measure the performance of a system. The execution of a measurement or a set of

978-83-60810-27-9/09/$25.00 © 2010 IEEE

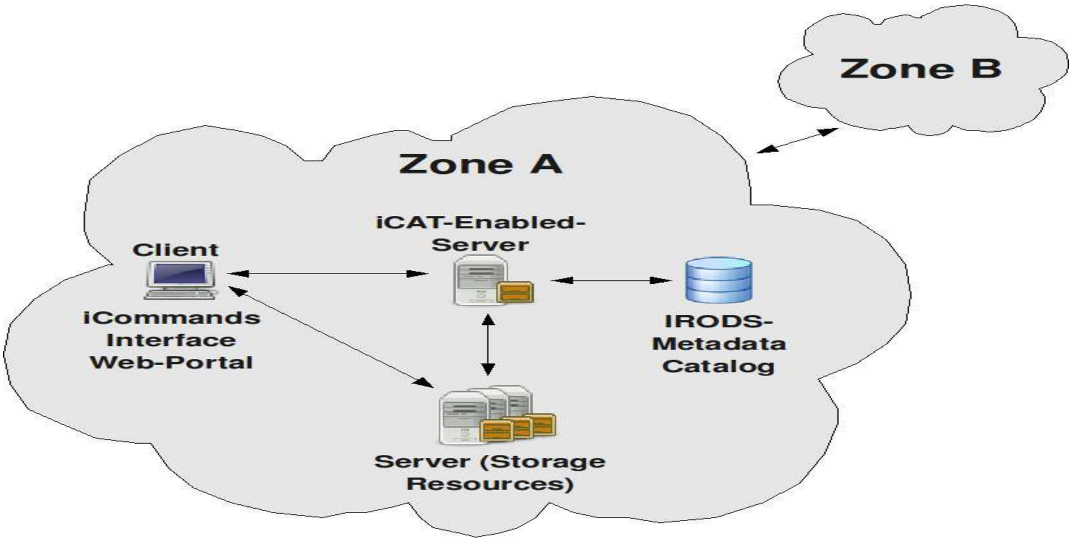


Fig. 1. Schematic representation of an iRODS environment

measurements is driven by a graphical utility providing an interface to the kernels. The interface allows the user to adjust the measurement. Relevant information on the measurement are saved together with the results in a file. The results are evaluated and diagrams of the measurement are generated. Additionally, the result file can be uploaded to a central server via a web interface [2] and can be compared with the measurements of other users.

The scenarios described in this paper were realized as kernels in BenchIT. The advantage of this approach is the use of the existing BenchIT environment with its graphical user interface and the functions to present and analyze the data.

IV. Scenarios

In the following a number of scenarios are defined that describe typical usage profiles when managing large datasets with iRODS. The scenarios are divided into four categories: data transfers with iCommands or Micro-Services, transfer of metadata, stress tests and a scenario to measure the performance of given Micro-Services. They use the client tools of iRODS such as the iCommands iput[1], iget[2], and imeta[3]. To characterize the performance the time for an event in a scenario is measured. Either the time or the derived data transfer rates are used. To avoid outliers in the results, the measurements were repeated several times and the average of the measured time were calculated.

A. Data transfer

1) Different stages of data transfer: The aim of this scenario is the measurement of different stages of a data transfer. A data transfer with iput or iget sets up the transfer environment at first. Then the connection to the iRODS server can be established. After the successful connection the client will be authenticated. If the client has the required rights the transfer starts. Usually, iRODS chooses automatically the number of parallel threads it uses for the transfer if more than one file are to be read or written. This scenario also allows a manual selection.

For the measurement a data transfer is divided in the following stages:

1) Set up of the transfer environment (Environment)
2) Establishing connection (Connection)
3) Authentication and authorization (Login)
4) Data transfer (including metadata writing) (Put File)

For these 4 stages the times are measured. Furthermore, the total time will be determined to set the single stages in relation to it. Thus, the user has the opportunity to see where the main fraction of time is used for the transfer and how it might change e.g. when increasing the number of simultaneously started data transfers. It is also possible to get an idea of how much time is used to write/read the metadata by transferring a one byte sized file because then the runtime for the last stage is mainly used for reading/writing metadata.

2) Parallel transfer of many files: This scenario transfers a defined number of files simultaneously and measures the runtime for the whole transfer process. The number of files transfered at once can be varied. In contrast to the scenario above the user gets no information about a single data transfer. The aim is to find the number of files/parallel transfers for which the iRODS environment works most efficient. The result also depends on the file size and the usable bandwidth.

3) Transfer of directories: The iCommands iput and iget provide the possibility to transfer the content of a whole directory at once. The difference between transferring a directory and transferring a number of single files is that in the first case the iRODS server only has to handle one request for the transfer and he can use all resources for this request. This

[1] iput - transfers files from the client to the iRODS server
[2] iget - transfers files from the iRODS server to the client
[3] imeta - writes metadata to the iCAT server

scenario measures the time for such a transfer with a varying number of files in the directory. The time mainly depends on the bandwidth and the time used to write metadata. The larger the files and the smaller the number of it, the more the runtime depends on the bandwidth. In the opposite case the time is mainly used to write the metadata.

4) Data transfer for varying file sizes: This scenario measures the performance of a transfer for varying file sizes. The workload for the iRODS server is small because only one file will be transfered and the iRODS server can use all resources for it. With small files the user can measure the latency and with large files the bandwidth. Furthermore the user can check the preferences set for parallel data transfers of the iRODS server for different files sizes. Therefore, it is possible to vary the number of threads for a file transfer and to compare the performance results with the results of the automatic settings.

B. Transfer of user defined metadata

To store user defined metadata on the iCAT server iRODS offers two ways. In the parallel case the metadata will be transfered with simultaneously started imeta requests each writing one element. For the performance evaluation the time to transfer all metadata is measured. In the sequential case the metadata are written in a file. The file is transfered to the iRODS server with one request, but then the metadata sets are written one by one sequentially.

The user can vary the number of metadata entries, compare the results and decide which kind of transfer is to prefer. Especially when writing large amounts of metadata these two ways can produce quite different performance results (see section V-D).

C. Stress tests

In the stress tests the iCAT server is analyzed in cases when it is flooded with requests or the number of files to store in iRODS (and thus needs to be managed by the iCAT server) increases dramatically.

1) Requests to the iCAT server: This stress test starts a defined number of parallel imeta requests to the iRODS server. The number of requests is varied to find the number of operations per second the iRODS environment can manage. Additionally, it is possible to get the number of requests the iCAT server refuse. This occurs when the maximum number of requests the database on the iCAT server can handle is exceeded.

2) Management of large metadata sets on the iCAT server: In this test the number of metadata sets the iCAT server has to manage is increased. For that, a directory with a (large) number of files is transfered successively to the iRODS environment. The time for every transfer is measured. With every file arriving in the iRODS environment the iCAT server needs to handle a larger number of metadata. This usually results in an additional time penalty when transferring the next data.

D. Time measurement of Micro-Services

Micro-Services allow the execution of small tasks directly on the iRODS server. They should not put to much workload on the server because this influences its data management capabilities. This scenario was created to measure the runtime of Micro-Services in any user defined rule. It allows developers as well as iRODS administrators to see how the performance of a Micro-Service behaves in different workflows. The scenario introduces three Micro-Services that measure the time before, between or after Micro-Services of a Rule. The results are collected at the iRODS server and moved to the BenchIT environment for analyses.

V. Results

This chapter presents measurement results of the scenarios described above. These were performed to optimize an iRODS installation to be used in production. After some generic measurements and stress tests a real use case from genetics (the future major user of the installation) was simulated.

A. The test system

The measurements were done on an iRODS installation at ZIH. The combined iRODS/iCAT server with 8 Dual Core AMD Opteron 885 (2.6 GHz) CPUs and with a memory of 32 GByte provides the storage resources and the database for the metadata over a 4 GBit/s SAN network. The client ran on a cluster with Intel Xeon Quad Core X5472 (3.00 GHz) CPUs and a 10GBits/s Ethernet network. iRODS version 2.1 was used with a PostgreSQL database for metadata storage.

B. Handling large sets of data

Figures 2 and 3 show the measurements for the single stages (see IV-A1) when 150 files are transfered simultaneously, each of them 1 Byte large. In figure 2 the iRODS environment contained no files (fresh install). Thus, the management overhead of the iCAT server is very small. Most time is used for the connection and for the file transfer including the writing of metadata. The results in figure 3 look quite different. Here, the iRODS environment already contained 13 million files and their metadata. The transfer time increased almost by a factor of 8. The reason is the increased effort to write metadata if the database is already large. This has also an effect on connection and login time. The iRODS server was not able to work off the requests as fast as before. This delay deferred the connection establishing and the login. In average the total time for the transfer increased by a factor of twenty.

C. Comparing the reading and writing of data

This test compares transfers done with iput and iget for three different states of the iRODS environment (empty, 7 million files and 13 million files stored in iRODS). A directory with a varying number of files (each with a size of 1 Byte) was written and read. Figure 4 shows the results. In the empty state reading needed 4 times longer than writing for 150 files. This means that finding the requested metadata and files needed much longer than writing both.

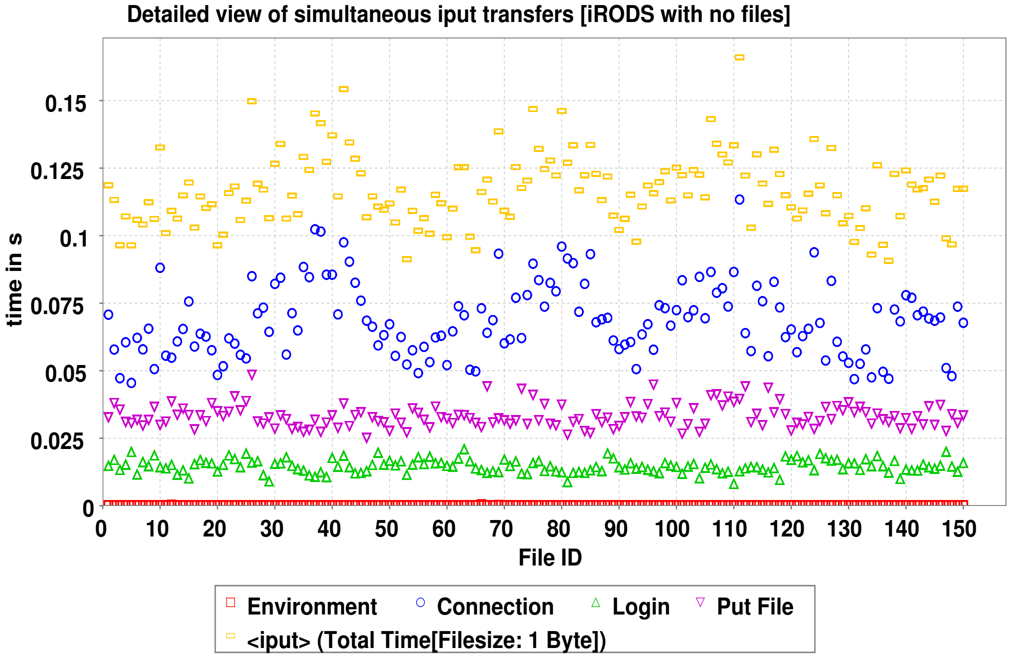


Fig. 2. Simultaneous data transfer of 1 Byte large files with iput. The iRODS environment contains no files. Shown is the time needed for every stage and the overall transfer time (stages, see IV-A1).

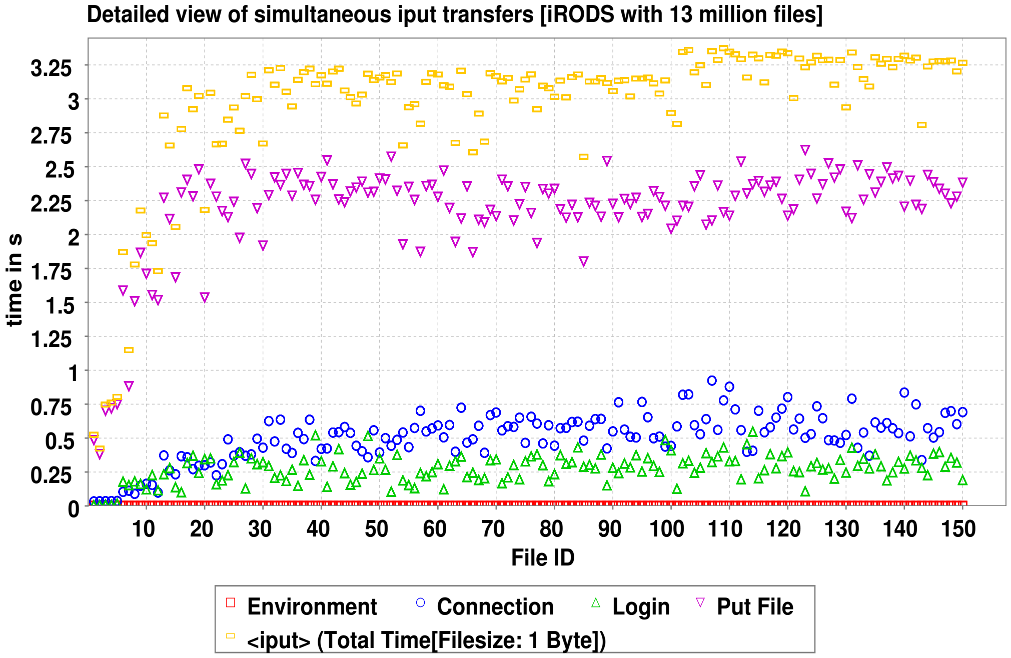


Fig. 3. Simultaneous data transfer of 1 Byte large files with iput. The iRODS environment contains 13 million files. Shown is the time needed for every stage and the overall transfer time (stages, see IV-A1).

In the next state (7 million files) the behavior changes. The time for writing data increased significantly compared to reading, where the increase is small. Nevertheless, the performance of writing was still better than of reading. The time difference between iput and iget decreased by a factor of 1.5 for 150 files written or read.

As expected, in the last case of the filled iRODS environment the performance decreased both for reading and writing. The increase of time iget needed is small compared to iput, which is 7 times slower than in the empty state. Now writing data is slower than reading files.

Summarizing one can say that the results of this test show that writing the standard metadata is an important performance factor and that the number of metadata stored on the iCAT server has more influence on writing than on reading files.

D. Writing user defined metadata

This test wrote the same amount of metadata to iRODS using the two transfer methods described in section IV-B. Furthermore, the measurements were performed with the iRODS environment in two states: empty and filled with 13 million files. Figure 5 shows the results.

In the case of the empty iRODS environment the difference between the performance of simultaneously and sequentially written metadata is small. This changed when the iRODS environment was filled with 13 million files. While the time of simultaneously written metadata increased already significantly (e.g. for a number of 150 metadata elements by a factor of 5), the performance of the sequential method got even worse (e.g. 24 times longer for 150 elements). The reason for the latter is the accumulation of the additional time (originating from the slowdown due to the filled iCAT server) needed for the sequential writing of the metadata (see description in IV-B).

E. Stress tests

On the iCAT server it is possible to configure the maximum number of requests allowed to the database. With scenario IV-C1 it is possible to measure the actual number of requests an iCAT server can handle. Measurements were done for three states - setting the maximum number of simultaneous requests to 200, 700 and 1000, respectively (the default value is 100). Additionally, the shared memory of the database was increased to 2000 MByte - otherwise the database had refused requests. The test varied the number of simultaneously started processes. Each process starts 10 imeta requests. Figure 6 shows the results for the three stages. Up to 100 processes iRODS can handle more than 35 operations per second. Between 100 and 500 processes iRODS finished the requests with about 34 op/sec. This value decreases faster, down to 27 Op/sec.

The second test (see IV-C2) continuously transfers 1000 files at once (each file with a size of 100 Byte) to the iRODS server. With every run the number of metadata to be managed increases. In the empty iRODS environment the transfer time is about 7 seconds and up to 150 files can be written per second (figure 7). With 13 million files already stored on the iRODS server, the time increases to 38 seconds and only 25 files can be transfered per second. The performance decrease is a factor of about 6. The major cause is the metadata handling in the database. The shift of the transfer time between 7 million and 10 million files was probably caused by a process on the iRODS server. Because the overall time of the measurement needed more than one day it was not possible to identify the exact reason for the shift when the measurement was finished.

F. Using the scenarios to simulate a real use case

In the following it is described how the benchmark tool and the scenarios were used to simulate a real use case. In genetics automatic microscopes take a large number of pictures of gene screening processes. For a gene screen ten thousands of pictures needs to be analyzed. The management of the pictures and their metadata is done with iRODS.

To find optimal usage parameters for this use case it was simulated using 10,000 files each with a size of 10 MByte. The transfer was done with one iput request. Figure 8 shows the time of the transfer as function of the different number of threads (parallel transfers) used and for different conditions of the iRODS environment (empty, 7 million files in iRODS and 13 million files in iRODS). The condition of the iRODS environment had more influence on the performance than the used number of threads. For such a large number of files to transfer the performance variation between the thread The difference of the writing time between an empty database and a database filled with metadata for 13 million files is about 20%.

Figure 9 shows how time and bandwidth will change if the files are stored in several directories and these are simultaneously transfered to the iRODS environment. The use of 5 directories instead of one reduces the time by more than the half and when using 10 directories the runtime is about a third. The reason is the better utilization of the bandwidth. That means in this case it is useful to divide the files on more than one directory by using subdirectories. The subdirectories can be transfered simultaneously and reduce the total transfer time to optimize the use of the bandwidth.

VI. Conclusion

In this paper scenarios for iRODS performance measurements were presented, which are integrated in the tool BenchIT. This allows administrators or users of iRODS to measure the performance of an iRODS installation easily and to adopt the usage patterns to an optimal performance. The scenarios provide performance measurements of data transfers, user defined metadata transfers, stress tests of the iCAT server and the runtime of Micro-Services. Furthermore, performance tests done on an existing iRODS environment and simulating a real use case were presented. Among other things the results show that the performance of data transfers decreases when the number of metadata the iCAT server has to manage grows significantly. But the measurement described in section (V-F) also shows that it is possible to optimize the performance by efficiently using the iRODS capabilities.

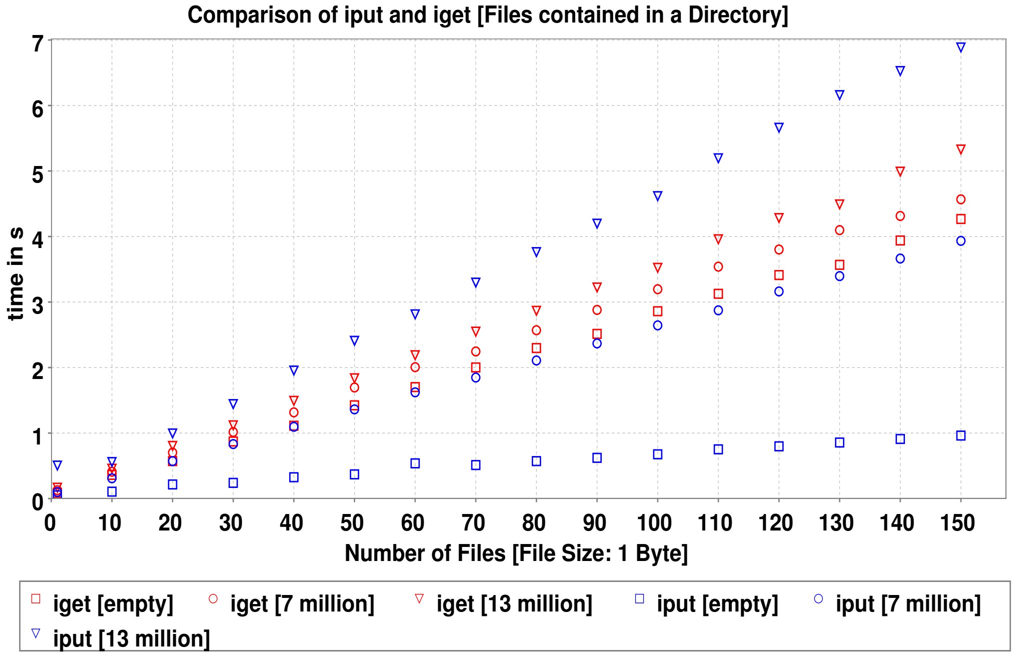


Fig. 4. Comparison of writing (iput) and reading (iget) a varying number of files (each with a size of 1 Byte) transfered at once in a directory. The figure shows the time needed for the transfers with iput and iget for three different states of the iRODS environment (no [empty], 7 million files and 13 million files already stored in iRODS).

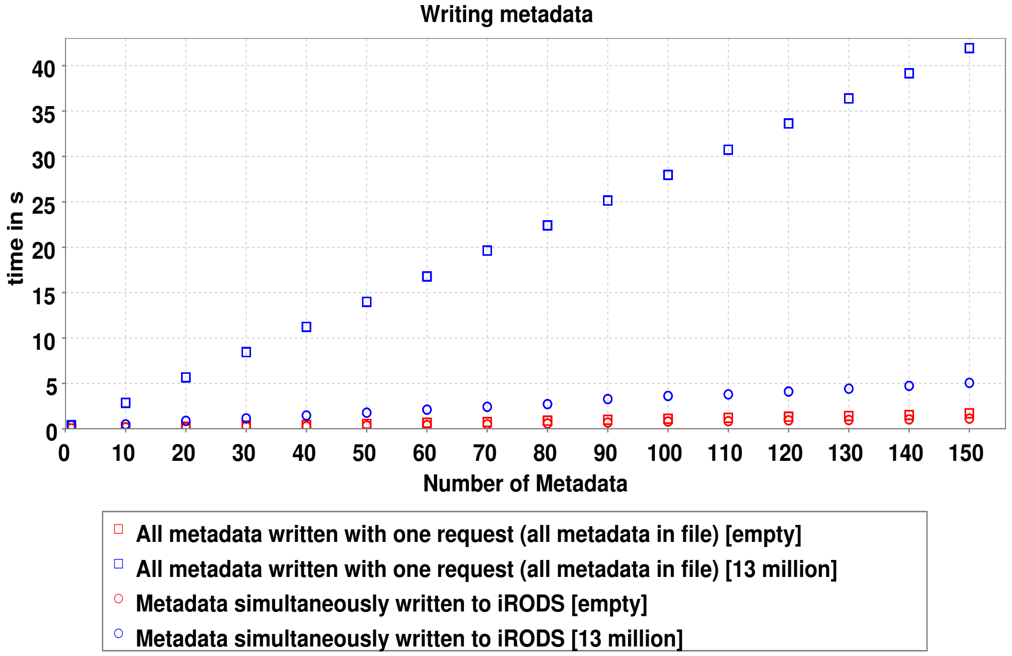


Fig. 5. Comparison of simultaneously (parallel imeta requests) and sequentially (in one file) written metadata. Shown is the time needed for writing a varying number of metadata for two states of the iRODS environment (empty and filled with 13 million files).

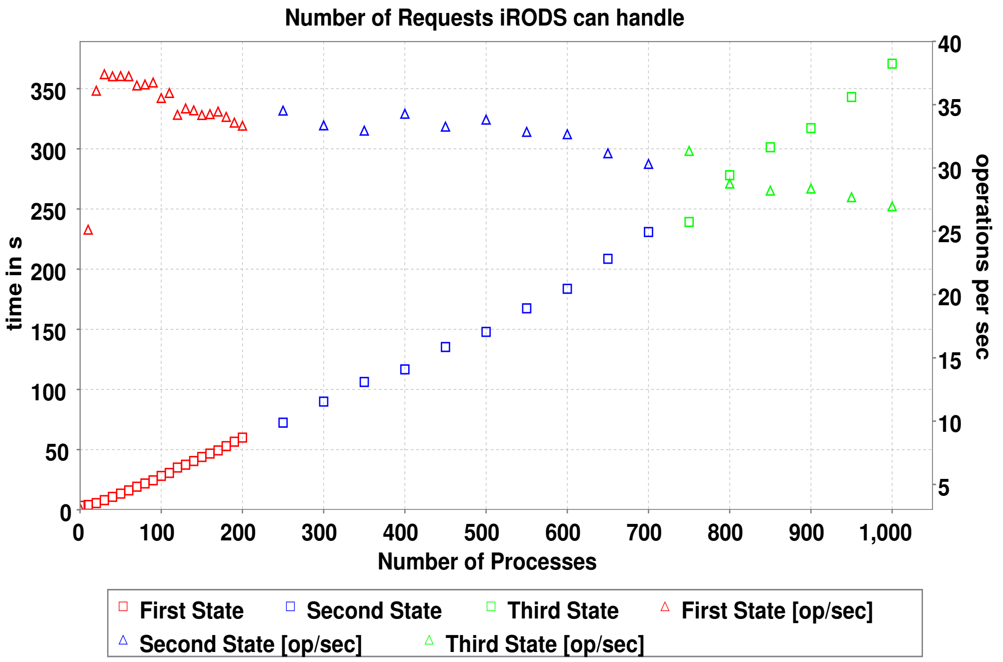


Fig. 6. Number of metadata requests: A varying number of processes were simultaneously started. A process starts sequentially 10 imeta requests. Shown are the time needed to process all requests and the resulting operations per second (op/sec). The three states correspond to the setting of the maximum number of requests allowed on the iCAT server (200, 700 and 1000).

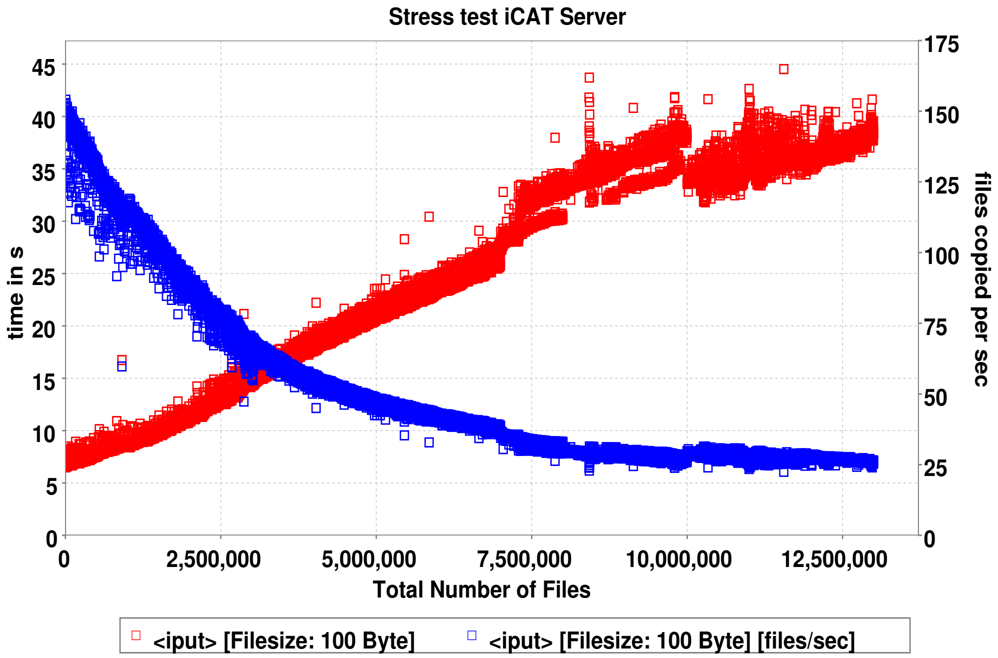


Fig. 7. Stress test: 13 million files (each 100 Byte) are transfered with iput. With each run 1000 files are transfered. Shown are the time for each run and the number of file transfers per second.

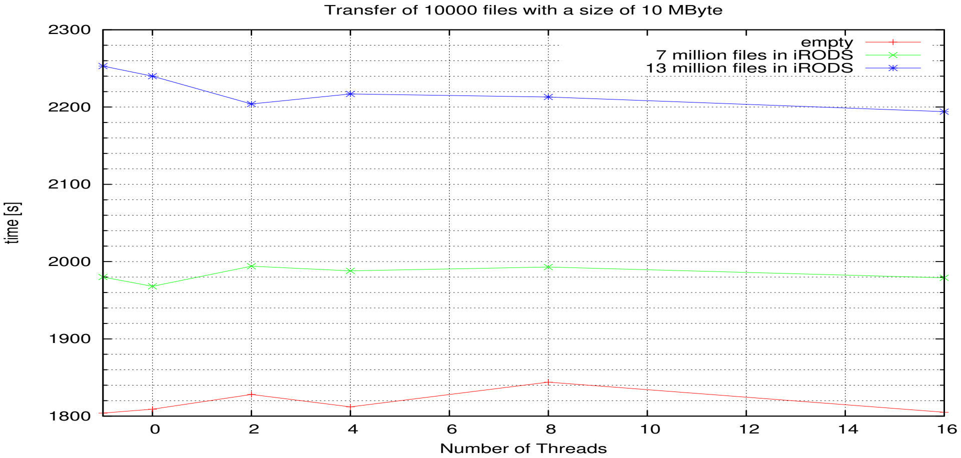


Fig. 8. Use case: 10,000 files (each 10 MByte) were written as part of one directory to the iRODS environment. Shown is the time needed for the transfer as a function of the threads/parallel transfers used. The thread number -1 means automatic thread choice by iRODS and 0 means no threading.

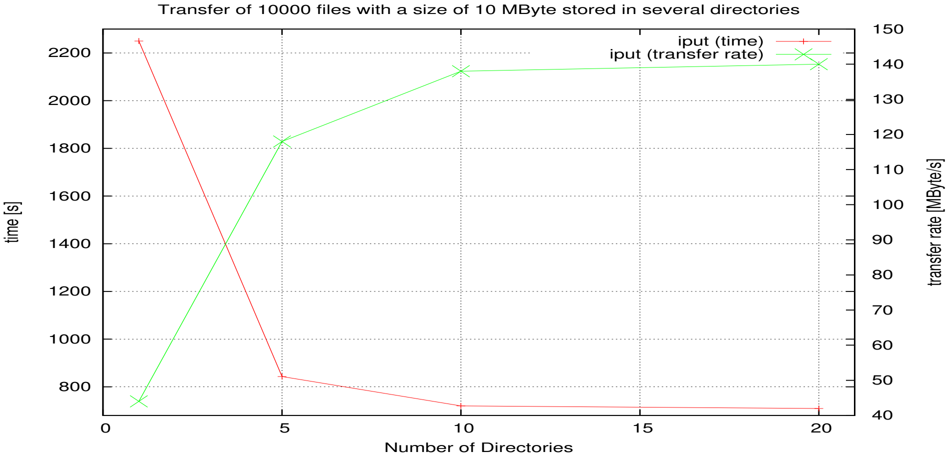


Fig. 9. Use case: 10,000 files are stored in several directories and are then written in parallel. Shown is the time needed for the transfer of the directories containing the 10000 files.

References

[1] Guido Juckeland and Stefan Börner and Michael Kluge and Sebastian Kölling and Wolfgang E. Nagel and Stefan Pflüger and Heike Röding and Stephan Seidl and Thomas William and Robert Wloch: BenchIT - Performance Measurement and Comparison for Scientific Applications. Parallel Computing: Software Technology, Algorithms, Architectures and Aplications, Elsevire Science, The Netherlands, pp 501-508, 2004

[2] http://www.benchit.org/

[3] Reagan Moore and Arcot Rajasekar: IRODS: Integrated Rule-Oriented Data System. White Paper: IRODS: Integrated Rule-Oriented Data System (2008)

[4] H. Shan and J. Shalf: Using IOR to analyze the I/O performance of HPC platforms Cray Users Group Meeting (CUG) 2007, Seattle, Washington, May 7-10, 2007

[5] Ben Martin: IOzone for filesystem performance benchmarking Linux.com, http://www.linux.com/archive/feature/139744, Retrieved 2009-10-15.

[6] Denis Hünich: Grid-Datenmanagement mit iRODS - Entwicklung von Komponenten zur Performance-Analyse. Diploma thesis, 2009

[7] Yoshimi Iida: iRODS perfomance and KEK, UK e-Science workshop ”Building data grids with iRODS”, NeSC Edinburgh, 27 May - 30 May 2008.

[8] Cesar Augusto Sanchez Baquero: Performance comparison between iRODS and Hadoop Distributed File Systems, Universidad Nacional de Colombia, 15.June 2009

[9] HADOOP Website: The Hadoop Distributed File System (HDFS) http://hadoop.apache.org/common/docs/current/hdfs_design.html

[10] iRODS Website: Ingestion Testing, https://www.irods.org/index.php/iCATStressTest_at_SDSC

[11] iRODS Website: General Query Stress Testing, https://www.irods.org/index.php/icatGenQueryStressTest_at_UMIACS

ISBN 978-83-60810-27-9
ISSN 1896-7094

Service level agreements for job control in high-performance computing

Roland Kübert
High Performance Computing Center Stuttgart
University of Stuttgart
Stuttgart, Germany
Email: kuebert@hlrs.de

Stefan Wesner
High Performance Computing Center Stuttgart
University of Stuttgart
Stuttgart, Germany
Email: wesner@hlrs.de

***Abstract*—A key element for outsourcing critical parts of a business process in Service Oriented Architectures are Service Level Agreements (SLAs). They build the key element to move from solely trust-based towards controlled cross-organizational collaboration. While originating from the domain of telecommunications the SLA concept has gained particular attention for Grid computing environments. Significant focus has been given so far to automated negotiation and agreement (also considering legal constraints) of SLAs between parties. However, how a provider that has agreed to a certain SLA is able to map and implement this on its own physical resources or on the ones provided by collaboration partners is not well covered. In this paper we present an approach for a High Performance Computing (HPC) service provider to organize its job submission and scheduling control through long-term SLAs.**

***Index Terms*—Service Level Agreements, High Performance Computing, Cloud Computing**

I. Introduction

FOR High Performance Computing resources scheduling of jobs is still realized in most cases using simple batch queues. While batch queues like OpenPBS [1], TORQUE [2] or others offer a quite comprehensive set of functionality for placing jobs in appropriate queues and optimizing the load of the cluster systems, also across sites, there is no mapping from business level requirements down to the low-level specifications. Low-level specifications are typical elements of a job description, for example desired number of CPUs, maximum wall- or run-time. The provision of such low-level properties requires a high level of expertise of the user and can only be specified if the target platform is pre-determined as different node and CPU architectures require different values. Additionally the number of queues is limited and therefore requirements have to be mapped to a particular queue. While advanced reservation, allowing a pre-defined start time, can be specified the drawback of potentially significantly reduced efficiency due to fragmentation of the schedule is not mapped to potential business penalties such as dynamically adapted pricing for such requests depending on the concrete loss in efficiency.

If an HPC provider wants to offer its services as utilities and aims to map different possible flavors of the services on different queue structures, the following problems can occur:

- Typically the use of certain queues is mapped to Unix credentials and groups. So all users of a certain group can or cannot use e.g. the express queue. However, depending on time of day or load situations, the “express queue service” might not be available to the same group of users all the time.
- While queues for specialized nodes (for example with graphics processing units (GPUs) or high memory nodes) are underutilized and normal nodes are oversubscribed there is no way to allow clients and providers to agree on special “discounts” for them. An automatic movement from normal to premium node queues would require interaction with accounting services.
- Customers might want to differentiate quite fine-grained about the treatment of their jobs. In such cases nowadays manual movement of jobs within the queues to “prioritize” them might be agreed beyond existing queue structures and group memberships. Such manual interactions cannot scale.
- Not all elements describing the Quality of Service (QoS) or Quality of Experience (QoE) can be mapped on queue properties and parameters. The overall service covers a wider range of properties such as the availability of a certain compute environment, application versions and licenses, proper treatment of data or specific configurations of the cluster system such as “require logical partition to isolate from other users”.

This limitation that only a few functional parameters can be specified when submitting a job (also reflected in standards like the Job Service Description Language (JSDL) [3]) means that there is basically no way for the user to express his requirements on a QoS or QoE level. Considering that the HPC provider is offering a utility potentially replaceable with other providers there is a clear gap between the demands from the user side and current offerings.

As a result a significant amount of work has been spent on realizing SLA frameworks allowing to mutually agree on the terms of the service between provider and consumer. However, while these frameworks cover well the necessary steps to realize SLAs also as a legally binding agreement, the concrete content of such an SLA and more important how these terms

978-83-60810-27-9/09/$25.00 © 2010 IEEE

can be guaranteed and provided from the service provider side are not adequately addressed.

So far we have the possibility for the consumer to express the requirements and agree the terms with the provider but

- terms within the SLA are not on the desired business level but mimic the low-level properties of the underlying queuing systems and
- the agreement process is typically detached from the underlying infrastructure such as current load situation of different resources, priority and importance of the consumer in a Customer Relationship Management (CRM) system and the accounting and billing services.

Consequently there is a gap between the demand of defining business level SLAs and their implementation using available methods and tools for the management of them on different type of computing facilities ranging from commodity of the shelf (COTS) clusters over specialized compute systems to cloud computing and storage systems.

Management systems on the provider side between the SLAs agreed with the consumer and the concrete physical resources need to interact with a range of different elements within the providers IT infrastructure and must look beyond individual SLAs to optimize the overall operation of all resources within a HPC computing service provider.

II. Related work

There are various approaches to the usage of service level agreements for job scheduling. While they differ in many respects - detail of the presentation, assumed parameters, implementation level, etc. - they all share the fact that they treat SLAs as agreements on a per-job basis. That means that, for each job to be submitted, a unique SLA is established before the job can be submitted. In [4], Yarmolenko et al., after having identified the fact that SLA-based scheduling is not researched as intensively as it could be, investigate the influence of different heuristics on the scheduling of parallel jobs. SLAs are identified as a means to provide more flexibility, increase resource utilization and to fulfill a larger number of user requests. Parameters either influence timing (earliest job start time, latest job finish time, job execution time, number of CPU nodes) or pricing. They present a theoretical analysis of scheduling heuristics and how they are influenced by SLA parameters and do not investigate how the heuristics might be integrated into an alredy existing setup. The same authors identify in [5] the need to provide greater flexibility in service levels offered by high-performance, parallel, supercomputing resources. In this work they present an abstract architecture for job scheduling on the grid and come to the conclusion that new algorithms are necessary for efficient scheduling in order to satisfy SLA terms but that little research has been published in this area. MacLaren et al. come to a similar conclusion, stating that SLAs are necessary in an architecture supporting efficient job scheduling [6].

SLAs that express a job's deadline as central parameter for deadline-constrained job admission control have been investigated by Yeo and Buyya [7]. The main findings were that these SLAs depend strongly on accurate runtime estimates, but that it is difficult to obtain good runtime estimates from job traces.

Djemame et al. present a way of using SLAs for risk assessment and management, thereby increasing the reliability of grids [8]. The proposed solution is discussed in the scope of three use cases: a single-job scenario, a workflow scenario with a broker that is only active at negotiation-time and a workflow scenario with a broker that is responsible at runtime. It is claimed that risk assessment leads to fewer SLA violations, thus increasing profit, and to increased trust into grid technology.

Dumitrescu et al. have explored a specific type of SLAs, usage SLAs, for scheduling of grid-specific workloads using the bioinformatics BLAST tool with the GRUBER scheduling framework [9]. Usage SLAs are characterized by four parameters: a user's VO and group membership, required processor time and required disk space. The work analyzes how suitable different scheduling algorithm are. Additionally, it comes to the conclusion that there is a need for using good grid resource management tools, which should be easy to maintain and to deploy.

Sandholm describes how a grid, specifically the accounting-driven Swedish national grid, can be made aware of SLAs [10]. It is presented how the architecture can be extended with SLAs and it is stated the greatest benefit would be achieved by insisting on formally signed agreements.

A comprehensive overview of resource management systems and the application of SLAs for resource management and scheduling is given by Seidel et al. [11]. The connection of service level and resource management to local schedulers is clearly shown as a gap in nearly all solutions.

In summary, it can be said that isolated aspects of the usage of SLAs have partly been investigated in detail: scheduling algorithms and heuristics, abstract architectures, parameters which are to be used as service levels, SLA negotiation etc. Gaps, however, can be easily identified: the analysis of the "big picture", that is the composition of individual aspects of SLA usage into a complete system and the integration of SLAs and SLA management with local resource management. This is not only true for the "traditional" field of high performance and grid computing but can also be extended to the field of cloud computing. Furthermore, SLAs are solely treated on a per-job basis, the analysis of SLAs as long-term contracts is not covered.

III. Benefits of SLAs for HPC service provisioning

The current operation model for high-end computing resources is conceptually still the same as fifty years ago where users placed a set of punching cards at the registry desk. The only difference is that users now can submit their compute jobs to a set of different queues and instead of the human operator the scheduling system is picking the jobs from the different queues depending on defined policies aiming for an optimized load of the system partially reaching 99% utilization. The major shortcoming of this approach is that the optimization strategy defined by the queues and the scheduling system policies is oriented towards a global optimization rather than an individual service offer.

If a user needs a special service (e.g. guaranteed start time of a job during a demonstration, interactive visualization or exhibition) beyond regular job submission the negotiation is typically done directly with the system operator and the performed steps are mostly done manually.

The availability of multi-core CPUs will lead to compute nodes with 32 cores and more in the near future, the rise of GPU-based computing with several hundred "cores" per card allows a reasonable number of applications to run on a single node. This is particularly true if the application is not targeting for a high-end simulation e.g. in the area of Computational Fluid Dynamics (CFD) domain with a very fine-grained mesh but more on exploring the problem space. Other examples are cases where the full simulation has been done before and now only small changes in geometry are done interactively demanding much less intensive computing to reach a stable state again as it is based on the previously achieved results.

Driven by the availability of cloud service providers and emerging products such as the Amazon Cluster Compute Instances also high-end computing service providers change their offers to be more *elastic* and realize a more *dynamically changing* infrastructure having certain queues available only during specific time periods or realizing a dynamic allocation of resources to logical partitions depending on the load situations or specific time bound agreements.

The pre-dominant use of high-end computing services will continue to be highly scalable technical simulations demanding a large number of compute nodes for exclusive use. However additional use cases have emerged driven by changes on the hardware level and competition with cloud service providers in particular for small scale simulations. The exclusive access for a user to one single node might even for compute intensive applications become a relic of the past. This substantially more complex management model for HPC service providers that cannot rely anymore on a quite homogeneous user behavior and long running jobs demands for a more complex management solution for operating their resources. The challenge is to integrate the demands of policies from different levels such as business policies (e.g. users with highly scalable and long running jobs should experience a preferred treatment) with more short term policies reacting on the current load situation (e.g. reducing prices or accepting more small jobs to fill gaps in the current schedule) and the demand of the users on a per-job basis.

The following sections aim to cover in examples the three major use cases driving the need for an SLA-guaranteed HPC service provision. Abstracting from concrete cases three different cases can be identified:

A. Interactive Validation

In many areas simulations have already replaced real experiments or physical prototypes during the development process. However at certain control points in the process simulation results have to be verified using physical prototypes. Within the IRMOS project augmented reality techniques are used to overlay real experimental data like a smoke train in the wind channel with a visualization of trace lines from the corresponding simulation. This "hybrid" prototype allows experts to directly compare the behavior of the real prototype with the results of the simulation. Such a design review session typically involving several people of a development team spread around the globe demands a fixed availability of the wind-tunnel, the computing resources, the visualization resources, the corresponding network resources and all involved experts, for example via video-conferencing.

In such a scenario simulation data will be generated continuously by a simulation running on a compute resource that is directly connected to the visualization resources. The current configuration of the wind channel like the air speed will be communicated as boundary condition for the simulation, thus the same parameters for both will be used while the experiment is running. This requires a coordinated and automated provision of the resources involved in the overall setting.

Such a scenario cannot rely on batch queue-based access as the computing and simulation part is just one piece in the overall setting. The demand for a co-ordinated availability also opens questions on how penalties are applied if one of the pieces in the overall setting is failing. For example if the compute resources are not provided as promised in time and the wind tunnel cannot be used the costs for it still accumulate. This applies also the other way around if the wind tunnel is not available or fails to communicate the boundary conditions for the simulations or the network connection is not delivering sufficient bandwidth.

As the resources needed for the full scenario are provided by different organizational entities the different quality levels needed by each individual contributor need to be put in a formal SLA, covering the terms of service as well as the agreed penalties in case of failures.

B. Guaranteed Environment

As outlined in [12] beside quality constraints there is also a demand to ensure a certain environment or other procedural constraints such as data handling, security policies or environmental properties (version of the operating system, available Independent Software Vendor (ISV) applications, etc.).

This is especially necessary for simulations performed as part of an overall design cycle for a complex product such as a car or airplane. A software environment is freezed for a full development cycle in order to ensure reproducible simulation results. This fixed environment is typically ranging from operating system over certain versions of numerical libraries up to application codes. A typical approach to address this requirement is to have beside a paper-based SLA agreed for a design cycle period a dedicated computing resource with the requested environment.

Advances in virtualization technologies as well as the possibility to apply different boot images in diskless cluster environments allow a more flexible treatment. Using such technologies a potentially unlimited number of pre-defined images, or even user-defined images, might be provided. As not all environments can be provided on all compute resources there must be a negotiation process between the user and the provider where a certain environment is demanded (e.g. expressed in a certain SLA bundle such as "Silver") and a corresponding reply about the conditions for the different options from the provider side is delivered.

The increased flexibility would allow to offer customized environments not only to large customers asking for resources for a long time period but also for users looking to meet their peak demands with outsourcing avoiding tedious customization activities of the environment reducing the entry gap.

C. Real-time Constraint Simulations

With the increasing role of simulations in design processes for complex products the demand to have a time-boxed simulation where results need to be delivered in time have emerged. This might be a set of simulations exploring a parameter space as input for a meeting of engineers the other day deciding on the focus for the future (long running simulation jobs). Another possibility is if the results of one single simulation (or a set of simultaneously running simulations) is the input to support an expert in taking a decision.

One important application area demanding for such an operation model is individualized patient treatment. For example in [13] a scenario for using simulations to validate different options to perform a bone implant for a specific patients is presented. In such cases the expert that needs to make a treatment decision has to ensure in advance of starting the simulation at a specific compute service provider that the results will be available in time before the treatment must be executed.

In such a scenario a negotiation with several providers would be started in parallel in order to make a case-by-case decision to which provider the job will be finally submitted. Such a loose binding to a specific provider would also require similarly to the scenario in the previous section a guaranteed or user-provided environment making the different providers interchangeable.

D. SLA Service Provision Benefits

From all the scenarios above it becomes clear that a much higher diversity of the offered services must be expected in the future. The requirements of the different scenarios on the provider's infrastructure are quite diverging. Additionally the consumer requirements are contradictory to the goal of the providers reaching a very high level of utilization of the provided resources.

As a result service providers will need to

- offer a mix of different services in order to combine the benefit of best-effort services (high utilization) with the benefit of special services (high value and price),
- offer a framework allowing consumers and providers to agree on the specific conditions for the service and
- actively manage their resources in a way that agreed SLAs are met, resources are most effectively used and any failures and incidents on the resource level are managed to avoid any impact on the agreed service levels.

The underpinning assumption presented in this section is that the provision of SLA controlled services is beneficial for consumers *and* providers. Consumer can negotiate guarantees and specific properties of the provided services as needed enabling new use models for high-end computing resources as outlined above. The provider perspective is clearly driven by business benefits to deliver as a part of the differentiation strategy specific products rather than aiming for a cost leadership approach. Consequently there is a clear need from the consumer side as well as a clear motivation from the provider side to deliver also in the HPC domain SLA based services. In other words the current model where the user needs to fully adapt to the provided environment and access model is changed to a model where the provider is offering certain possibilities or a kind of toolbox where the consumer can arrange the service offer according to their needs. Realistically this space of options needs to be discrete and limited allowing a management of the service offer from the provider side.

IV. Using long-term service level agreements for job control

Service level agreements, when they are used for the scheduling of compute jobs, are normally assumed to be on a per-job basis. That means that an individual SLA only contains terms for one specific job and a new SLA needs to be established for each job (see for example [14], [7] and [8]). This may be ideal to investigate the influence of SLAs and parameters specified therein on the scheduling of jobs in an isolated environment but does not correspond with the reality of how contracts are handled at HPC providers. At HPC providers, users usually agree to a contract that specifies charges for computational times and storage for available machines [15]. Jobs are then submitted in accordance with the acknowledged charges which therefore can be thought of as a long-term contract. This contract is, however, missing a specification of service levels. There may be some service level-related parameters specified - for example the availability of different machines to users and their characteristics, for example CPUs per compute node and memory size per node, but these are only specified in order to compute the amount finally billed to the user. By adding service levels to this contract, a long-term service level agreement is formed.

Long-term SLAs add the missing specification of service levels but keep the familiar contract behavior used by HPC providers intact. A simple specification of priorities, for example, might be realized through the following service levels:

Bronze Computational time is cheap, but there is no assurance on the scheduling of a job. This corresponds to the best-effort services provided today at HPC centers.

Silver Moderate prizing for computing time due to prioritized scheduling. Silver jobs can have timing

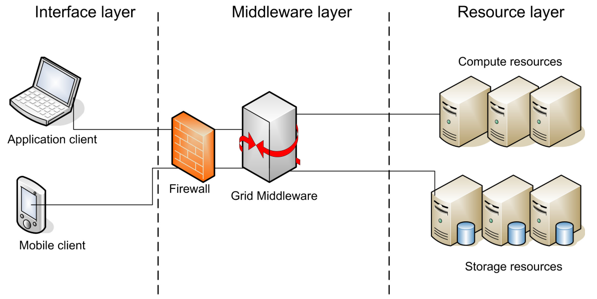


Fig. 1. Layered architecture

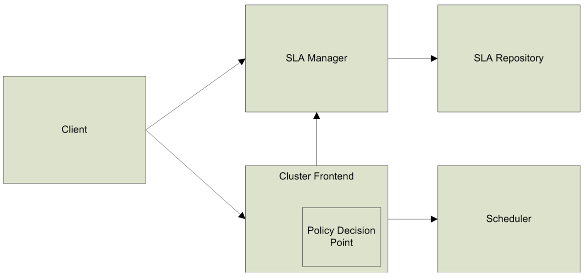


Fig. 2. High-level SLA management components

guarantees and might preempt best-effort jobs. The increased prize is justified since guarantees on the job's scheduling are given.

Gold High-prized jobs that are only rarely used, for example for urgent computing when computations need to be started immediately.

In contrast to the current situation the possibility of providing different service levels allows users to potentially have multiple contracts in place in parallel. On job submission time, a user decides which contract to reference in the submission depending on the current requirements and conditions such as urgency of the simulation result, load situation of the provider(s) etc. This can be seen as a using the middle way between using SLAs on dynamic, per-job basis and solely having singular long-term contrasts. In contrast to dynamic, per-job SLAs, this approach reduces the amount of negotiation as only few contracts are in place. Additionally, it avoids the problem that an urgent job cannot be submitted due to a failed negotiation. In contrast to having only singular contracts, this approach is more flexible and allows users to choose necessary priorities depending on the prize they like to pay.

V. An integrated approach to service level management

The basic service level approach given above can be realized with current techniques, for example the usage of specialized priority queues; however, providing more complex service levels cannot be realized that easily but requires the integration of service level management techniques across interface, middleware and resource layer.

Figure 1 shows the typical three-layered setup that is used by HPC providers. The left-hand sides shows clients of the HPC provider, either static or mobile[1]. The middleware layer is positioned between the client and the low-level resources and serves as a central entry point to the HPC provider's system and provides access through grid middlewares, for example the Globus Toolkit. The Grid middleware takes jobs submitted by the client and passes them on to low-level resources by means of a resource manager which employs a job scheduler in order to determine which jobs are placed on which resources.

A. *Service level selection by the client*

Enhancing the client with the ability to select service levels is very straightforward. This can be either done by changing the job submission client, adding SLA information to the message sent to the middleware or by integrating SLA functionalities into the application, if job submission is performed directly out of it.

B. *Enhancing the middleware*

SLA management on a middleware level has been investigated by various research projects and therefore different components and solutions already exist [16] [17] [18] [19] [20]. As these solutions often have the drawback of being very complex, a simpler solution is preferable, as it eases the amount of work necessary for installation, integration and maintenance.

Figure 2 is a diagram depicting how easily SLA management can be implemented on a middleware level and the underlying resource layer. The client thereby can either communicate with the SLA Manager, a central component on the HPC provider side responsible for SLA management, or submit jobs to the cluster front-end in the manner already explained.

The SLA Manager provides data regarding the long-term SLA contracts, for example contract information, accounting pertaining to contracts etc. It uses an internal SLA Repository for storing the contracts and other relevant information and is the central point that is queried by other components regarding SLAs. The cluster front-end, for example, on submission of a job, can query the SLA Manager for the validity of SLAs and can, after job completion, send accounting data to the SLA Manager.

[1] Mobile in this context should not be mixed with cellular phones and is understood as a nomadic user that is connecting from different locations without pre-defined IP addresses

The SLA functionality for the cluster front-end in the grid middleware can be realized in a non-intrusive way, for example through a policy decision point (PDP) that checks incoming requests and their SLA specification for validity. Incoming requests that do not contain SLA specifications can be mapped internally to a default SLA specifying a best-effort style service level, thereby realizing complete SLA-functionality and being backwards-compatible to clients.

C. Acknowledgement of SLAs

Honoring service levels of submitted jobs depends on the software used on this low level. Very simple schedulers, like the default scheduler supplied with the TORQUE resource manager, cannot honor service levels and need to be replaced by an SLA-enabled scheduler. This can be implemented by the provider itself, but this is a time-consuming and error-prone job. Rather, an SLA-enabled scheduler, for example the Moab Cluster Suite, should be used. It allows the formulation of quality of service levels for resource access, priority and accounting.

The providers main task is then to express the high-level SLAs offered to customers in such a way that the scheduler can implement them on the resource layer. Additionally, the incoming job requests have to be mapped to the corresponding service levels.

VI. From high-performance to cloud computing

In the previous sections we have elaborated a concept to enhance "classical" high-performance computing with service levels through the use of long-term service level agreements. Cloud computing, in many terms similar to the previous scenarios, seems like a logical step for the provisioning of services and can be a sensible offer to provide for HPC providers besides their usual role. Even though the term cloud computing is not clearly defined, it can be seen as distributed computing with virtual machines. Virtualization allows for more flexibility, scalability and abstraction of the underlying resources. Accounting and billing are usage-dependent. [21]

Cloud computing brings benefits both for consumers and providers. Virtualization allows the provider to use free resources for the execution for virtual machines as the underlying hardware is mostly irrelevant, although requirements specified by the user of course still need to be met. The usage of virtual machines means that the provider can offer a multitude of different environments tailored to customers, which was previously infeasible. Users might even be allowed to provide their own virtual machines, therefore giving them control over the complete environment.

Cloud computing began on a best-effort basis and many solutions provided today don't offer any more service [22] [23]. Service level agreements for cloud computing are, however, provided by some service providers, but they provide only minimal service levels [24] [25].

It has been shown that both Infrastructure as a Service and Platform as a Service - two types of cloud computing where the first one offers the concept described above and the second one offers a scalable, flexible but pre-defined environment to users - can benefit from service level agreements as well [26].

VII. Conclusions

The preceding work has described how an integrated approach to using service level agreements for the control of compute jobs allows HPC providers to offer support for various quality of service levels. Due to the solution including both high-level SLA management and low-level resource management and job scheduling, service providers can take advantage of service level agreements through the complete infrastructure. This is even valid for the recently introduced cloud computing paradigm.

The concept described above has been partially realized for an HPC scenario where the SLA management layer has been implemented and a simple integration with the grid middleware and resource layer has been achieved. Following the trend to provide Infrastructure as a Service solutions, we have decided to adapt the general concept for the Gridway meta-scheduler which is compatible with the OpenNebula cloud toolkit. This will enable HPC providers to offer IaaS with distinct service levels, which is not possible at the moment.

The offering of service levels can be a distinctive advantage for HPC providers as current contracts normally do not foresee the provision of service levels. Customers gain flexibility by having the possibility to choose between different service levels when submitting jobs. This also allows providers the option of offering previously unsupported service models, for example for urgent computing, which can generate a new revenue stream.

Acknowledgments

This work has been supported by the IRMOS project (http://www.irmosproject.eu/) and has been partly funded by the European Commission's IST activity of the 7th Framework Program under contract number 214777. This work expresses the opinions of the authors and not necessarily those of the European Commission. The European Commission is not liable for any use that may be made of the information contained in this work.

References

[1] Argonne National Laboratories, "OpenPBS Public Home," http://www.mcs.anl.gov/research/projects/openpbs/.

[2] Cluster Resources Inc., "TORQUE Resource Manager," http://www.clusterresources.com/products/torque-resource-manager.php.

[3] A. Anjomshoaa, F. Brisard, M. Drescher, D. Fellows, A. Ly, S. McGough, D. Pulsipher, and A. Savva, "Job submission description language (jsdl) specification, version 1.0," http://forge.gridforum.org/sf/go/doc12582?nav=1, [Online, accessed 8-March-2010].

[4] V. Yarmolenko and R. Sakellariou, "An evaluation of heuristics for sla based parallel job scheduling," in *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, April 2006, p. 8

[5] R. Sakellariou and V. Yarmolenko, *Job Scheduling on the Grid: Towards SLA-Based Scheduling*. IOS Press, 2008. [Online]. Available: http://www.cs.man.ac.uk/~rizos/papers/hpc08.pdf

[6] J. MacLaren, R. Sakellario, K. T. Krishnakumar, J. Garibaldi, and D. Ouelhadj, "Towards service level agreement based scheduling on the grid," in *Proceedings of the 2 nd European Across Grids Conference*, 2004, pp. 100–102.

[7] C. S. Yeo and R. Buyya, "Managing risk of inaccurate runtime estimates for deadline constrained job admission control in clusters," in *ICPP '06: Proceedings of the 2006 International Conference on Parallel Processing*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 451–458.

[8] K. Djemame, I. Gourlay, J. Padgett, G. Birkenheuer, M. Hovestadt, O. Kao, and K. Voß, "Introducing risk management into the grid," in *e-Science*. IEEE Computer Society, 2006, p. 28.

[9] C. L. Dumitrescu, I. Raicu, and I. Foster, "Usage sla-based scheduling in grids: Research articles," *Concurr. Comput. : Pract. Exper.*, vol. 19, no. 7, pp. 945–963, 2007.

[10] T. Sandholm, "Service level agreement requirements of an accounting-driven computational grid," Royal Institute of Technology, Stockholm, Sweden, Tech. Rep. TRITA-NA-0533, September 2005.

[11] J. Seidel, O. Wäldrich, P. Wieder, R. Yahyapour, and W. Ziegler, "Using sla for resource management and scheduling - a survey," in *Grid Middleware and Services - Challenges and Solutions*, ser. CoreGRID Series, D. Talia, R. Yahyapour, and W. Ziegler, Eds. Springer, 2008, also published as CoreGRID Technical Report TR-0096.

[12] S. Wesner, "Integrated management framework for dynamic virtual organisations," Dissertation, Universität Stuttgart, Stuttgart, Germany, 2008.

[13] R. Schneider, G. Faust, U. Hindenlang, and P. Helwig, "Inhomogeneous, orthotropic material model for the cortical structure of long bones modelled on the basis of clinical ct or density data," *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 27-29, pp. 2167 – 2174, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/B6V29-4VNH3RH-B/2/1be5c0dd92d2a3f8604519f9cad33a2e

[14] V. Yarmolenko and R. Sakellariou, "An evaluation of heuristics for sla based parallel job scheduling," in *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, April 2006, pp. 8.

[15] M. Resch, *Entgeltordnung fr die Nutzung der Rechenanlagen und peripheren Geräte des Höchstleistungsrechenzentrums Stuttgart (HLRS) an der Universität Stuttgart*, http://www.hlrs.de/fileadmin/_assets/organization/sos/puma/services/Entgeltordnungen/Entgeltordnung_16-09-2008.pdf, 2008.

[16] BEinGRID Consortium, "BEinGRID project home page," 2008, http://beingrid/.

[17] BREIN Consortium, "BREIN project home page," 2008, http://www.eu-brein.com/.

[18] IRMOS Consortium, "IRMOS project home page," 2008, http://irmos-project.eu/.

[19] NextGRID Consortium, "NextGRID project home page," 2008, http://nextgrid.org/.

[20] FinGrid Consortium, "FinGrid project home page," 2008, http://141.2.67.69/.

[21] C. Baun, M. Kunze, J. Nimis, and S. Tai, "Web-basierte dynamische it-services," 2009.

[22] E. Systems, "Eucalyputs - your environment. our industry leading cloud computing software." [Online; accessed 22-June-2010].

[23] O. P. Leads, "Opennebula: The open source toolkit for cloud computing," [Online; accessed 22-June-2010].

[24] A. W. S. LLC, "Amazon EC2 SLA," http://aws.amazon.com/ec2-sla/, [Online; accessed 2-March-2010].

[25] M. Corporation, "Download details: Windows Azure Compute SLA document," http://go.microsoft.com/fwlink/?LinkId=159704, 2010, [Online, accessed 2-March-2010].

[26] G. Gallizo, R. Kuebert, K. Oberle, A. Menychtas, and K. Konstanteli, "Service level agreements in virtualised service platforms," in *eChallenges 2009, Istanbul, Turkey*, 2009.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 663–670

ISBN 978-83-60810-27-9
ISSN 1896-7094

A Modeling Language Approach for the Abstraction of the Berkeley Open Infrastructure for Network Computing (BOINC) Framework

Christian Benjamin Ries
University of Applied Sciences Bielefeld
Computational Materials Science & Engineering
Wilhelm-Bertelsmann-Str. 10,
33602 Bielefeld, Germany
Christian_Benjamin.Ries@fh-bielefeld.de

Thomas Hilbig, Christian Schrñder
University of Applied Sciences Bielefeld
Computational Materials Science & Engineering
Wilhelm-Bertelsmann-Str. 10,
33602 Bielefeld, Germany
{Thomas.Hilbig, Christian.Schroeder}@fh-bielefeld.de

***Abstract*—BOINC (*Berkley Open Infrastructure for Network Computing*) is a framework for solving large scale and complex computational problems by means of public resource computing. Here, the computational effort is distributed onto a large number of computers connected by the Internet. Each computer works on its own workunits independently from each other and sends back its result to a project server. There are quite a few BOINC-based projects in the world. Installing, configuring, and maintaining a BOINC based project however is a highly sophisticated task. Scientists and developers need a lot of experience regarding the underlying communication and operating system technologies, even if only a handful of BOINC related functions are actually needed for most applications. This limits the application of BOINC in scientific computing although there is an ever growing need for computational power in this field. In this paper we present a new approach for *model-based development* of BOINC projects based on the specification of a high level abstraction language as well as a suitable development environment. This approach borrows standardized modeling concepts from the well-known *Unified Modeling Language (UML)* and *Object Constraint Language (OCL)*.**

I. Introduction

VOLUNTEER computing technologies allow to realize low-cost high-performance computing projects in certain application areas. A very prominent framework based on the principle of Public-Resource Computing (PRC) is BOINC (*Berkley Open Infrastructure for Network Computing*). BOINC provides an Application Programming Interface (API) with about one hundred functions of different categories, e.g. *filesystem operations*, *process controlling* and *status message handling* [1], [2]. A few of the most important functions are listed in section I-B. PRC is based on a server-client communication infrastructure mechanism. Here, the client retrieves a project specific application from the server along with a so-called workunit, i.e. a number of parameters usually provided in data files of simple ASCII or binary format that are optionally needed by the application to perform a specific task. BOINC is strongly focused on autonomic applications, each BOINC project has its own server, applications and tasks. The client executes the application, i.e. performs the calculations and sends its results back to the server which assembles these into a “global” result or stores these results at specific places.

The setup of a BOINC project heavily relies on the use of file-based scripting techniques. For instance, the programming language *Python* is utilized for the creation of a standard BOINC server infrastructure with database initialization, website and administration interface configuration and an optional BOINC test application. A few scripts are implemented using GNU BASH to sign executable files with encryption keys and yet another script uses the C shell (csh) to monitor network traffic. Extensible Markup Language (XML) files are used for the runtime server configuration. *All* files have to be edited *manually* by the project developer, scientist or administrator which bears the risk of making a large number of typical errors. For example, wrong spelling of parameter names or simulation relevant values as shown in I-A would have a significant effect on the system’s integrity and the application’s performance [8]. One way to cope with these problems is to provide a tool support which allows to automate the manipulation, generation, and checking of all necessary scripts.

In this paper we discuss a model-based approach for the development of BOINC projects that makes it possible to implement application specific changes while always keeping a valid configuration of the BOINC infrastructure. We give an idea on how to create a proper high-level domain specific language (DSL) in order to develop and maintain a complete BOINC application. This DSL forms the basis of an easy-to-use programming environment along with a high-level programming language and a suitable development process. Additionally, we present a way to model a complete BOINC installation including the client application for different target computer architectures and processor types. Aspects of single- and multicore processor units (CPU) and graphics processing units (GPU) are also discussed.

This project is funded by the German Federal Ministry of Education and Research

A. Typical errors during the BOINC server configuration process

The following list shows examples of typcial errors that can occur due to manual editing of the BOINC server configuration files. As a consequence of these errors one can expect a significant effect on the system's integrity and application performance.

- *uldl_dir_fanout* is the parameter that contains the number of subdirectories inside the upload and download directories on the server computer. A wrong value set here may dramatically slow down the system's server performance because of too many hard disk drive accesses.
- *shmem_key* names the allocated memory that is needed for the interprocess communication (IPC) between all BOINC applications on every BOINC project server. It is required that this value is unique, never changed during the runtime and is used by all BOINC server applications.
- *msg_to_host* must be included in the BOINC server configuration to enable sending of trickle-down messages[1] to the BOINC client nodes.
- *tasks* describes a set of parameters for applications which should execute in a cycle period, i.e. a crontab. Suitable values are needed to avoid problems like extremely large logging files or a outdated statistics, i.e. how many workunits are left for working or have errors during the computation.
- *daemon* contains a set of command descriptions. It is useful to start more than one daemon process to get a good load balancing of user requests, e.g. when the BOINC projects are much in demand.

One of our goals is the automatic determination of these most important parameters for different target computer architectures and processor types [20].

B. The BOINC Application Programming Interface

BOINC offers few example applications in which the number of lines of code range from 38 to 308. The first one only includes some elementary functions and no BOINC specific commands, e.g. a for-loop which just keeps the processor busy for one second. The second one is a more useful example since it contains BOINC specific function calls, e.g. how to retrieve the name of the checkpoint file or the actual processing state. Implementing a complex scientific application using BOINC is far more complicated and requires a broad experience of the developer. However, one can show that only 23 different BOINC functions are necessary to create a successfully running research relevant distributed computing application [23], [21].

II. State of the Art

Model-driven engineering (MDE) is becoming the dominant software engineering paradigm to specify, develop and maintain software systems. For example, Brunelière *et al.* [6] propose a *Modeling as a Service* (MaaS) initiative for cloud computing projects with an emphasis on topics like scalability, tool interoperability, and the definition of modeling mash-ups as a combination of MDE services from different vendors. Moreover, the definition of domain-specific languages (DSL) [10], [26], [27] along with the development of tools which support the developer during the DSL conception [9], [17], [26] are currently under investigation. However, none of the above mentioned approaches is related to public resource computing nor can it be directly used for the modeling process of BOINC projects. In the following chapters we therefore propose a first modeling language approach for the abstraction of the BOINC framework.

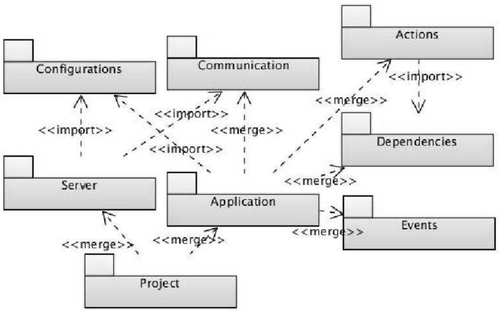


Fig. 1. Logical packages and dependencies of the BOINC functionalities

III. Abstraction of the BOINC Framework

The BOINC framework offers many very useful functionalities that help the developer to create his application. As a first step towards our modeling language approach we have subdivided the BOINC functionalities into different logical packages as shown in Fig. 1. Each package contains functions that cover a specific aspect during the development process [22] and can be used independently from functions of other packages which minimizes the number of dependencies. The whole BOINC project including the server installation components and application specific implementations is contained in the package *Project*. This package directly depends on the packages *Server* and *Application*. The package *Application* contains all application specific implementations and depends on the following child packages:

- *Events* - This package describes the abstractions for all possible events that can occur during the execution, e.g. exceptions like *'File not found'* or *'Segmentation fault'*. It contains routines for clearly defined error handling.
- *Actions* - Every execution statement is gathered within this package including wrapper routines to call third-party applications, i.e Matlab, or other domain-specific simulation tools.
- *Dependencies* - All libraries that are needed for the whole project are contained in this package. This includes the BOINC libraries as well as application specific runtime libraries and external sources.

[1] Trickle messages are asynchronous, ordered, and reliable messages between the BOINC server/clients and let applications communicate with the server during the execution of a workunit.

- *Communication* - This package includes the BOINC core client component which is the interface between the client installations and project servers and performs the information exchange between them.
- *Configurations* - In this package one keeps all system relevant parameters coded in XML files.

The complete server installation and maintenance process is provided by the package *Server*. This package describes the relationships between all components of a complete BOINC server installation, i.e. all configuration files, a list of the parameters for the workunits, description of installed applications with the corresponding architecture and processor targets. The package *Server* imports its required information from the contents of the *Configurations* and *Communication* packages.

IV. The modeling language approach for the abstraction of BOINC

Nowadays, models and model-based techniques are the fundamental means by which engineers are able to cope with otherwise unmanageable complexity and reduce design risk. In particular, software models have the distinct advantage that they can be *evolved* from high-level views of possible designs into actual implementations.

The *Unified Modeling Language (UML)* – a widely adopted, widely supported and customizable industry standard – plays a key role in modern software development. With the possibility of creating standardized UML profiles it provides the fundamentals for a true engineering-oriented approach to the construction of software. That is, system models can be used to understand and assess designs and predict design risks in meaningful (e.g., quantifiable) ways. Full automatic code generation from UML models facilitates preservation of proven model properties in the final implementation.

In our approach to a model-based development of BOINC projects we focus on the use of quasi standard software tools available within the Eclipse development environment. These tools enable us to develop all necessary components, like diagram editors, including graphical representations of modeling elements, code generators, etc. in one and the same development environment. Specifically, the code generation should be realized using a template engine which could also be used to generate important documentation files.

A. Graphical modeling environment

Throughout our project we exclusively use the Eclipse Modeling Framework (EMF) as released by the Eclipse Model Development Tools (MDT) project [26]. A detailed description of all components can be found in [9]. A key feature of our approach is the definition of a suitable standardized UML2 profile [22]. Here, we make use of the EMF-based implementation of the *UML2* and *UML2 Tools* subproject. EMF specifically allows implementing of constraints based on the Object Constraint Language (OCL) [18].

Fig. 2 gives an overview about a simplified modeling process. Here, the developer uses graphical modeling elements within diagrams to design an application as described in Sec. IV-B below. For a more detailed modeling of the application logic the developer is required to use a textual DSL as described in Sec. IV-D. The *Graphical Model* will be implemented using the Graphical Modeling Framework (GMF). GMF includes EMF, the Graphical Editing Framework (GEF) and Draw2D. GEF is a framework built upon the Model-View-Controller (MVC) pattern and handles the view and logical components with own instances. The controller handles the logic between these instances. The *Textual Model* and the *Graphical Model* are synchronized, i.e. every change in one of the models causes changes in the other one and each model will automatically be adjusted. Both models will be transformed into one (yet to be defined) so-called *VisualGridML Genmodel*. This can be done by exploiting the model-to-model transformation framework Query/View/Transformation (QVT). The QVT operation mapping language is capable of dealing with multiple input and output models and also supports OCL statements.

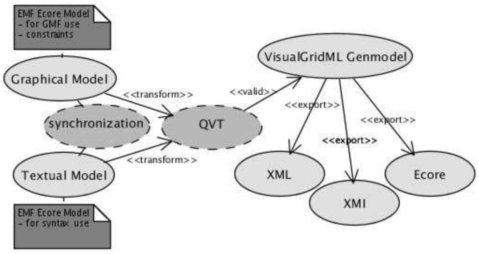


Fig. 2. Simplified view of the modeling process

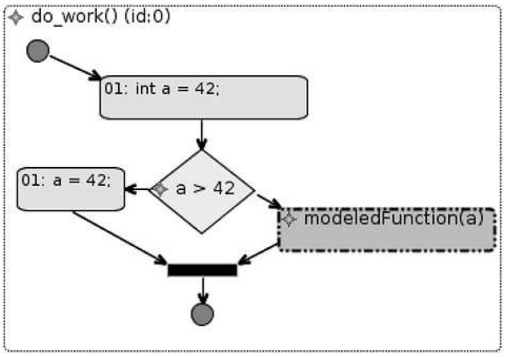


Fig. 3. Example flowchart diagram for instruction sequences.

One of the key issues of our approach is to define a proper *VisualGridML Genmodel*. The *VisualGridML Genmodel* can be exported into different other formats, e.g. XML, XML Metadata Interchange (XMI), or Ecore. The *VisualGridML Genmodel* will only be generated when the previously created models are valid. In order to support the developer during the verification and validation process adequate error messages and output comments will be created.

B. Graphical modeling elements

The proper definition of suitable graphical modeling elements is vital for our modeling language approach for the

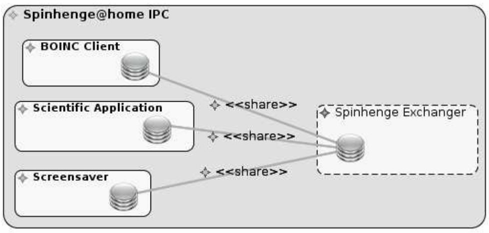


Fig. 4. Interprocess Communication within a BOINC application

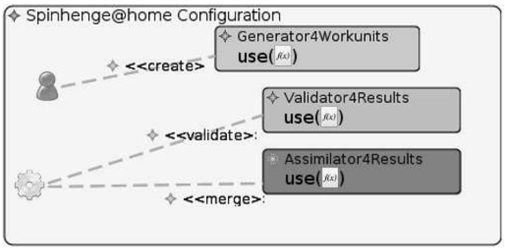


Fig. 5. Elements for configuration purpose

abstraction of BOINC. In the following we present some examples along with sample modeling fragments. Fig. 3 shows a flowchart of a high-level description of instruction sequences. The graphical notation is adapted from the UML2 flowchart definition [17, Fig. 12.36].

Fig. 3 shows an example that uses seven graphical modeling elements. As mentioned above these modeling elements are defined in the *Actions* package and have the following meaning [22]:

- *Start*, which describes the entry point of an application.
- *Stop*, which defines the end of execution, i.e. after that no other instructions will be executed and the application will shut down.
- *Action* is a modeling element which can execute native C/C++ instructions.
- *Decision* describes an `if-else` condition.
- *Join* merges two or more subdivided instruction sequences.
- *Execute* executes external C/C++ functions which are implemented in header and source files.

In Fig. 4 we show an adapted version of a UML2 collaboration diagram [17, Tab. 9.1, 9.2]. The dashed rounded rectangle on the right hand side specifies a shared data space. This data space is defined using one ore more port descriptors. Different ports could include different data descriptions and could also be connected with different so-called handlers. The three elements on the left hand side are examples of such handlers. Each handler handles a specific functionality and is connected with one application implementation. This example contains handlers for the *BOINC Core Client*[2], a scientific application, and a component for a screensaver session [1]. The handler can have only one data port. Data ports can only be connected to data ports of shared data space. Connections between ports may also be named as shown in this example.

In Fig. 5 we show an adapted version of a UML2 Use-Case diagram [17, Fig. 16.10] which contains elements of the *Configuration* package [22]. The elements on the left hand side are the actors for the use cases of the right hand side. The use cases contain a reference to predefined functions which can be modeled with a flowchart diagram. The dashed line connects the use cases with the appropriate actor and defines the type of execution. The lower actor describes the BOINC server which could be a cronjob. The upper actor describes a user. In this example the user defines workunits.

C. Aspect-Oriented Programming and Feature-Oriented Programming

Aspect-Oriented Programming (AOP) aims at separating and modularizing cross-cutting concerns [12]. The idea behind AOP is to implement so-called cross-cutting concerns as *aspects* and the core (non-cross-cutting) features are implemented as *components* [5]. Using *pointcuts*[3] and *advices*[4], an aspect weaver glues aspects and components at *join points* together. Fig. 6 shows on the left hand side (1) two aspects (A1 and A2) which extend the class definitions (C1 and C2). In AOP aspects can be added to the programming logic where ever functions are called. It is also possible to replace any functionality dynamically or to define the order of execution by precedence.

In Feature-Oriented Programming (FOP) the program functionalities can extended during the compilation and execution process, e.g. two or more functions can be combined to create extented features [7]. Fig. 6 shows on the right hand side (2) a simple overview of FOP. Here, F4 inherits the properties and methods of F1. Additionally, F6 refines F4, which can be done during runtime or while the compiling process creates the application.

Aspects and features in their current representation are intended for solving problems at different levels of abstraction [4], [14], [16]. Whereas aspects in AspectC++ [24] act on the level of classes and objects in order to modularize cross-cutting concerns, features act on the software architecture level [3].

For our approach, we are expecting that AOP and FOP are suitable methodologies for dynamic binding of applications as stated in Sec. IV-E. For example, the BOINC project result file format must be defined and created manually by the scientist or developer for a specific scientific application. This file can be generated during the code generation process as soon as definitions of the BOINC validator and BOINC assimilator are existing. After that, the BOINC project can be deployed with error-free validator and assimilator configurations.

D. Domain-specific language for BOINC

Domain-specific languages (DSL) are language definitions tailored to the development needs of specific problem domains

[2] The BOINC Core Client communicates with schedulers, uploads and downloads files, and executes and coordinates applications.

[3] The point of concern to execute an *advice*.

[4] Additional code that should apply to the existing model.

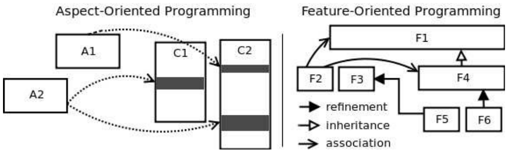


Fig. 6. (1) Two aspects extend two classes, (2) Two features refine two other features

[10]. For example the Structured Query Language (SQL) is designed for database queries. Another vital part of our modeling language approach to BOINC is a proper definition of a DSL for BOINC applications. Here, a key issue is the use of Xtext which allows to create an application by code generation. Xtext offers in combination with Xpand a template-based code generation engine [26]. By using Xtext and Xpand is has been shown that it is possible to create a complete BOINC application [21].

Generally speaking, all BOINC functionalities can be defined using a set of specific language elements. Our model-driven approach allows to develop applications independently of the target type, i.e. for CPU or GPU targets. Furthermore, BOINC offers various diagnostic parameters which enable or disable checks, e.g. *memory leaks* or *heap violations*. With the help of DSL statements these specific options can enabled or disabled for a subsequent automatic code generation. For example, the following code fragment defines a single processor environment which enables the above mentioned diagnostic flags:

```
target cpu single;

diagnostics {
 dumpcallstack
 heapcheck
 memoryleakcheck
 redirectstderr
 tracetostderr
}
```

In order to create applications with multicore or GPU computing support the following statement can be used:

```
target cpu mode multi with 10;
  // or
target gpu dim(10) block(4);
```

The first line describes a multi-thread application with up to 10 threads. The last statement enables the support of GPU computing. In GPU computing the process is splitted into *dim* threads, executed in 4 *blocks* [13]. It is also possible to include manual implementations or third-party libraries. The following examples show this in more detail. The required dependencies to third-party libraries or functionalities could be also defined with only a few lines of code. This code is used for the "make" process of an executable application. On Linux or Unix like operating systems a makefile is generated whereas on Windows systems it is possible to generate different project files which can be imported by integrated development environments like Visual Studio or Eclipse. Using this feature one can realize a *platform-independent* approach.

```
includes AppInclude {
 "~/boincadm/framework"
 "~/boincadm/src/api"
 "~/boincadm/src/lib"
}

libraries AppLibrary {
 "/lib", "pthread"
 "~/boincadm/framework", "visualgrid"
 "~/boincadm/src/api", "boinc_api"
 "~/boincadm/src/lib", "boinc"
}
```

It is common to use parameter files for the BOINC applications. The native way of doing this is to create *mapping files* on the server with a few parameters. Whenever clients connect to the server, they retrieve the application and all necessary parameter files. The following DSL fragment describes this procedure using the `infile` statement. The content is specified as an XML tree and could be iterated with a reference, e.g. `ObjectName1`. As a consequence of, binary data must be base64 encrypted [25]. Each reference contains the parameter as a *struct* or *class* definition.

```
infile "metropolis_data.xml"
                  as ObjectName1;
infile "param.jj" as ObjectName2;
infile "param.nn" as ObjectName3;
infile "param.ww" as ObjectName4;
```

After the execution of the client application the results are stored in *result files* defined by the statement `outfile` and are uploaded to the server.

```
outfile "metropolis_out.erg"
                  as ObjectResult1;
```

In general there exist several ways using a modeling process to develop a certain client application. As a matter of fact, every developer differs in his way to develop applications and it is necessary that the DSL supports this variety. For example, it is possible to link third-party libraries to the application using DSL statements. Furthermore, the application code can also be directly implemented into the `worker` part. The following DSL fragment contains an example which is used in [21]. Here, the `worker` starts the environment for execution instructions with the name `Spinhenge`. The statement `exec` defines pointcut expressions which are used by the AOP weaver process to deploy the scientific application. In this definition, the pointcut expression describes the working function in Fig. 8.

```
worker Spinhenge {
 exec "void %::Spinhenge::doWork(...)";
}
```

This statement could be replaced by other instructions, e.g.

```
worker Spinhenge {
 cpp {
  int a = 42;
 }
 action(modeledFunction(a));
}
```

Here, `cpp` starts an inline code area which contains native C/C++ statements. The variable `a` is available right after its definition and optional initialization within the context. It can be used by DSL defined functions like `modeledFunction(variable)`. Third-party applications can be executed using the DSL statement `wrapper`. It allows to call an external application, so called *legacy application*, and only one call is allowed to realize. Optional parameters for the application can be set in the *Configuration* package.

```
worker Spinhenge {
  wrapper("Matlab", "Argv[1] Argv[2]" [,
       weight, checkpoint_filename,
       fraction_done_filename, ...]);
}
```

These optional parameters are defined by the wrapper interfaces [11], [15], [19].

```
screensaver Spinhenge {
  render "% Screensaver
              ::Spinhenge::doWork(...)";
}
```

During the execution of an application different events can occur, e.g. events which could also be logged for diagnostic analysis in the above mentioned definitions. Furthermore an exception handling is described by the following DSL definition:

```
handle TypeOfException (: optionalName) {
 /* to be defined handler */
}
```

Predefined exception handlers are reusable by other exceptions. This is enabled by using the `ref` statement which uses the optional name `optionalName` of the previous example.

```
handle AnotherTypeOfException
   ref optionalName;
```

The BOINC framework uses interprocess communication (IPC) to exchange data between different application instances, e.g. scientific application, and screensaver. The native way to use IPC needs the definition of shared variables, e.g. in a C/C++ `struct` or `class` definition and furthermore different functions which handles these variables. A strict well-formed definition would be easier to use and reduces the effort of changing code parts in the source files. The DSL statement `exchange` describes the structure of the IPC with C/C++ language elements like datatypes which are usable in every application. The keyword `feature` defines an AOP pointcut expression which is used to assign values to the variable on the left hand side, e.q. `update_time` keeps the delta in milliseconds between the value updates. The listed AOP pointcut expressions are implemented in the *Visual Grid Framework* which is described in Sec. IV-E.

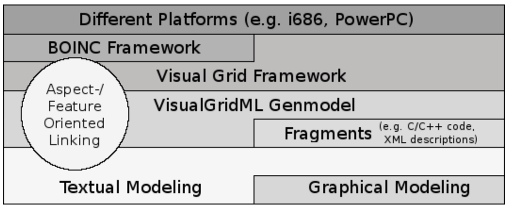

Different Platforms (e.g. i686, PowerPC)
BOINC Framework
Visual Grid Framework
Aspect-/ Feature Oriented Linking
VisualGridML Genmodel
Fragments
(e.g. C/C++ code, XML descriptions)
Textual Modeling
Graphical Modeling

Fig. 7. *Visual Grid Framework* Abstraction Layer

```
exchange {
 double update_time :
   feature "% Boinc::updateTime(...)";
 double fraction_done :
   feature "% Boinc::fractionDone(...)";
 double cpu_time :
   feature "% Boinc::cpuTime(...)";
}
```

To handle BOINC Trickle Messages the `TrickleUp` and `TrickleDown` commands are stated in the following listing. The `TrickleUp` needs two AOP pointcut expressions to check if a trickle up must handled and the trickle up handler itself.

```
 TrickleUp "bool checkTrickleUp(...)"
     do "% handleTrickleUp(...)";
 TrickleDown "% handleTrickleDown(...)";
```

The handler defined by `TrickleDown` is called frequently to manage the incoming messages by the BOINC server, e.g. command to abort one workunit or informations of the current BOINC credit points.

Furthermore, general descriptive information about the application can be defined with the statement `info`.

```
info {
 author="Christian Benjamin Ries";
 email="cries@fh-bielefeld.de";
 license="FH Bielefeld";
 description="Spinhenge@home Example";
 project="Spinhenge@home";
 version="3.16";
}
```

E. Visual Grid Framework

Fig. 7 shows the proposed *Visual Grid Framework* layer which is defined between the layer that describes the underlying computer hardware, the BOINC framework, and the *VisualGridML Genmodel*. The *Visual Grid Framework* offers a less complex access to the BOINC functionalities and handles the creation of applications for different platform targets, e.g. Intel 686 based 32-bit or 64-bit architectures.

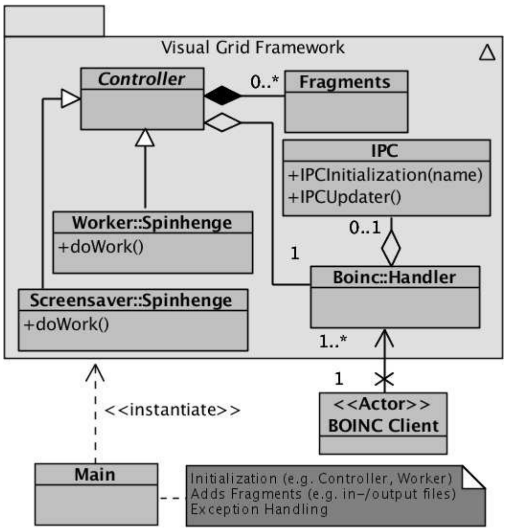


Fig. 8. Composition of the Visual Grid Framework Approach

The left side describes one approach to generate the function calls and logical program parts between the textual-/graphical modeling tools, fragments, *VisualGridML Genmodel*, *Visual Grid Framework*, and BOINC Framework. In Fig. 8 we show how the *Visual Grid Layer* works.

The `Main` class is generated by the code generation process and contains the implementation of the starting routines. This routine instantiates the interface to the BOINC framework as well as to the scientific application. The entry point of the scientific application is called using the *doWork()* functions of the *Worker::Spinhenge* and *Screensaver::Spinhenge* classes. These two classes are specializations of the abstract *Controller* class. The *Controller* class keeps track of all necessary data management, e.g. input, and output data files, checkpoint definitions, etc. The *IPC* class (Interprocess-Communication) is completely generated and implements the initialization and update routines. This class is used by the *Worker::Spinhenge* and *Screensaver::Spinhenge* classes for the exchange of data. The external *BOINC Client* class is an actor and therefore represents just the interface to the client. The BOINC Client has the full control of the executed applications and processes.

As a consequence, there exist two ways to generate an application within the *Visual Grid Framework*, (1) creation of a class which is derived from the abstract *Controller* class, (2) the definition of AOP *joinpoints* and *advices* as defined by the DSL in Sec. IV-D.

As a first test of our approach we have created an application which is similar to the BOINC sample to perform a transformation of lowercase to uppercase texts. The BOINC sample is based on approximately 400 lines of code, including the makefile, C++ header, source files, and 31 BOINC specific function calls. In contrast to this, our generated application contains only about 90 lines of code, including 60 lines of DSL code, and 30 lines of application specific code which performs the transformation. All defined dependencies, e.g. for the initialization of the process or exception handling, are resolved by generating files. Furthermore, a makefile is generated which on execution builds the complete application for the defined client platform architecture.

V. Conclusion

We have presented a first modeling language approach for developing Public Resource Computing applications on the basis of BOINC. We have demonstrated that the complete BOINC framework can be divided into a few logical packages that provide the necessary graphical and textual model elements to allow for a model-based development of applications with subsequent source code generation. Our approach is technically realized by using standardized and well-defined technologies [26]. Thus far, we have just implemented a small part of the BOINC functionalities. Key features like graphical and textual modeling elements, the *VisualGridML Genmodel* have been defined to an extent that we could show the general feasibility of our approach. However, further investigations with regard to the abstraction of the system architecture, dependencies, error checking, and the transformation to different target languages are needed. We have successfully performed a first test of our approach by modeling an existing BOINC application with just a few lines of DSL code using external libraries for the core computational routines and abstraction of the BOINC functionalities. However, most steps of our modeling process are still performed manually, and we are currently working on the creation of a unified development environment that supports the wide range of technologies, including a one and only graphical modeling framework which minimizes the need of textual modeling fragments, automatic dependencies resolving, completely error-free code generation, and higher support for legacy applications.

References

[1] D. P. Anderson, *BOINC: A System for Public-Resource Computing and Storage*, 5th IEEE/ACM International Workshop on Grid Computing. November 8, 2004, Pittsburgh, USA

[2] D. P. Anderson, C. Christensen, and B. Allen, *Designing a Runtime System for Volunteer Computing*, IEEE Computer, 2006

[3] S. Apel, T. Leich, M. Rosenmùller, and G. Saake, *FeatureC++: Feature-Oriented and Aspect-Oriented Programming in C++*, Technical Report, Department of Computer Science, Otto-von-Guericke University, Magdeburg, Germany, 2005

[4] S. Apel, T. Leich, and G. Saake, *Aspectual Mixin Layers: Aspects and Features in Concert*, In Proceedings of International Conference on Software Engineering (ICSE), 2006

[5] S. Apel and D Batory, *When to Use Features and Aspects? A Case Study*. In Proceedings of ACM SIGPLAN 5th International Conference on Generative Programming and Component Engineering (GPCE'06), Portland, Oregon, October 2006

[6] H. Brunelière, J. Cabot, and F. Houault, *Combining Model-Driven Engineering and Cloud Computing*, AtlanMod, INRIA RBA Center & EMN, France, Nantes, 2010

[7] D. Batory, J. N. Sarvela, and A Rauschmayer, *Scaling Step-Wise Refinement*, IEEE Transactions on Software Engineering (TSE), 30(6), 2004

[8] T. Estrada, M. Taufer, and D. P. Anderson, *Performance Prediction and Analysis of BOINC Projects: An Empirical Study with EmBOINC*, in J Grid Computing, Springer, 2009

[9] R. C. Gronback, E. Gamma, L. Nackmann, and J. Wiegand, *Eclipse Modeling Project, A Domain-Specific Language (DSL) Toolkit*, Addison-Wesley, 2009, ISBN: 978-0-321-53407-1

[10] A. Hessellung, *Domain-Specific Multimodeling*, IT University of Copenhagen, Denmark, 2008

[11] P. Kacsuk, J. Kovacs, Z. Farkas, A. C. Marosi, G. Gombas and Z. Balaton, *SZTAKI Desktop Grid (SZDG): A Flexible and Scalable Desktop Grid System*, Journal of Grid Computing, 2009

[12] G. Kiczales et al., *Aspect-Oriented Programming*, In Proceedings of European Conference ob Object-Oriented Programming (ECOOP), 1997

[13] D. Kirk, and W. W. Hwu, *Programming Massively Parallel Processors: A Hands-On Approach*, Morgan Kaufman Publ Inc, 2010, ISBN: 978-0123814722

[14] K. Lieberherr, D. H. Lorenz, and J. Ovlinger, *Aspectual Collaborations: Combining Modules and Aspects*, The Computer Journal, 46(5), 2003

[15] A. C. Marosi, Z. Balaton, and P. Kacsuk, *GenWrapper: A Generic Wrapper for Running Lagacy Applications on Desktop Grids*, 3rd Workshop on Desktop Grids and Volunteer Computing Systems (PCGrid 2009), 2009 May, Rome, Italy

[16] M. Mezini and K. Ostermann, *Variability Management with Feature-Oriented Programming ans Aspects*, In Proceedings of ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE), 2004

[17] OMG Adopted Specification formal/2009-02-02, *OMG Unified Modeling LanguageTM (OMG UML)*, Superstructure, Version 2.2, OMG, 2009

[18] OMG Adopted Specification formal/2010-02-01, *OMG Object Constraint Languag*e, Version 2.2, OMG, 2010

[19] C. B. Ries, and C. Schrñder, *ComsolGrid - A framework for performing large-scale parameter studies using Comsol Multiphysics and BOINC*, COMSOL Conference, Paris, France, 2010

[20] C. B. Ries, *Performance measuring and automatic calibration of BOINC installations*, University of Applied Sciences Bielefeld, Germany, unpublished

[21] C. B. Ries, T. Hilbig, C. Schrñder et al., *Spinhenge@home - Monte Carlo Metropolis*, Version 3.16, University of Applied Sciences Bielefeld, Germany, http://spin.fh-bielefeld.de

[22] C. B. Ries, T. Hilbig, and C. Schrñder, *UML 2.2 Profile: Visu@lGridML*, University of Applied Sciences Bielefeld, Germany, unpublished

[23] C. Schrñder, "Spinhenge@home - in search of tomorrow's nanomagnetic application", *to appear in Distributed & Grid Computing - Science Made Transparent for Everyone. Principles, Applications and Supporting Communities*, 2010

[24] O. Spinczyk, D. Lohmann, and M. Urban, *Advances in AOP with AspectC++*, Software Methodologies, Tools and Techniques (SoMeT 2005), IOS Press, September, 2005, Tokyo, Japan

[25] T. Imamura, B. Dillaway, and E. Simon, *XML Encryption Syntax and Processing*, W3C, December, 2002, http://www.w3.org/TR/xmlenc-core

[26] Xtext - programming language framework, Xpand - a template language, http://www.eclipse.org/modeling/mdt

[27] U. Zdun, *Concepts for Model-Driven Design and Evolution of Domain-Specific Languages*, In Proceedings of the International Workshop on Software Factories OOPSLA, pp. 1-6, October, 2005

ISBN 978-83-60810-27-9
ISSN 1896-7094

Green Methodologies in Desktop-Grid
An invitation for discussion by the DEGISCO project

Bernhard Schott
AlmereGrid and VCOdyne SAS, Le Chesnay, France
Email: bernhard.schott@vcodyne.com

Ad Emmen
Almere Grid, Almere, Netherlands
Email: ad@almeregrid.nl

***Abstract*—Desktop-Grids have been around since the very early days of Grid computing, scaling into the millions of PCs, well established as regular distributed computing infrastructure for many research projects. Desktop-Grids collect CPU cycles from PCs contributed by donors, by volunteers who are willing to support science and research. This paper focusses on the energy efficiency aspects of Desktop Grid computing: the Green Desktop Grid, as a task of the EU-FP7 DEGISCO project [20].**

The key advantage of Desktop-Grids with regards to Green-IT over service Grids and data centres based on clusters of servers is the minimal heat density. Compute Clusters without energy intensive air-condition would run into thermal disaster within minutes. PCs participating in Desktop-Grids usually do not make use of any air-condition.

We will have a closer look on several aspects of energy consumption and computational performance in Desktop-Grids describing several distinct Green Methodologies to optimize compute unit specific energy consumption.

I. The need for Green Desktop-Grids

DESKTOP-GRIDS provide compute-power to scientists by contributions of resource owners, typically individuals at home but also institutions and companies. Compute time harvested this way does not request large upfront investment by the scientist; it is a low cost approach towards significant scientific output. Typically implemented using BOINC [17], sometimes XtremWeb [19] or other packages, Desktop-Grids are found among the largest Distributed Compute Infrastructures (DCI) [1]. Also known as Volunteer-Computing, Desktop-Grids have been around since the very early days of Grid computing [18], scaling into the millions of PCs contributing compute time every day. The aggregations of so many machines result in significant performance well beyond Petaflop/s for selected applications. For example: BOINC network averages about 5.1 Petaflop/s as of April 21, 2010 [2]. Key difference to service Grids like EGEE (now EGI) is the voluntary character of the resources – citizens contribute their PC's compute time to the Desktop-Grid projects in order to support scientific challenges of their choice. The FP7 project DEGISCO aims to support Desktop-Grid deployments in and beyond Europe, especially countries that strongly collaborate with the European Union. A focus topic of DEGISCO is the energy efficient handling of Desktop-Grid workload and management of resources, provided as configuration advice to Desktop-Grid operators. DEGISCO is accompanied by the EDGI project that continues to maintain and further develop the EDGES-Bridge [3], a gateway transparently connecting gLite, Unicore, and KnowArc based infrastructures (Service-Grids) to Desktop-Grids by automated translation of the job-languages.

Why Green-Desktop-Grids? As stated above, Desktop-Grids can aggregate hundred-thousands of machines. Power consumption of such large amounts of devices should be considered when making use of them. And yes, when used for computation energy consumption of PCs (like of any other computer) goes up. In the end, the contributor, the volunteer, who allows and enables the use of her or his machine, not only provides compute time for free but also pays for the additional electrical energy to cover the computation induced power consumption. Key advantage of Desktop-Grids is the minimal power density compared to conventional data centres. Typically, PCs participating in Desktop-Grids in Europe are not hosted in air-conditioned environments. Without the energy burden of air-conditions, Desktop-Grid are intrinsically “greener” than data centre based clusters and thereof built Service-Grids.

Green IT has been a hype topic claimed by the hardware vendors: buy new, buy more power efficient machines and you will become “Green”. Supported by the EC (European Commission) issued “Code of Conduct on Datacenters” [4], the massive refurbishment or fresh built of data centres has been encouraged. The effective outcome in order to “Green” our planet lags behind expectations: number one reason is the lack of investment budget. Number two reason might be lack of interest – maybe due to the fact that energy is still too cheap: some data centre CIOs still do not know their energy-bill nor -cost structure. Efforts to improve the energy efficiency of data centres by minimal impacting methods, by reorganization of usage and workload distribution [5] have been presented in the course of OGF Green-IT working group, but not led to significant uptake yet.

The need for compute power, the progress of Computer Aided Science (CAS), is massive and unstoppable. Researchers from all sectors are urgently looking for more compute power. By today and well ahead of operations start, PRACE resources are already overbooked by a factor of 5 [6]. Although Desktop-Grid suitable workload is *not* HPC and only the lean data fraction of all HTC, significant scientific

DEGISCO is supported by the FP7 Capacities Programme under grant agreement nr RI-261561

output has been, is, and will be produced with their help [7]. With the growing importance of Desktop-Grids in the scientific process and the significantly growing deployments, the need to optimize the use of energy is obvious both for general environmental considerations as well as for the attraction of contributors. Citizens providing their machines are interested in finding their contribution used in the most optimal way, producing more science and less waste-energy.

II. DEGISCO Green Desktop-Grid Methodologies

The environmental impact of IT and its specific usage patterns have been investigated in the recent years manifold. In the course of the roadmap consultation process, DEGISCO gathers analysis and best practices on Green aspects of distributed computing, focused on Desktop-Grids, but also comparing to classical technologies like clusters and data centres. Conceptually different methodologies will be investigated, based on technology means like Desktop-Grid client based ambient metrics, application profiling or adjustment of energy consumption per time interval, exploitation of natural ambient conditions, and possibly more. Some of those methodologies are technological, some are pure organizational. DEGISCO will promote this methodology inside the International Desktop-Grid-Federation [8] especially as part of a road map. This federation will offer consulting and advice based on these methodologies, continuing the roadmap process to reflect and integrate future findings and developments. One focus topic in the roadmap process is the application of Green Methodologies to achieve reduction in carbon dioxide footprint of research infrastructures.

A. Aims of Green IT: CO$_2$ footprint and the energy mix

Green IT aims were originally formulated to reduce CO2 footprint of IT activities. As extended scope reduction of energy consumption in general and especially thermal emissions in metropolitan areas, both impacting a) global climate b) local (micro) climate and c) human quality of life. Production of CO2 and accordingly the reduction of CO2 footprint are difficult to measure from the perspective of a concrete IT activity like computation.

Even if energy consumption as such is accounted for, it depends on the local energy mix how much CO$_2$ this is equivalent to. The electrical energy mix, the combination of electrical energy sources, depends on national specifics. The electrical energy mix (Table 1) in Germany [9] includes a nuclear energy portion of 27,5%, scheduled to be phased out, while the French [10] one (78,3%) is stable on a higher level. Denmark [11] produces 25% of its electricity consumption from wind – sometimes up to 150% (when strong winds produce more electricity that the Denmark needs) causing negative energy prices at the spot market [12]. Neither the single PC owner who contributes compute time to a scientific project nor the operator of the Desktop-Grid server that offers the respective workload item is in control of the energy source used to produce electricity consumed by the computational effort. Desktop-Grids are natively internationally or globally dispersed. While a specific Desktop-Grid server and its hosted projects may be presented to users from specific communities, there is no effective mechanism in place to "steer" or "control" the workload to contributors of a specific location or region. And it seems completely beyond the means of Desktop-Grid computing to control the workload may be executed only on machines supplied with energy of a "Green" qualified source. It seems to be possible to offer workload for the execution in a specific region – given that the contributing person takes into account and chooses to comply with the according recommendations. Nevertheless, the contributors have little control or even knowledge on the energy mix of their regional electricity providers. With a more general understanding of environmental protection, it is difficult to prefer nuclear power plants with their unresolved radioactive waste issues over natural gas powered electricity generation, although emitting carbon dioxide. As being in control only for the second half of the energy life cycle, IT activities can still use environmental friendly policies targeting to reduce energy consumption in general. It seems adequate to rephrase the core aim of Green IT to: Save energy!

Table I.
Zero CO2 electricity sources by 2007

Energy Mix	**Total**	**Nuclear**		**Renewables**	
Electricity	[TWh]	[TWh]	%	[TWh]	%
Germany	607,0	167,1	27,5%	58,2	9,6%
France	572,2	448,2	78,3%	66	11,50%
Denmark	40,5	0	0,0%	10,2	25,2%

B. € - metrics for Green IT success

In order to measure the effectiveness of energy saving policies and methods, we need to introduce a metric that can be "metered". The obvious advantage of "kWh" as the base metric for Green IT is the simplicity of measurement: electricity is metered everywhere. Different from data centres and conventional Service-Grids, policies and methods are applied and executed in Desktop-Grids only by the volunteer effort of the resource contributor. As success metric for Green IT, the translation into cost, into money, is helpful to connect to business considerations and propel motivation. With € (for kWh) as metric, contributors can relate their choice of workload and policy-compliance to the personal electricity bill: Green Desktop-Grids help the planet and your budget!

III. Six Green Desktop-Grid Methodologies

DEGISCO starts with a shortlist of 6 methodologies which are collection of best practices, techniques and policies:

- Ambient metrics based Green optimization
- Cool strategy: avoid air-condition use
- Energy profiling of applications

- CPU speed steps
- Exploitation of natural ambient conditions
- Time-of-day dependent energy tariffs

In the course of the roadmap process these methodologies will be challenged, refined or replaced, according to feedback and feasibility tests supported by contributors.

A. Ambient metrics based Green optimization

In order to tune DEGISCO connected Desktop-Grids towards saving of energy suitable configurations and parameters are to be identified enabling the Desktop-Grid client to intelligently select adequate workload. A regular PC [13] almost doubles its power consumption from idle 160W to 300W under full CPU load. Ambient temperature measurement or at least estimation (compare below: On the Difficulties of Temperature Measurement) could be used to control and potentially prevent download of workload items if the PC and its environment are too hot for comfortable or safe operations. The measurement and observance of ambient conditions, mainly temperature, is essential for several advanced Green Methodologies, too.

B. Cool strategy: avoid air-condition use

Desktop-Grids are the real Green Grids: lower energy density than clusters results in less energy wasted for cooling. However, this may not longer be true if air-conditions are used to assure proper operation of Desktops. Principles of cooling: Energy consumption by air-conditions range from 30% to >200% of the energy dissipated by the IT device (payload), depending on the cool-reservoir temperature the heat pump can utilize to get rid of the heat. The Code of Conduct on Datacenters quotes that most European data centres are actually worse: they consume more than 200% of the IT related energy for cooling, UPS and power distribution losses. The prime advice to configure Desktop-Grids: avoid air-condition use! Selection criteria for the “maximum temperature” as described above could be that temperature which would just not yet trigger the start of the local air-condition.

What if air-condition is unavoidable?

Should we recommend to participate in Desktop-Grids when resources are located in hot ambient? The answer clearly depends:

- If the additional workload by Desktop-Grids would cause proportional air-condition power consumption, a different strategy could be considered. Maybe by restricting the acceptance of workload to night times would help.
- If the air-condition is in full power use anyway – like in tropical ambient – the additional heat dissipation during compute load processing may not impact the total energy balance too much.

Example: light building structures with poor thermal insulation and continuously running air-conditions are de-facto standard in sub-tropical and tropical regions globally. If we assume a 3,5kW air-condition (2-3 room flat, small house) to run non-stop in order to keep the ambient temperature 15°C below the 40°C outside, additional heat dissipation of a standard office PC (60W idle, 120Watts fully loaded) would raise the ambient temperature by ~1°C (assumed 50% efficiency of the air-con). The 120 Watts compare to the 100W approximate basal metabolic rate + 20-40W brain activity of the human body – so the user of the PC will raise the ambient temperature for another 1°C. The raise in room or ambient temperature is minimal since the thermal balance in this example is dominated by the heat flow through the building structure. Massively higher impact on the room temperature is caused by cooking activities (in the n*kW range). This discussion is not finalized. We explicitly ask for support either by pro or contra arguments and references.

C. Energy profiling of applications

Applications will be investigated with regards to their energy profile. Different applications and codes consume more or less CPU at any given time, resulting in different energy consumption per time interval; they behave differently in raising machine and ambient temperature. According to our findings within the DEGISCO available pool of applications (The inherited EDGeS pool of applications: [14]), these are classified accordingly with a heat index as +, ++ and +++ for example. We refrain from using “green”, “orange”, and “red” at this point: The +++ index marks an application that makes maximum use of a given machine, is raising its temperature, but finishes the computation quickly. This behavior may total in less “energy consumed”/computation than the application which creates less heat/time. Still heat/time is an important parameter from a green operations point of view. As PC owners can select the project and by this the application they want to contribute to, they can take into account their specific knowledge of local operations conditions, primarily how much additional heat they can accept.

Limitations and implementation risk mitigation: If the energy profiles of the applications within DEGISCO and partner projects happen to be too similar, the demonstrated impact of the approach become less obvious. Means to adjust application energy consumption per time interval have been discussed and could be used for demonstration.

D. CPU Speed Steps

A similar effect could be achieved by exploiting processor speed steps, avoiding additional preparation work on the application side. Current processors provide multiple steps (8-16) for CPU speed, thus controlling energy consumption. Gruber and Keller discuss the use of “SpeedStep” among other methods in order to use the minimal CPU frequency to run an application at full memory bandwidth [15]. Different from the application, the OS and tools installed at the PC are not under control of the Desktop-Grid operator but the contributor, rendering the applicability of methods like “SpeedStep” questionable due to lack of resource control. Whether reduced preparation work on the application level balances the lack of control on the resource level needs to be found out. DEGISCO will call for volunteers supporting the installation of CPU speed step tools and monitoring and report about the benefits and difficulties.

E. Exploitation of natural ambient conditions

A completely independent green strategy we are going to investigate exploits a DEGISCO specific advantage: the aggregation of partners from various different geographies allows benefiting from differences in regional weather situations in order to save energy. Workload indexed as "+++" may systematically be offered to contributors located in low temperature areas while those in sunny summer weather will be offered to contribute for "+" workload.

Different locations yield different climates

- Kazakhstan, Amaty: http://worldweather.wmo.int/070/c00152.htm
- Russia, Moscow: http://worldweather.wmo.int/107/c00206.htm
- Hungary, Budapest: http://worldweather.wmo.int/017/c00060.htm
- Denmark, Copenhagen: http://worldweather.wmo.int/173/c00190.htm
- Spain, Zaragoza: http://worldweather.wmo.int/083/c01240.htm

Recruit regions with opposite weather conditions! Only "+++"-workload today? Sorry Zaragoza, Copenhagen is cooler! DEGISCO partners from Kazakhstan, Russia, and Spain confirmed the weather conditions reported on "world-weather" as already averaged – the peak temperatures exceed both into heights and lows significantly. Again, as DEGISCO just started so is this discussion – and your contributions are welcome!

F. Time-of-day or weather dependent energy tariffs

The value of electrical energy is usually changing according to the conditions of generation as well as by changing consumption. Accordingly, the tariffs for electricity are changing.

While in Germany electricity prices are high during lunch time, in Kazakhstan the energy prices go up in the evening – in both cases dependent on consumption.

Energy prices at the Spot markets vary depending on excess production capacity. Since wind energy can deliver significant amounts of energy, these spot market prices can even turn negative [12].

To improve the energy cost situation and to take advantage of excess Green electricity, advice could be given to contributors how to configure their Desktop-Grid-Clients to prefer workload during low tariff times.

IV. Desktop-Grid – the anarchistic virtual data centre

A major difference between a data centre situation and volunteer based Desktop-Grids is the almost complete lack of central control over the compute resources. Further, Desktop-Grid applications are executed as user with limited permissions (no root rights). Accordingly, installation of support tools, in our case for temperature and energy consumption measurement, is not possible without active voluntary contribution by the resource owner, installing tools with administrator (aka "root") rights. This cannot be done as regular Grid job: different from service Grids, Desktop-Grids implement highest security standards also on the execution side. Applications that are downloaded and executed on the contributed Desktop-Grid client are security validated and, dependent on the Desktop-Grid technology used, even rewritten to execute exactly that computation as described – and nothing more. Any activity beyond the sandbox, e.g. accessing local HW devices like sensors, is off limits for Desktop-Grid jobs. When DEGISCO is looking to gather detailed temperature and energy consumption data we will ask for volunteers to download and install tools and allow the upload of resulting metric data.

V. Difficulties of temperature measurements

In order to apply the Green Methodologies described, it is necessary to adequately understand the ambient conditions of the PC, especially the ambient temperature. Least effort would be the use of PC's built in temperature sensors – but there are difficulties to overcome. The temperature sensors built into PCs and laptops are optimized to support energy management of the PC and its components – not to provide ambient conditions, the kind of information we need. Mainly the position of the sensor determines what is measured. On-die temperature sensors may reflect the CPU internal temperature quite precisely while "system" temperature sensors are placed "somewhere" on the mainboard – delivering temperature measurements that cannot be interpreted meaningful without precise and detailed knowledge of the individual board. Although this seems doable in a lab situation, it is completely beyond scope and capabilities in case of real world deployed Desktop-Grids. The situation is not very much better in regular data centres: depending on the placement of the temperature sensor in the rack, a hot spot will be detected or not. A detailed temperature measurement at several positions in the rack is not commonly found. For safety reasons, the single temperature sensor is placed to detect the (known or anticipated) hot spot, caused by poor local airflow – delivering information misleading with regards to average, typical, or total (=full rack) energy consumption. Further complication is deriving from the application of temperature aware fan speed controls embedded in the systems. Originally developed for Desktops in order to keep their operations noise level convenient for living room conditions, meanwhile regular servers are controlling their fan speeds to provide exactly that amount of cooling needed to keep board temperature within the targeted operations range while enhancing the lifetime of the fans. Side note: The formerly already mentioned "Code of Conduct on Datacenters" explicitly requests control of fan speeds also on the data centre level. The result for our aim to understand the ambient conditions of a machine by reading its temperature sensors gets complicated by these features.

Still the information retrieved may well be sufficient for our aims:

1) Understand values delivered by PC internal temperature sensors as non-linear non-calibrated relative information on machine cooling effectiveness.
2) For ambient temperature use meteorological data by independent sources.

3) To calibrate and QA the methods, call for participation by contributors in a temperature measurement campaign.

Even qualitative temperature information is suitable to distinguish condition “too hot for workload” from “cool and ready to work”. To verify our understanding on ambient conditions, we started working with Mathias Dalheimer, Fraunhofer Institute ITWM, Kaiserslautern. Mathias developed a simple and low cost temperature sensor [16] that can be connected to the desktop or laptop (USB) and delivers proper ambient metrics. This temperature sensor may be offered to Desktop-Grid volunteers by mail-order, requesting the commitment to provide temperature measurement data for automated upload.

The sensor implements an USB1.1-device. The whole USB stack is implemented in software and does not rely on dedicated hardware, keeping the component count low. Digital temperature sensors provide high accuracy measurements. Since the sensors are attached via the Onewire bus, a single USB dongle can query up to eight different temperature sensors which can be placed at different locations around the PC. The software stack is available for Windows, Mac OS and Linux and allows users to query the temperature sensors. Both the hardware and software components are open source and can be modified to accommodate additional requirements.

Currently sponsors are looked for. If the device is produced in a small batch, the cost of an individual device should be around 10 Euro.

VI. Call for contributions: join the Desktop-Grid-Federation

Due to the specific character of Desktop-Grids as voluntary effort, the personal contribution of individuals is of prime importance and very welcomed. In order to improve Desktop-Grid service for science, the dear reader may consider joining the International Desktop-Grid-Federation, described in detail below. Especially welcome are your contributions to Green optimization of Desktop-Grids. To advance the Green optimization beyond the described basic methods, we need support by individuals, institutions, and projects. Contact: mail to green@desktopgridfederation.eu

Looking forward to hear from you!

Appendix: The International Desktop-Grid-Federation

Desktop Grids, Desktop Clouds, allow to employ otherwise idle computing time of Desktop computers for large computational programs.

Desktop Grids can be used inside an organization, or they can collect computing time from volunteers all over the country, or even all over the world. However, operating a desktop grid and developing programs for desktop grids poses specific challenges. That is why the International Desktop Grid Federation is formed.

A. Who should become member?

Organizations and persons can become member of the International Desktop Grid Federation. When should you consider joining? Your organization is operating a Desktop Grid and would like to share experiences with operators, get support and trainings Your organization has a lot of Desktop computers, and would like to make better use of them You want to develop computation programs for Desktop Grids Your organization wants to install a volunteer Desktop Grid, and wants to know how to attract volunteers You are interested in further developing Desktop Grid and Desktop Cloud technology You would like to integrate a Desktop Grid with other Service Grids (for instance based on gLite, KnowArc or Unicore). Your organization has large computational needs, and wants to know whether Desktop Grids could be used. You care about the environment and want to know how Desktop Grid can contribute to a Green ICT infrastructure with less power consumption.

B. Services offered by the International Desktop Grid Federation

The Federation offers several services for its member: Meetings, workshops and conferences: were you can meet and discuss with other members Training sessions and tutorials. Technical support in operation Desktop Grids Technical support in connecting Desktop Grids with other Grids and Clouds Web site and information centre. Material will be available in several languages.

C. Supporting projects

The services offered by the International Desktop Grid Federation are supported by two European projects: EDGI and DEGISCO. EDGI focuses on Europe and on integrating clouds, DEGISCO has an international approach.

D. How to become member

All services are available to all members. So become a member today. We have several options, for companies, research organizations, and persons. More information: http://desktopgridfederation.org

References

[1] Folding@Home. Client statistics by OS. [Online] [Cited: 21 04 2010.] http://www.boincstats.com/stats/project_graph.php?pr=bo.

[2] BOINC. Credit overview. [Online] [Cited: 21 04 2010.] http://www.boincstats.com/stats/project_graph.php?pr=bo..

[3] EDGeS. EDGeS Bridge. *Enabling Desktop Grids for e-Science.* [Online] 01 01 2010. http://www.edges-grid.eu:8080/web/edges/57.

[4] Institute for Energy, European Commission Joint Research Centre. EU Code of Conduct for Data Centres. *http://re.jrc.ec.europa.eu/energyefficiency/html/standby_initiative_data_centers.htm.* [Online] http://re.jrc.ec.europa.eu/energyefficiency/pdf/CoC%20DC%20new%20rep%20form%20and%20guidelines/Best%20Practices%20v2.0.0%20-%20Release.pdf.

[5] Schott, Bernhard. OGF25 Catania. *Energy optimization of existing datacenters.* [Online] 05 03 2009. http://www.ogf.org/OGF25/materials/1654/Energy+Optimization+of+Existing+Datacenters+-+Bernhard+Schott+-+Platform.pdf.

[6] Lippert, Thomas. Contributions to HPC 2010 Cetraro. *High Performance Computing, GRIDS and clouds, June 21 – 25, 2010, Cetraro, Italy.* [Online] 21-25 06 2010. http://www.hpcc.unical.it/hpc2010/ctrbs/lippert.pdf.

[7] EDGeS. EDGeS presentations. *Downloads.* [Online] 01 01 2010. http://www.edges-grid.eu:8080/web/edges/7?p_p_id=

110_INSTANCE_a1KF&p_p_action=0&p_p_state=normal& p_p_mode=view&p_p_col_id=column-2&p_p_col_count =1&_110_INSTANCE_a1KF_struts_action= %2Fdocument_library_display %2Fview&_110_INSTANCE_a1KF_folderId=32559.

[8] Federation, International Desktop Grid. International Desktop Grid Federation. [Online] 01 06 2010. http://desktopgrid federation.org/.

[9] Comission, European. GERMANY – Energy Mix Fact Sheet. [Online] 01 01 2007. http://ec.europa.eu/energy/ energy_policy/doc/factsheets/mix/mix_de_en.pdf.

[10] [Commission, European. FRANCE – Energy Mix Fact Sheet. [Online] 01 01 2007. http://ec.europa.eu/energy/ energy_policy/doc/factsheets/mix/mix_fr_en.pdf.

[11] Comission, European. DENMARK – Energy Mix Fact Sheet. [Online] http://ec.europa.eu/energy/energy_policy/doc/ factsheets/mix/mix_dk_en.pdf.

[12] Spot, Nord Pool. Nord Pool Spot implements negative price floor in Elspot from October 2009. *Press release.* [Online] 04 02 2009. http://www.nordpoolspot.com/Market_Information/ Exchange-information/No162009-Nord-Pool-Spot-implements-negative-price-floor-in-Elspot-from-October-2009-/.

[13] PCWelt. CPU-Leistungsexplosion Intel Core i7 Prozessor. *Stromverbrauch und Energieeffizienz.* [Online] 03 11 2008. http://www.pcwelt.de/start/computer/prozessor/tests/185273/intel_core_i7_prozessor/index3.html.

[14] EDGeS. Applications available on the EDGeS infrastructure . *Enabling Desktop Grids for e-Science.* [Online] 01 01 2010. http://www.edges-grid.eu:8080/web/edges/49.

[15] Gruber, Ralf and Keller, Vincent. *HPC @ Green IT.* Berlin Heidelberg : Springer-Verlag, 2010. p. 184ff. DOI 10.1007/978-3-642-01789-6_1.

[16] Dalheimer, Mathias. USBTemp: Continuous Temperature Monitoring. [Online] 03 01 2009. http://gonium.net/md/2009/ 01/03/usbtemp-continuous-temperature-monitoring/.

[17] BOINC. Open-Source Software für Volunteer Computing und Grid Computing. [Online] 06 09 2010. http://boinc.berkeley. edu/

[18] GIMPS. Great Internet Mersenne Prime Search. [Online] 06 09 2010. http://mersenne.org/various/history.php

[19] XtremWeb. XtremWeb: the Open Source Platform for Desktop Grids. [Online] 06 09 2010. http://www.xtremweb. net/

[20] DEGISCO. Degisco project website. [Online] 06 09 2010. http://degisco.eu/introduction

[21]

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 677–684

ISBN 978-83-60810-27-9
ISSN 1896-7094

Resource Fabrics: The Next Level of Grids and Clouds

Lutz Schubert, Matthias Assel, Stefan Wesner
HLRS – University of Stuttgart, Dpt. of Intelligent Service
Infrastructures and Dpt. of Applications & Visualisation,
Nobelstr. 19, D-70569 Stuttgart, Germany
Email: {schubert, assel, wesner}@hlrs.de

***Abstract*—With the growing amount of computational resources not only locally (multi-core), but also across the web, utility computing (aka Clouds and Grids) becomes more and more interesting as a means to outsource management and services. So far, these machines still act like external resources that have to be explicitly selected, integrated, accessed etc. - much like the concept of "Virtual Organisation" prescribes. This paper will describe how the development of dealing with increased scale and heterogeneity of future systems will implicitly open the door for new ways if integrating and using remote resources through a kind of web-based "fabric".**

I. Introduction

OVER the last few years distributed computing has gained in relevance, not only as a concept, but – more importantly – as a commercial reality: through "Clouds" and multicore processors which make concurrent compute units available for the average user [1, 2]. Notably, usage of these two environments differs significantly: Cloud systems are essentially server-like machines available over the internet and accessed accordingly (via e.g. remote desktop or SSH); as opposed to that the capabilities of multicore machines are locally available – implicitly there are no specific means needed to access the resources.

What is more, cloud systems typically deal with scaling a specific service or application multiple times according to access and availability needs, i.e. perform scale by replication. As opposed to this, multicore systems, just like high performance computers deal with scaling a single application "vertically" across the resources in the form of instantiating multiple processes that *together* form the application logic, as opposed to individually [3]. It should be noted in this context though that desktop systems and high performance computing systems differ in this respect: whilst the latter typically only deal with execution of one single process or thread per core at a time, desktop usage typically implies concurrent execution of multiple applications in a time-sharing manner.

In both the cloud and the multicore case we talk of "distributed systems" in the sense that the system on which the service / application is being executed consists of multiple resources connected via a communication and / or messaging link. From this perspective, we can specifically note that essentially clouds and multicore systems just provide different capabilities on essentially the same environment. Section II will elaborate how a common "denominator" of such a system could look like.

By exploiting the commonalities, rather than focusing on the differences, it would in particular become possible to exploit remote resources as if local, thus truly realising the grid's original concept [4, 5]. For example, an application could be replicated beyond the restriction of the local system, and new applications executed remotely without interfering with any local executions, thus competing over resources. What is more, even simple laptop systems would be enabled to execute demanding applications in the same fashion as on a home or office PC [6]. Whilst this is not a new idea as such (see [4]), its realisation has many impacts not only on a middleware, but more importantly on operating system and programming model. Sections III and IV of this paper will highlight in particular to how such a "resource fabric" could be realised and to which specific technological issues apply, respectively.

By overcoming these obstacles, new systems and implicitly new types of applications would become possible that allow following the worldwide trend of internet integration, dynamic outsourcing etc. in short, the future internet. Section V will describe these application areas in more detail.

As will be shown, it is not yet possible to overcome all technical issues easily, in particular since the "natural" technological development (i.e. the industrial provisioning) follows the laws of stepwise development, rather than disruptive (r)evolution. We will conclude this paper with an analysis of the main outstanding issues in section V.

II. The "New" Von Neumann Architecture

"Distributed systems" in their widest sense , i.e. including multi-core processors, clusters, clouds and grids all have in common that they integrate compute units over a communication link. The main difference thereby being the specifics of the linkage: whilst communication between cores in a multi-core system has a very low latency; cloud and grid systems in particular use the intra- / internet for communication, meaning high latency but considerably large bandwidth; and finally, HPC systems integrate different levels of linkage, ranging from multi-core interconnects to fast, broadband networks (100GB Ethernet, Infiniband).

This work supported by the S(o)OS project under FP7 grant agreement no. 248465 – http://www.soos-project.eu

In principal, latency can be compensated by bandwidth, i.e. if the delay is l and the bandwidth b, the effective data d_{eff} communicated in a set of messages over a time-frame t is

$$d_{eff}(t) = t * \left(\frac{b}{l}\right) * seconds$$

In other words, latency is anti proportional to bandwidth: if latency is high, bandwidth should be large too, respectively a low latency can compensate a small bandwidth, in order to reach the same effective data throughput.

However, an important factor has been subsumed in this calculation, namely the numbers of communications within that time-frame. Obviously latency impacts only per individual invocation, respectively message exchange (actually leading to half the throughput per communication side), meaning that the full throughput depends on the number of invocations, which is accordingly high if the latency is small, and low with a large latency.

$$Number\ of\ invocations = \frac{t}{l}$$

For non interactive systems this does not play a major role, the delays in communication are not noticeable as such – however, in applications that directly interact with the user (such as a word processor), any delay occurring between user input and system reaction beyond 0.1 seconds leads to the impression of a "non-reactive" system, and beyond 1 second, it will even disrupt the user's flow of thought [7].

Browsers in particular take a position somewhere between interactive and non-interactive and users show a slightly higher tolerance towards waiting time than in local applications (4 seconds for maintaining the flow of thoughts according to [8]). Implicitly, offering applications *via* the web typically leaves an impression of being slow and unresponsive.

Latency is always a source for problems, when task execution is *synchronously* dependent on communication, i.e. when the querying task is blocked whilst it is waiting for a response. This affects in particular parallelised applications where the individual processes need to synchronise data.

The average developer however is not aware of the connection details – not only are they difficult to acquire, but even more difficult to represent in the program in some form. The general tendency is therefore to make use of asynchronous messaging in the web domain (clouds, grids) and synchronous messaging in compute clusters.

However, embarrassingly parallel applications scale well in any environment and by removing the interactive from the processing part, web based systems can even be exploited for demanding user applications. To achieve this, we do no longer need to distinguish between different resource types, but between their connectivity. From this perspective, the different notions of distributed systems fall together as a complex system consisting of computation resources connected over a communication link. Modern systems are therefore no longer building up a strict von Neumann architecture, but a modular system that connects any amount of von Neumann *like* units.

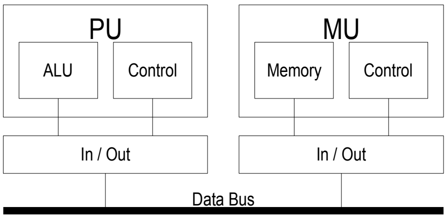


Figure 1: A modified von Neumann architecture that depicts any form of distributed machine as a type of linked compute units.

In particular from a programming perspective, we can therefore derive a modified von Neumann architecture as depicted in Figure 1: in particular the I/O moves closer to the PU (processing unit) thus allowing for connectivity between multiple PUs without an explicit single central instance. We also distinguish between processing and memory units (MU), which are connected through a dedicated I/O. It must be noted that current multi-core systems do not make use of an explicit I/O unit for core-2-core communication – whilst this is expected to change, the according I/O will most likely differ from the one connecting outside the processor. It must also be mentioned that cache access is not linked to an I/O as such, but to a memory controller – again, long-term developments (e.g. ring over caches) will impact here. All this however, has no impact on the programming layer.

In this view, clusters, multi-core processors and cloud or grid based systems are essentially identical. So far this model is essentially realised through according means on the middleware level (with the highest level of abstraction being represented through distributed workflows).

A. Data management in the "new" model

The massive distribution of data over multiple machines (i.e., storage points) as provided by this new architecture fundamentally changes current data management concepts, too, in particular data generation, exchange and storage. These phases are usually extremely time-consuming, and thus require new mechanisms, strategies and tools that manage the movement of particular data sets (similar to [9]) automatically across a storage hierarchy.

Similar to the code (see below), data must therefore be managed in a more intelligent fashion, according to the points of access and the degree of synchronisation, respectively coherency across usage points (see also [10]). As such, data that is frequently used shall be moved to highly parallel dynamic storage (e.g. parallel file systems like Oracle's Lustre (http://www.lustre.org/) or virtual distributed file systems like GFS [11] or a combination of both), while archived data shall reside in "passive" storage devices (e.g. low-cost storage devices or robotic tape libraries). To allow for this separation, a fundamental requirement is the ad-hoc allocation, use and release of particular storage space. Hence, algorithms shall automatically request necessary space but also track and delete unused data from these dynamic storages, so as to minimise storage costs and increase throughput.

Similarly, replication of data represents a key issue to increase data locality and availability. Management of replicas has to carefully consider lifetime issues to remove outdated pieces immediately. New replicas should be dynamically created, distributed and / or destroyed based on users' and applications' needs but also according to technical requirements [12]. In shared environments, i.e. where multiple access points require the same data, serious consistency issues across nodes have to be addressed [13].

Of particular relevance is also that data source and applications are not necessarily directly coupled. Collections of data sets shall be organised as hierarchical directories. Such abstraction will essentially change the way the I/O is expressed by applications and will involve data exchange and storage management in a form that maps data sets into physical devices without affecting the application's behaviour.

III. Weaving Resource Fabrics

The classical approach towards dealing with distributed systems consists in providing a form of middleware that translates function invocation, instantiation etc. into a set of remote procedure calls or web service calls. The actual details depend not only on the realisation but also on the domain applied to:

In multicore systems, the actual transaction logic is encapsulated and realised via the hardware; supercomputer clusters employ some form of dedicated communication programing model (with either explicit (e.g. MPI) or implicit (e.g. PGAS) messaging) that is translated into ports and sockets at compile time and the grid / cloud support typically builds on http protocols that are typically translated into ports and sockets at run-time.

In other words, the actual type of communication bridge is transparent to the user, though he will still have to use it in different ways, depending on use case and environment. Though there are ways of controlling and configuring the communication link, there is little possibility to exploit this dynamically for maintaining distributed code – in other words, parallel programming models that use synchronisation for control base on the assumption that the underlying infrastructure is effectively homogeneous.

Considering however that neither the resource infrastructure, nor the program is effectively homogeneous: typical applications, even in the strong scaling based HPC domain, the actual algorithm consists of both parallel and sequential segments that interchange during execution. What is more, e.g. in typical (unstructured) grids algorithms (such as blood--flow simulation etc.), the communication relationship between the parallel processes is not symmetrical, meaning that the code would benefit from a distribution on the infrastructure aligning the connectivity between compute units with the neighbourhood model of the code.

The classical means to developing parallel applications consists in either segmenting the work or the data by identifying natural partitions to either of those [14]. Whilst this is the most common approach, it suffers from the drawback that it a) requires good knowledge about the program and the concurrency in work and data, necessitating additional development work; b) the partitioning may be too small or to big to be efficient, in particular if the communication exchange between segments is not considered properly; c) heterogeneity and structure of the underlying system are mostly unused – what is more, if the according effort is undertaken to adapt code to the specific system, it will become less portable; and finally d) not all kind of segments can be identified this way. Heterogeneity and architecture of the system are typically regarded as obstacles, but program execution can actually benefit from this structure by reflecting the "natural" behaviour of an application.

We can identify the following key aspects of any algorithm:

1. Concurrency – some functions are executed without explicitly sharing data or resources. In this case these segments can be executed in parallel.
2. Parallelism – similar to concurrency, some functions operate on a common data set, but the actions they perform are not directly dependent on one another. This is typical for loops over large grids ("loop unrollment").
3. Interactivity – in particular in desktop applications, the interface towards the user can be easily defined by the connectivity to external input resources.
4. Background Tasks – are tasks waiting for specific events to execute and with little relationship to the main execution (regarding data dependency).

Similarly, the communication needs are not the same between all these parts, though there is no general statement about the relationships possible: concurrency and parallelism do not necessarily imply high connectivity (low latency), as e.g. embarrassingly parallel applications show; on the other hands, events processed in the background may require immediate response and interactivity does not mean that all processing on the input data has to be executed immediately (cf. word count and spell-checking in modern word editors: even though they react "immediately" to any input, the delay until the actual results are available is of no concern to the user and hence typically not even noticeable. Again, keep in mind that if the result is in some form relevant (in the sense of often checked by the user), the time frame for maintaining a flow of work is 1 second [7].

A. The structure of applications

The structure of an application can therefore be used as an indicator for its distributability (and, to a degree, parallelisability). The (runtime) behaviour provides additional information about the actual connectivity between the individual segments and thus its requirements towards the communication model, i.e. the relationship of latency versus bandwidth.

Implicitly, runtime *behaviour* effectively provides more information about the potential code distribution than the programmer can currently encode in the source code. This is simply due to the fact that this is not in-line with our current way of writing programs and is implicitly not directly supported by programming models. The foundation is however laid out by integration of remote processes (web services) and dedicated synchronisation points in parallel processes – this does not always reflect the best distribution though, as the according invocations are mainly functionality- rather than communication-driven.

A way of identifying and exploiting the "behavioural" structure of the application for distribution purposes consists in run-time analysis and annotation of the code and data memory to produce a form of dependency graph (cf. Figure 2) which depicts the invocations of memory locations (code) and read / write operations on data.

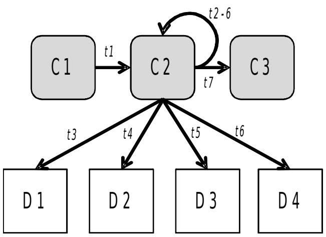


Figure 2: A simple code (C) and data (D) dependency graph of a sequential application with a loop over an array (C2). The graph denotes a sequence of actions with t_x representing the x^{th} transition, respectively access action.

In order to acquire the according code and data blocks, the basic starting point consists in identifying jumps and non-consecutive data accesses. Figure 2 exemplifies how the graph information can be used to identify an unrollable loop (C2) that consecutively accesses unrelated memory blocks to produce a result. In this simplified case there is no data dependency between C1 and C2, or even between C2 and C3, which means that the unrolled C2 blocks do not even have to be synchronised. An example of such a loop would be

C2: for (int i=0; i<4; i++) a[i]=0;

As opposed to this, Figure 3 depicts the example of a loop that can not be unrolled due to dependency issues in C2. This loop could look like

*C2: for (int i=1; i<3; i++) a[i]=a[i-1]*2;*

Notably, the examples given are not sensible for parallelisation, as the actual work load per iteration is too small to compensate the overhead for spreading out, communication etc. All analysis so far is based on the simple fact of data relationship and dependency – not unlike to the approach pursued in StarSS [15]. Accordingly, even though a pattern based approach like the one presented above does principally provide information about the distribution and parallelisabilty, it does not actually increase the execution performance.

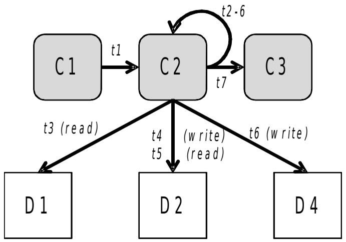


Figure 3: Example of a loop that is not unrollable and its representation in a dependency graph.

In order to not only identify principle points of distribution and parallelisation, but to also make code execution more efficient, so as to exploit the specific benefits of a parallel infrastructure, further information about the code behaviour is needed – in particular the “strength” of code / data relationships, and the size of the segment. Note that timing can be derived through more detailed sequential information (i.e. in Figure 2 or Figure 3 by storing actual timestamps in t_n). Strength of relationship is thereby proportional to amount of invocations divided by the full execution time.

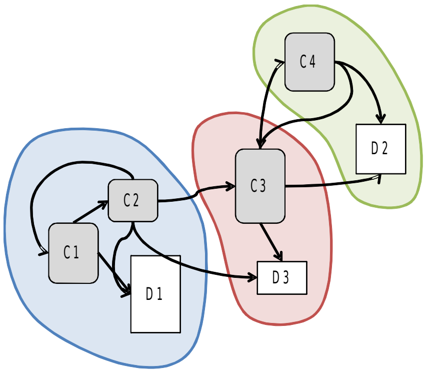


Figure 4: A dependency graph with implicit relationship strength (length of the vector) and size information (size of the block). The three areas depict potential areas of segmentation.

With this information, we can derive a graph (Figure 4) where the the strength of the relationship (e.g. C2 calls C3 less often than C1 calling C2) and the size of the underlying code / data is encoded as weights of vertices and edges. For simplicity reasons we left out timing information in the figure and concentrated on a very simple code structure (without loops or similar) – according data will be provided in future publications.

The dependency information in this graph, in combination with the size information can be used to extract different segments in the form of subgraphs according to nearness (connection strength) and combined size. Or to put it in computational terms again: according to the number of memory accesses with fewer accesses implying a potentially good cutting point. Each segment reflects the code to be executed on one compute unit (core, node etc.) and connections across segments need to be realised through a cross-process communication link. Information about the bandwidth and latency of the system's specific communication links can thereby serve as an indicator for cutting point identification, if it is respected that access across segments is effectively identical to data passing – accordingly, the temporal sequence and dependency of access constrains the segments' independency and hence concurrency. In order to not exceed this paper's limitation, we will not elaborate the segmentation algorithm in detail here.

It is obvious that one of the major problems for an efficient segmentation consists in the right data gathering granularity: whilst too fine data will cause memory and algorithm to go over bounds, too coarse information will lead to too strong relationships and too large segments, so that the effective gain through distribution is counterweighed by the overhead for communication and memory swaps.

B. Life-cycle of applications in a resource fabric

As noted, the major part of the information about the code is acquired at actual run-time – implicitly, the distribution information may change during execution, leading to potential instabilities. In this section we will examine the full lifecycle of an application executed in such an environment and implicitly the steps involved to better exploit a scalable (and potentially dynamic) environment can be captue:

1. Analysis of the application behaviour ("Analyse"): In an initial step (first time the application is executed), there is little to no dependency information available, unless an according programming extension (such as StarSS) was applied. This means that initially the application is executed locally in a virtual memory environment that logs the run-time behaviour and hence the dependencies between code partitions. Implicitly first time invocation is effectively identical to sequential execution – even though user provided parallelisation (if any) can still be exploited.

2. Identification of appropriate resources ("Match"): The dependencies in the application graph reflect the communication needs between segments and thus implicitly indicate the required infrastructure architecture, including type and layout of interconnects. But also specific core types can be exploited to a degree – e.g. Figure 2 shows clear vectorizable behaviour and other microarchitecture specific patterns can be identified, which would exceed the given space though.

3. Distribution and adaptation of code and data ("Distribute"): When appropriate resources could be identified, the code segments can be distributed across the infrastructure accordingly. In the simplet case, all partitions will be uploaded prior to actual execution, in which case no additional data has to be distributed at runtime (besides for the data transported during communication). However, in principle, it is possible to distribute the segments in their order of invocation, though this runs the risk of potential delays.

4. Execution and run-time analysis ("Execute"): Actual execution is principally identical to any distributed program execution with explicit communication between process instances. However, as opposed to an explicitly developed parallel program, the source code in this model has not been altered and the communication points and tasks are unknown to the algorithm itself. Accordingly, the infrastructure has to take care of communicating the right data at the appropriate time. This is principally identical to the PGAS approach (http://pgas.org/), which provides a virtual shared memory and deals with the communication necessary to enable remote data access. Effectively the system for enabling distributed execution in the model proposed here must enact the same tasks directly on the (virtual) memory environment of the operating system, rather than on the programming level.

During execution, the system may continue analysing behaviour (cf. step 1 "Analyse") in order to further improve the segmentation information and granularity. As program behaviour changes according to code dependency, the main problem to avoid in this phase consists in ensuring and maintaining an *efficient* stability of the distribution. Monitoring and segmentation must therefore not only consider the dependencies, but also higher-level parameters that implicitly define the stability of a given segmentation. Though there is some relationship to SLA based monitoring, these parameters should not be confused with common quality metrics.

5. Information storing (Store): Behaviour and distribution information should be stored after execution in order to be retrieved in the next iteration, thus allowing to skip step 1 and thus improving the process.

The main issues in the life-cycle consist obviously data exchange and synchronisation of the segments which are treated as distributed processes. If data and execution are not carefully aligned, inconsistency may lead to serious crashes, deadlocks, or serious efficiency issues due to overhead. It is therefore not only relevant, from where data is accessed, but also when and in which order. Ideally, data is being transported in the "background" whilst the segment is mostly inactive. Due to lack of space, we will not elaborate these issues here – suffice to say that by treating the segments completely isolated and swapping all memory during execution pass-over, coherency and consistency is ensured.

IV. A Middleware for Resource Fabrics

In order to achieve this goal, the execution environment must accordingly provide some capabilities to capture memory access and intervene with code progression so as to pass the execution point at the boundary of the individual segments. Effectively this means that the system provides a virtual environment in which to host and execute the code – a full virtual system however would impact on execution performance. Whilst this may be acceptable for some type of applications where simplicity of development ranks higher than performance, in particular in scalable environments, efficiency typically ranks higher.

The system does not necessarily need to provide a *full* virtual machine, but in particular a virtual memory environment and a set of interfaces to access (shared) resources – in other words, an operating system. Since effectively in all execution parts memory access needs to be controlled, a centralised operating system approach would create a serious non-scalable bottleneck and additional delays due to message creation and communication would turn out a major performance stopper. This relates to the major reasons why monolithic, centralised operating systems show bad horizontal scaling capabilities [16].

In order to achieve better scalability the individual compute units hence need some local support to reduce overhead and enable virtual memory management in a form that can also identify and handle segment passover. The S(o)OS project (http://www.soos-project.eu) funded by the European Commission investigates into new operating system architectures that can deal with exactly this type of scenarios. The approach thereby essentially bases on a concept of distributed microkernel instances that fit into the local cache of a compute unit (processor core) without obstructing it, i.e. leaving enough space for code and data stack.

We can essentially distinguish between two types of OS instances in a resource fabric: firstly, the main instance that deals with initiating execution and distribution, as well as scheduling the application as a whole. Secondly, the local instances that effectively only deal with communication and virtual memory management. In effect, all instances could have the same capabilities [16], which would make them larger and more complex to adapt to specific environments though. In S(o)OS the principle is extended with on-the-fly

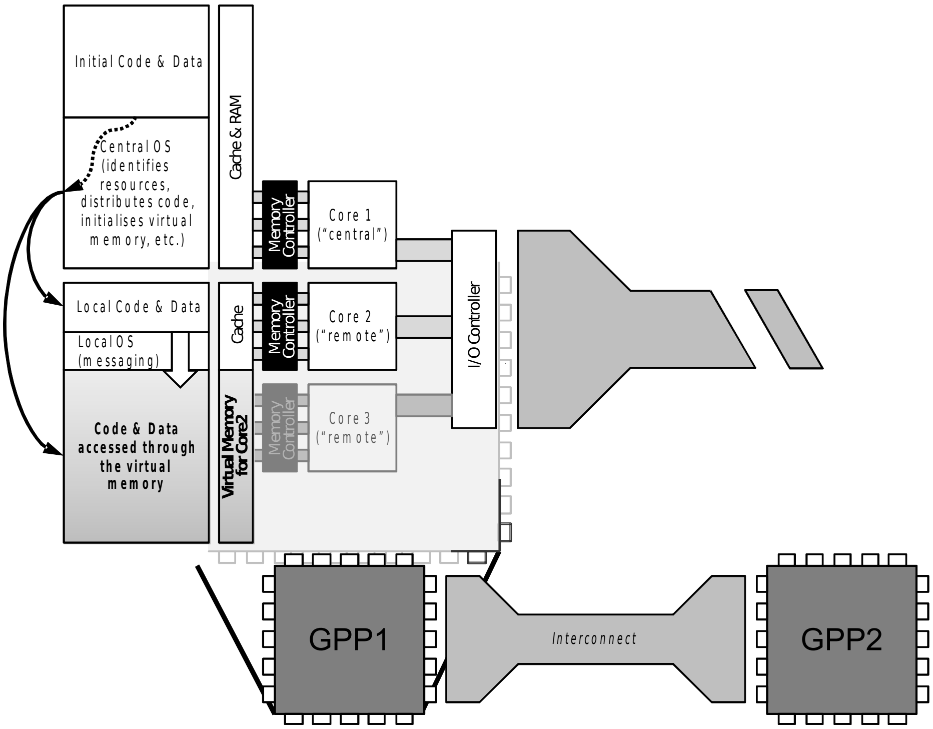


Figure 5: A distributed micro operating system and its relation to the individual compute units in the system. "Remote" cores can be hosted within the same processor or principally a remote machine.

adaptation of local OS instances according to application requirements and resource capabilities – this way, the kernel can even support adaptation to resource specifics without the central instance having to cater for that [17].

In Figure 5 we depict the principle of such an operating system with respect to the relationship between cores (or compute units, comes to that): typically one selected unit will take over the initial responsibility for code and data, i.e. loading it from storage, analysing it (respectively retrieving the annotations) and distributing it accordingly. The segmentation information (i.e. the memory structure derived from the dependency graph) will be passed with the code segments, allowing the local instance to build up the local virtual memory and implant system invocations at the appropriate time (during segment pass over).

A. Distributed execution

Pure distributed (as opposed to parallel) execution is effectively similar to context switching during time-sharing, only that the "new" state is not uploaded from local memory, but provided over the communication link between the initiating unit and the one taking over. However, parts of the state (code and base data) can principally already be available at the destination site due to predistribution according to the dependency graph (cf. above).

Similar to context switching, the application itself does not have a dedicated point at which to perform the switch (or even enable it), though with additional programming effort context switching can be avoided (thus leading to single-task execution, up to a point). As opposed to multi-tasking operating systems, however, a S(o)OS like system must initiate the context switch (execution pass-over) at a dedicated point, according to the segmentation analysis (cf. Figure 4).

This point is effectively the end of the local segment and can thus be initiated by a simple *jmp* (jump) command into the virtual memory address space of the new segment and can thus simply be appended to the partition. Just like with any access to remote data (be that due to branching or data access), the operating system has to intercept the request prior to its execution, similar to classical virtual memory management. The MMU (Memory Management Unit) of modern processors supports this task already, but would need to be extended to also cater for the specific needs of a distributed (cross-unit) memory virtualisation.

B. Access across

Modern systems supporting a fully distributed environment will hence not only need to check for *local* accesses to virtual memory addresses, but also for remote ones. Whilst a classical local cache miss would lead to a page swap, in the environment promoted here, a cache miss could indicate that the actual endpoint is hosted remotely. Accordingly, the destination address needs to be matched against the dependency graph based distribution to identify the real endpoint.

In principle, this endpoint may not be known to the local instance (in particular true for dynamic environments) in which case the memory address resolution request needs to be forwarded to other OS instances – though this could be the central one (cf. Figure 5), a broadcast to available instances may be more effective due to concurrency issues.

Far more important, the operating system needs to identify whether the remote access is a data access request (*mov, in, out* etc.) or a code branch (*jmp, call*). In the latter case, local execution will be halted and the overall execution point be passed over to the remote instance (cf. above). The destination therefore must be able to interpret the entry point like a local jump (out of virtual space).

C. Data Maintenance

As opposed to passing over execution, data access across the units means that the remote information needs to become locally available in some form. The OS instance has principally 3 options to deal with data access:

1. preemptive distribution: provide the data to all requestors, before they actually try to access it
2. context switch: pass all status data when passing over the execution point to a remote instance (does not apply to parallel processes)
3. on demand: make data available and accessible the moment the remote process requests it

The actual decision will not only depend on time of access (in particular for parallel processes), but also on size of the data and on outstanding tasks of the processing segment. For example, if the data will only be ready by the end of the segment's processing, there is not point in distributing it earlier. However, if the data size is too large for single provisioning without introducing unnecessary delays, partial data updates according to the data segments (cf. above) may be distributed ahead of time (pre-emptive). Note that current processor architecture does not allow for easy background data transmission and in most cases the communication will stall execution of the main process.

As noted, a particular issue in this context consists in ensuring data coherency across segments and avoiding deadlocks. Intel's MESIF protocol over the Quick Path Interconnect [18] is one means to ensure cache coherency in a distributed environment at the cost of access delays. The basic principle of this approach consists in checking the consistency of a datum at access time by verifying it against all other replicated instances. Obviously this protocol does not scale very well and requires a specific type of architecture that will most likely not be supported in future large scale environments any more [19].

V. Exploiting Resource Fabrics

Since the system primarily caters for distribution of an application across a (potentially large-scale) environment for effectively sequential execution, how would this approach help solve the problem of dealing with future infrastructures? The system offers two major contributions that will be discussed in more detail in this section:

A. Supporting High Performance Computing

Though the primary concern is distribution and not parallelisation, the features and principles provided by a S(o)OS like environment deal with essential issues in large scale high performance computing: namely (1) exploiting cache to its maximum, (2) providing a scalable operating system, (3) matching the code structure with the infrastructure architecture and (4) managing communication and synchronisation in a virtual distributed shared memory environment.

Whilst the system does not explicitly provide a programming model that deals with scalability over heterogeneous, hierarchical infrastructures, it does support according models by providing additional and enhanced features to deal with such infrastructures. In particular it implements essential features as pursued e.g. by StarSS and PGAS (Partitioned Global Address Space): the concurrency information extracted from the memory analysis is consistent with the dependency graph that StarSS tries to derive [15] and could be used as an extension along that line. The main principle behind PGAS on the other hand consists in providing a virtual shared memory to the programmer, with the compiler converting the according read / write actions into remote procedure calls, access requests etc.

B. Office@World

A slightly different use case supported by the S(o)OS environment consists in the current ongoing trend of resource outsourcing into the web (see Introduction). As has been noted before, an essential part for web exploitation consists in maintaining interactivity whilst offloading demanding tasks. As we have shown, this does not only affect code, but equally data used in an application – accordingly, the features are of particular relevance in a future environment where most code and personal data will be stored in the web.

As discussed in detail in [6], the system thus not only allows that code and data becomes accessible and usable from anywhere at any time (given internet connectivity), but also virtually increases local performance of the system. This enables in particular frequent travelers to exploit a local system environment for their applications whilst using data and code from the web – future meeting places could thus offer compute resources that can be exploited by a laptop user, but also the other way round, i.e. a home desktop machine could replicate the full environment of the laptop. A low-power, portable device could thus turn into a highly efficient system by seamlessly integrating into the "resource fabric" without requiring specific configuration or development overhead [6]. This is similar to carrying an extended secure full work, respectively private profile with you.

VI. Conclusions

We have presented in this paper an approach to dealing with future distributed environments that span across the internet, multiple resources and multiple cores, but also across different types of usage and applications (such as High Performance Computing and Clouds).

"Resource Fabrics" are thereby an effective means to integrate compute units non-regarding their location and specifics. We have shown how a general view on computing resources can serve as basis for such a development, enabling higher scalability. To this end, a more scalable operating system is required that enables the explicit usage of resource fabrics as discussed here.

The according OS is still in a highly experimental stage, and more data is needed for concrete performance estimations – however, the basic principle has already been proven implicitly by in particular the Cloud Platform as a Service movement, where the dichotomy of interactive and "executive" part of the application is used to exploit remote resources. Many issues still remain speculative, though, for example regarding the execution of parallel applications where time-critical alignment is crucial. Along this line in particular the concurrent exploitation of remote resources, leading to potential time sharing of individual units (and their resources, such as I/O) still needs to be assessed more critical. With the general movement towards multi-cores, it can however generally expected that time-sharing during execution time is no longer a valid execution model.

In this paper we have used a lot of simplifications, mostly due to size-constraints. It will be noticed however, that we did not even touch upon the issue of security, which will be a major concern in scenarios such as the "office@world" one. So far, we did not touch upon this as other more technological issues need to be solved first - according concepts are expected in the near future.

References

[1] M. A. Rappa, "The utility business model and the future of computing services ," *IBM Systems Journal,* vol. 43 (1), 2004, pp. 32-42

[2] W. Fellows, "The State of Play: Grid, Utility, Cloud". Presentation at CloudScape 2009, OGF Europe. Available at http://old.ogfeurope.eu/uploads/Industry%20Expert%20Group/FELLOWS_Cloudscape Jan09-WF.pdf

[3] L. Schubert, K. Jeffery, B. Neidecker-Lutz, and others, "Cloud Computing Expert Working Group Report: The Future of Cloud Computing," European Commission 2010. Available at: http://cordis.europa.eu/fp7/ict/ssai/docs/cloud-report-final.pdf

[4] I. Foster and C. Kesselman (eds.), "The Grid. Blueprint for a New Computing Infrastructure.: Blueprint for a New Computing Infrastructure," Morgan Kaufmann Publishers, 1998

[5] J. Waldo, G. Wyant, A. Wollrath, and S. Kendall, "A Note on Distributed Computing," *Sun Microsystems Technical Report*, 1994. Available at: http://labs.oracle.com/techrep/1994/smli_tr-94-29.pdf

[6] L. Schubert, A. Kipp, B. Koller, and S. Wesner, "Service Oriented Operating Systems: Future Workspaces," *IEEE Wireless Communications*, vol. 16, 2009, pp. 42-50

[7] R. B. Miller, "Response time in man-computer conversational transactions," In: *Proceedings AFIPS Fall Joint Computer Conference* Vol. 33, pp. 267-277. ACM, New York: 1968

[8] J. Young and S. Smith, "Akamai and JupiterResearch Identify '4 Seconds' as the New Threshold of Acceptability for Retail Web Page Response Times," Akamai Press Release, 2006. Available at: http://www.akamai.com/html/about/press/releases/2006/press_110606.html

[9] D. Yuan, Y. Yang, X. Liu and J. Chen, "A data placement strategy in scientific cloud workflows," *Future Generation Computer Systems,* Elsevier, 2010, in print. DOI:10.1016/j.future.2010.02.004

[10] D. Nikolow, et al. "Knowledge Supported Data Acess in Distributed Environment," Proc. of Cracow Grid Workshop - CGW'08, October 13-15 2008, ACC-Cyfronet AGH, 2009, Krakow, pp. 320-325

[11] S. Ghemawat, H. Gobioff, S. Leung, "The Google file system," *ACM SIGOPS Operating Systems Review,* vol. 37, 2003, p. 29.

[12] G. Aloisio and S. Fiore, "Towards Exascale Distributed Data Management," *International Journal Of High Performance Computing Applications,* vol. 23, 2009, pp. 398-400.

[13] S. Grottke, A. Köpke, J. Sablatnig, J. Chen, R. Seiler and A. Wolisz, "Consistency in Distributed Systems," TKN Technical Report TKN-08-005, Technische Universit¨at Berlin, 2007 - available at http://www.tkn.tu-berlin.de/publications/papers/consistency_tr.pdf

[14] I. Foster, "Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering," Addison Wesley, 1995.

[15] J. Planas, R.M. Badia, E. Ayguadé, and J. Labarta, "Hierarchical Task-Based Programming With StarSs," *International Journal of High Performance Computing Applications*, vol 23 (3), 2009.

[16] A. Baumann, P. Barham, P.E. Dagand, T. Harris, R. Isaacs, S. Peterand others. "The multikernel: a new OS architecture for scalable multicore systems," In: *Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles,* pp.29-44. ACM, 2009

[17] L. Schubert, A. Kipp, and S. Wesner, "Above the Clouds: From Grids to Service- oriented Operating Systems," Towards the Future Internet - A European Research Perspective, G. Tselentis, J. Domingue, A. Galis, A. Gavras, D. Hausheer, S. Krco, V. Lotz, and T. Zahariadis, Amsterdam: IOS Press, 2009, pp. 238 – 249.

[18] Intel Corporation,"An Introduction to the Intel QuickPath Interconnect," Intel Whitepaper. Available at: http://www.intel.com/technology/quickpath/introduction.pdf

[19] F. Petrot, A. Greiner and P. Gomez, "On Cache Coherency and Memory Consistency Issues in NoC Based Shared Memory Multiprocessor SoC Architectures," In: *Proceedings of the 9th EUROMICRO Conference on Digital System Design*. IEEE Computer Society, 2006

2nd International Workshop on Medical Informatics and Engineering

MI&E 2010 will bring researchers, developers, practitioners, and users to present their latest research, results, and ideas in all areas of medical informatics, engineering, and artificial intelligence in medicine and health care.

MI&E 2010 will provide a forum for the exchange of ideas between physicians and computer scientists and engineers to address the important issues in those areas. Papers related to methodologies and applications in medical informatics, telemedicine, and eHealth are especially solicited.

The topics of interest include, but are not limited to:

- Health Care Information Systems
- Clinical Information Systems
- Medical Image Processing and Techniques
- Medical Intelligent Systems for different medical tasks
- Medical Expert Systems
- Applications in Medical and Health Sciences
- Artificial Intelligence Techniques in Health Sciences
- Medical Databanks, Databases, and Knowledge Bases
- Neurocomputing in Medicine
- Ontology and Medical Information Science
- Bioethics and Health informatics
- Web-Based Medical Diagnosis
- Health Care Process Management
- Intelligent Agents

Workshop Chairs

Abdel-Badeeh M. Salem (Chairman), Ain Shams University, Egypt

Kenneth Revett, Harrow School of Comp. Science,U of Westminster, London, United Kingdom

Organizing Committee

Florin Gorunescu, Dept. of Computer Science, Faculty of Mathematics and Computer Science, University of Craiova, Romania/Dept. of Mathematics, Biostatistics and Computer Science, University of Medicine and Pharmacy of Craiova, Romania, Romania

Gabriela Lindemann, Humboldt Univ. of Berlin, Dept of Computer Science, Lab of AI, Berlin, Germany

Abdel-Badeeh M. Salem (Chairman), Ain Shams University, Egypt

Kenneth Revett, Harrow School of Comp. Science,U of Westminster, London, United Kingdom

Program Committee

Sayed Abdel-Wahab, Information Systems Dept., Sadat Academy for Management Sciences, Egypt

Ennio Amori, Technology Department, Parma University Hospital, Italy

Amr Badr, Faculty of Computers and Information,Cairo University, Egypt

Hariton Costin, Fac of Medical Bioengineering, "Gr. T. Popa" Univ of Medicine and Pharmacy, Romania

Dursun Delen, Oklahoma State University, USA

Angela Di Tommaso, ASL of Catanzaro, Italy

Wojciech Glinkowski, Medical University of Warsaw, Poland

Amr Goneid, Computer Science Dept.,American University in Cairo, Egypt

Florin Gorunescu, Dept. of Computer Science, Faculty of Mathematics and Computer Science, University of Craiova, Romania/Dept. of Mathematics, Biostatistics and Computer Science, University of Medicine and Pharmacy of Craiova, Romania, Romania

Yutaka Hata, Graduate school of engineering, University of Hyogo Himeji, Japan

Halina Kwasnicka, Wroclaw University of Technology, Poland

Athina Lazakidou, Faculty of Human Movement and Quality of Life Sciences,University of Peloponnese,Department of Nursing, Sparti, Greece

Frank Lievens, International Society for Telemedicine & eHealth, Belgium

Gabriela Lindemann, Humboldt Univ. of Berlin, Dept of Computer Science, Lab of AI, Berlin, Germany

Yaohang Li, North Carolina Agricultural And Technical State University, USA

M. Madheswaran, Muthayammal Engineering College, Rasipuram, Tamil Nadu, India

Urszula Markowska-Kaczmar, Wroclaw University of Technology, Poland

Erzsébet Merényi, Electrical and Computer Engineering, Rice University, USA

Peter Millard, Emeritus Professor of Geriatrics, St. George's, University of London, Editor of Nosokinetics News, United Kingdom

Ioana Moisil, "Lucian Blaga" University of Sibiu,Faculty of Engineering -Dept. of Computer Science, Romania

Efstratia Mourtou, Hellenic Open University ,Head of Informatics Dept. of St. Andrew General Hospital Patras, Greece

Abdel-Badeeh M. Salem (Chairman), Ain Shams University, Egypt

Jan Rauch, Center of Biomedical Informatics, Institute of Computer Science of Academy of Sciences, Czech Republic

Kenneth Revett, Harrow School of Comp. Science,U of Westminster, London, United Kingdom

Thomas Schrader, Dept. of Informatics & Digital Media, Univ. of Applied Sciences Brandenburg, Germany

Francesco Sicurello, University of Milan Bicocca, Italy

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 687–691

ISBN 978-83-60810-27-9
ISSN 1896-7094

Agile methodology and development of software for users with specific disorders

Rostislav Fojtik
University of Ostrava, Department
of Computer Science, 30.dubna
22, Ostrava, Czech republic
Email: rostislav.fojtik@osu.cz

Abstract—**The paper deals with possibilities of information technologies when improving communicative skills of children with specific disorders, such as autistic spectrum disorders, Down syndrome, mental retardation, etc. The development of an application stemming from the communication system PECS (The Picture Exchange Communication System) and its Czech variant VOKS is the base of this paper to show specificity of the development and verification of software for the given group of handicapped users. The paper shows suitability of using agile methods of software development for a concrete application which is designed for users with specific disorders. It tries to show advantages and disadvantages of new methodologies, particularly Extreme Programming. Agile methodologies of software development appeared in the second half of the 90's of the last century. Thus it concerns new ways which have not been spread massively yet.**

***Key words*: Agile methodology, autism, Down syndrome, Extreme Programming, mental retardation, VOKS, RUP, testing**

I. Introduction

USE OF information and communication technologies has reached massive expansion recently. Computer users are to be found in every social and professional group. Computers simplify a lot of work tasks, make methods of communication easier and more accessible, intermediate new information, etc. These modern technologies can play a role of important supporting and compensational means for a group of users with specific needs. It is mainly people with some mental and health problems who face difficulties with communication, and common means of communication are usually unusable for them. Computers can act as a positive agent in such cases. However, hardware and software must be adjusted to the specific needs and it is not possible to implement experience gained and applied with common groups of users.

There is a potential group of users with specific disorders who could make use of the development of communication skills. E.g. people with autistic spectrum disorders, Down syndrome, forms of mental retardation and other development defects which cause problems to learn and use spoken language, written text, and other commonly used means of communication. Various methodics bringing interesting results have been worked out to develop communication skills of people with the above mentioned disorders. For instance, such methodics is the system PECS (The Picture Exchange Communication System) and its Czech variation VOKS. The base is a use of pictures representing particular concepts, things, activities. The user gradually becomes familiar with new concepts – pictures, which are then incorporated in their list. Choosing pictures and placing them on a sentence strip helps them learn to create even simple sentences. [7]

II. Metodics for learning

Based on the requirements from special school, we are developing an application which should facilitate communication with a child with specific disorders such as autism, Down syndrome, mental retardation and others. Autism is a mental handicap which demonstrates in worsened communication skills. Speech formation of people with an autistic spectrum disorder is delayed and some of them do not speak all their life or they use only limited number of words, frequently corrupted. Several methodics to improve their communication skills have been developed. [15]

One of them is the PECS methodics created in 1985. It is an alternative education and communication method for communication between an adult and a child handicapped by autism, mental retardation, and other specific disorders. [16] The fundamental principle is an exchange of a picture representing a particular thing for a real one. Continuous practice leads the child to learn how to use concepts correctly and improve its communication skills. The whole process is emphasised by stimulation. For instance, the child gets a chocolate if it brings the correct picture. Follow-up education leads not only to knowledge of pictures – concepts, but also to composing a short sentence from the pictures. The last phase of education encourages the children to comment independently the action around them and to answer direct questions. The system also helps some pre-school children to develop speech. An elaborate methodics of work with a system provides six training phases with examples, manuals, and advice [3] [2].

The Czech system VOKS mainly stems from the methodics and principles of PECS. However, it brings some improvements and it tries to adjust to particularities of the Czech language. Unlike PECS, it emphasises visual support of speech of both communicating partners. Thus there are changes to basic situations in all lessons. The methodics is divided into two basic parts. The first contains important information concerning teachers of the VOKS system and training environment. In addition, it describes preparation of individual tools for communication and the way of reward choice before the whole training of the communication begins. The second part contains educative lessons, which form the backbone of the education. The client learns to ask spontaneously for a favourite thing in exchange for a picture, to go on his own to the symbol container to get a picture and then to hand it in subsequently to the partner. They learn to ask different people about anything in an unfamiliar environment, to choose appropriate symbol from several pictures, to complete correctly a simple sentence on a sentence strip using pictures and then to ask by help of the sentence strip. Finally, they learn to react to various questions and to comment on the environment around them. The supplementary lessons then develop picture inventory and syntax [13].

III. Application voks

When designing a new application environment, it was necessary to get rid of all control features which were not directly related to the main use of the programme, and which could lead to early closing, new configuration, initialisation of new actions, etc. A lot of children find difficult to concentrate on realizing the needed performance. They frequently unwillingly click the mouse or wander the cursor round the screen. Thus it is highly important that children could not consciously or by mistake initiate functions indirectly related to the course of the programme.

The application control must not be demanding, unclear, or complicated at all. Generally common activities (such as double-click on a mouse) are almost impossible for numerous users with specific disorders. The control features of the programme must be conveniently large and their start-up and control as easy as possible. An ideal case would be if the application, having been opened, took up the whole screen of the desktop, being maximized all the time with no possibility of any change. The use of the mentioned software counts with the use of special hardware to control the computer. Users with motor handicap are e.g. supplied with IntegraSwitch, which works as an aspiration-expiration switch. Classical mice and keyboards are replaced by alternative positioning devices and keyboards, such as BickTrack, KidTrack, Roller Joyistick, BigKeys, IntelliKeys, various sensor buttons, and others. The main advantages are larger control features and more robust construction. On the other hand, the disadvantage is usually the price which exceeds several times the price of commonly used hardware. Purchase of such special devices can become unaffordable for many families with handicapped children. The solution could be use of touch screens, e.g. in new types of computers generally called nettops. It is a kind of cheaper computers containing all parts in one case together with a touch screen. The user can control the applications directly on the touch screen, which seems to be more convenient especially for children with specific disorders than using mouse or other positioning devices. Some children have motor problems when using mouse, moreover they have problems with moving the mouse while concentrating on another place - on the screen. Nevertheless, despite all of the above mentioned special hardware tools, it is convenient to create the developed software in a way to be usable even on common desktops, or notebooks with their usual peripherals.

IV. Agile methodologies

Current software development is characterised by shorter and shorter lifecycle. In addition, the development reflects running changes and dynamic technological development. It can be seen not only when developing complex and extensive information systems, but when developing specialised applications as well. Traditional software development methodologies do not meet current conditions, therefore those are agile methodologies that have taken up their place in recent years. Despite a certain difference, they have similar principles, which were expressed in Agile Manifesto in 2001 [9]

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Apart from extensive projects with clear and detailed assignment solved by large development teams, there are also small projects. Their development is secured by a small group of developers. In many of these projects, we cannot pre-define all requirements for the application, or the customer cannot do it. An example can be specialised applications for a targeted group of end-users. We could class here educational software for users with specific disorders.

Recent years have proved that so-called agile methods of software development can be used for this purpose. They were created on the basis that traditional rigorous methodologies are no longer suitable due to their formality and hugeness. It typically concerns web applications and closely specialised software. Customers are not usually willing to wait for months for a web application to be created. They prefer fast implementation even at the cost of continuously created modules. [1] [5]

Basic requirements on agile methodology cover:

- development is controlled by current requirements on functionality
- emphasis on continuous communication between the development team and customer
- emphasis on teamwork and team self-organisation
- regular and frequent transition of completed work to the customer
- not to avoid changes in the programme
- emphasis on the output programme quality before documentation
- changes should be actively followed and commented by the customer
- customer can actively interfere into the development

Agile methodologies include

- Dynamic Systems Development Method (DSDM),
- Adaptive Software Development (ASD),

- Feature–Driven Development (FDD),
- Extreme Programming (XP),
- Lean Development,
- Scrum,
- Crystal Methodology,
- Agile Modeling.

TABLE 1
COMPARISON OF BASIC DIFFERENCES BETWEEN TRADITIONAL AND AGILE METHODOLOGIES.

Agile methodologies	Traditional methodologies
Requirements on the application change frequently	Requirement do not change in the course of the development, or only minimally
Principle of freer cooperation of the development team	The development principle is based on fixed order
Developers should have experience needed for process adaptation	Less experienced developers
Emphasis on team communication	Directive management
High tolerance to changes	Low tolerance to changes
Testing throughout the whole course of the development	Testing is usually done at the end of the development
Documentation and models do not play the most important part in the development process	Documentation and the created model have high importance and the developers must follow them
The customer must be a part of the team	Customer's role is reduced to only the starting and final stages of the development
Smaller teams (2 to 10 developers)	Designed rather for larger teams

Agile methodologies are not suitable for all types of projects and all types of developing teams. Therefore there is a frequent combination of these two approaches. As for example, we can take methodology RUP, which gradually incorporates lots of agile techniques. RUP is currently representing a kind of a framework including both traditional and agile methodologies.

Despite the fact that the use of agile methodologies is increasing, their expansion is not still common. For example, 66% out of 3061 respondents of an inquiry held in 2008 said that they used at least some of agile procedures, such as iteration planning, unit testing, daily standup, release planning, continuous integration. [12] The most frequently accepted methodologies are XP, Scrum, Agile MSF, Agile Unified Process. On the other hand, according to inquires made in the Czech Republic, the situation is much worse. Only 43% of the respondents are aware of agile methodologies, but majority of companies do not use these methodologies at all. [4]

V. AGILE DEVELOPMENT OF VOKS APPLICATION

Initial efforts in the development stemming from the waterfall model have proved unusable. The customer (health staff of special school) did not have a clear idea about all functionalities and qualities of the developed communication software. Lots of the requirements had to be corrected, specified, and re-defined. Thus we proceeded to change the strategy and the software was developed according to an agile methodology. Elements and processes of Extreme Programming were taken as the basis. The basis of the whole XP software development is code-writing and testing. XP methodology is primarily designed for smaller teams with two to ten members, who work on frequently or less known assignments. Projects which take long or have difficulties in getting feedback (e.g. from technological point of view) are not suitable for this methodology. Automatic testing or version assembly is necessary for its implementation. XP is a flexible agile methodology emphasising interconnection of the proposal and implementation stages.

Basic activities are

1. Planning and Managing
2. Designing
3. Coding
4. Testing

XP has the following characteristic qualities:

- Continuous revision of the program code – frequent use of pair programming, when a pair of programmers works together on one code. Application of the principle led to removal of previously occurring problems in the code of the developed application.
- Testing – apart from unit tests, continuous testing by the customer was also successful. In our case, it was primarily health staff, but also children with specific disorders. Only the health staff from special school could define whether the application was suitable.
- Short iterations – classic methodologies usually transit the application to the customer after a long period, usually at the end of the development. This approach in the development of the educational application did not prove to be suitable. When an agile development methodology started to be used, we tried to make the shortest and most frequent iterations. They were given by time possibilities of the participants and we can see from the acquired experience that it would be convenient to make iterations more often than have been made so far.

Extreme Programming consists in five values:

- *Communication* – a large number of development problems lie in incorrect communication, not only among the team members, but with the customer as well. If XP is used, large teams assign a special role, so-called coach, who detects communication failures and secures correct communication. However, small teams should not underestimate good, continuous communication. Frequent communication is important not only among the developers, but with the customer as well.
- *Simplicity* – the methodology tries to develop software as easy as possible, not to deal with functionalities that are not currently important and that might be used in future. XP methodology says we should not create a more robust architecture than necessary for the moment. It also proved crucial in the development of the PECS application. With respect to frequently unclear customer's requirements, it was the most effective to solve the application for future needs. The development required more time and energy and

certain functionalities frequently proved to be unnecessary or unusable for a specific group of users.

- *Feedback* – is very important for correct development. It runs at several levels. One of them is testing, which should be performed at all development stages and not after the implementation stage. When developing program PECS, it proved necessary to test it continuously by its users. The main reason was the need of the developers to become familiar with possibilities of users with specific disorders. Many of commonly used approaches (particularly in the area of program control) are not usable for this group of users. Thus it was necessary to test more frequently and in all stages of the development what suits the users or what does not. The second target group of users was the health staff of special school that had to define and set up in the configuration part of the program an individual educational program for individuals. The configuration program requires simple control, still it offers a wide range of possibilities of individual setting according to specific requirement of the tutored child.
- *Courage* – A very important value of XP is courage to correct and remove errors at all costs. It even means removing a great part of the code or fundamental re-doing of the so-far architecture design. According to experience with the practical use of XP in companies, it seems that this requirement is difficult to be applied (more difficult in Europe than in the USA). The developers feel that removal of a great part of the code signs their failure and they are less likely to try further. [12] In the case of the PECS application development, it was necessary to remove a part of already created code several times because the corresponding application functionality proved to be unsuitable for a user with specific disorders. Despite those fundamental changes, it appears that the process helps to achieve the objective better than traditional methods.
- *Respect* – The team members should be interested in their colleagues' work. In case individuals work alone, with no close relations to others, XP will be unusable. This XP value closely corresponds to the emphasis on communication.

Why is agile methodology suitable for developing educational and communication software for users with specific disorders?

- The submitter does not have a clear idea about functionality of the application. Because there is a lack of experience with similar software, the development is frequently changed or corrected.
- A closely specialised group of future users requires frequent communication with submitters as well as continuous testing on by the developers and future users.
- The number of developed similar applications is very small, therefore there is a lack of experience with similar projects. Traditional development processes cannot be used in many aspects, particularly in the area of program control.
- The developers must get thoroughly acquainted with the environment and users, with whom they do not often get in touch and thus they do not know their abilities and requirements.

The used methodology has its disadvantages and risks. The requirement on frequent iterations proved to be hardly realisable. With respect to the nature of the solved project, direct communication of the developing team and the staff of the special school was necessary. It was quite a difficult task to secure this way of communication (taking into consideration time possibilities and distances between the participants). In order to achieve faster advance in the development of educational software, it would be necessary to communicate n shorter intervals, which was not possible to carry out successfully.

VI. Testing of voks application

Hand in hand with an increase in requirements and properties of software tools, there is a need of appropriate and profound testing. Despite the fact that the above described programme does not include any complicated functional structures or algorithms, and work of complicated and expensive devices is independent of its activity, it is necessary to secure its high reliability. Children suffering from e.g. autism find difficult to get used to a new environment and unfamiliar things. The time needed for the child to accept the programme and learn to work with it is usually very long. In case of serious disorders, there is a danger that the child will decline it or refuse to work with it. During testing of an application by children, there should not be any fundamental adjustments of the graphical interface and its control. Apart from classical procedures of testing of the programme functionality, it is important to observe how the child masters the application. It is not enough to use only usual static and dynamic testing means, analysis of the source code, monitoring by testing programme, special tests on memory usage or load, etc. [6] Thus the application incorporates internal mechanisms which monitor user's activities. Monitoring data about manipulation with programme objects are stored in XML files. The application contains an interface designed purely for parents and health workers which enables to evaluate both child's skills to work with the programme and its functionality. The output data then acts as a feedback for the creators of the programme as well as for parents and health workers, who can adjust the process of education more effectively. Another way to verify the programme functionality is methods used in quality oriented pedagogical research. It primarily concerns the method of observation [10]. The children use the application together with the parents or health workers. The adults act both as a pedagogue, who teaches the child to use and communicate with the programme, and an observer, who check for reactions and skills of the child. The observation results are very important not only for the educational process, but also for the development of the programme.

Unlike commonly used software, there are specific problems when testing:

- The target group of users is not large enough.
- The target group of users is very diverse and the level of the disorder considerably influences ability to work with the programme. It is necessary to take into consideration individual needs of the users and to enable more possibilities of setting and adjustment.

- The phase of learning how to use the programme is very time consuming. This results in longer period of testing than with usual applications. Children with higher level of disorder can take months to pass from one phase of the programme to another.

It is important to largely cooperate with parents and health workers because the children find difficult to get used to changes in environment and unknown people.

VII. Conclusions

Extreme Programming and other agile development methods have been implemented more and more in the past decade. It does not concern methodologies that would be suitable for all projects, they find their place in smaller developing teams. They can be used in projects with no clear initial definition, where the customers do not have a clear idea about the output product. Extreme Programming does not emphasise documentation of the development and its strict control. The basic element is a high level of communication among all team members and customers as well as frequent iterations. An advantage of this methodology was its possibility to react fast on customer's changes in requirements and possibility to adapt the program to users with specific disorders, even at the cost of removal of a great part of already written code. The methodology prefers fast reaction to a change before the plan completion, which proved to be important in the case of the developed software. There were plenty of changes and new requirements during the development, and their solution was more important for the output quality than following the time schedule of the development.

References

[1] Ambler S. Agile software development methods and techniques are gain traction, 2006, http://www.it-smc.com/Articles/Survey%20Says%20-%20Agile%20Works%20in%20Practice.pdf [on-line]

[2] Begeer S., Banerjee R., Lunenburg P. Brief Report: Self-Presentation of Children with Autism Spectrum Disorders, *Journal of Autism and Developmental Disorders*, Volume 38, Number 6 / July, 2008, ISSN 1573-3432

[3] Bondy S. A., Frost L. *What is PECS?*, http://www.pecs.com/WhatsPECS.htm

[4] Buchalcevová A. Agilní metody, jak dál? *Tvorba softwaru 2008*. VŠB TU EF, Ostrava 2008, s. 30–34. ISBN 978-80-248-1765-1.

[5] Buchalcevová A. Selection of the Appropriate Methodology for Concrete Project. In: *Objekty 2009*. Univerzita Hradec Králové, 2009, s. 37–48. ISBN 978-80-7435-009-2

[6] Buchalcevová A, Kučera J. Hodnocení metodik vývoje informačních systémů z pohledu testování. *Systémová integrace*, 2008, roč. 15, č. 2, s. 42–54. ISSN 1210-9479

[7] Charlop-Christy M. H., Carpenter M, Le L., LeBlanc L, Kelley K. Using the Picture Exchange Communication System (PECS) with children with autism: Assessment of PECS acquisition, speech, social-communicative behavior, and problem behaviors. *Journal of Applied Behavior Analysis*, 35, 213-231, 2002, http://seab.envmed.rochester.edu/jaba/articles/2002/jaba-35-03-0213.pdf\

[8] Collins Ch. T., Miller R. W. Adaption: XP Style, 2001, http://www.christophertcollins.com/papers/adaptationXpStyle_final.pdf [on-line]

[9] Fowler M. The Agile Manifesto: where it came from and where it may go, http://martinfowler.com/articles/agileStory.html [on-line]

[10] Gavora P. *Úvod do pedagogického výzkumu*, Paido, Brno 2000, ISBN 80-85931-796

[11] Hoekstra R. A., Bartels M., Cath C. D., Boomsma D. I. Factor Structure, Reliability and Criterion Validity of the Autism-Spectrum Quotient (AQ): A Study in Dutch Population and Patient Groups, *Journal of Autism and Developmental Disorders*, Volume 38, Number 8 / September, 2008, ISSN 1573-3432, http://www.springerlink.com/content/m184028q8305/?sortorder=asc&p_o=10

[12] Kadlec V. *Agilní programování*, Computer Press, Brno 2004, ISBN 80-251-0342-0

[13] Knapcová M. *Výměnný obrázkový komunikační systém - VOKS*. Praha: Institut pedagogiko – psychologického poradenství ČR, 2005, ISBN 80 – 86856 – 07 – 0

[14] Scherz J., Meng-Ju T., Broston S. *Adult Preferences Between Two Symbol Sets: Comparing Boardmaker and Overboard*, Wichita State University 2006, http://convention.asha.org/2006/handouts/855_1293Scherz_Julie_073114_111306111037.pdf

[15] http://en.wikipedia.org/wiki/Autism

[16] http://www.pecs-usa.com/

[17] http://www.mayer-johnson.com/ProdDesc.aspx?SKU=M125

[18] The State of Agile Development, 2008, http://www.versionone.com/pdf/3rdAnnualStateOfAgile_FullDataReport.pdf [on-line]

[19] Perez J. J., Guckenheimer S. *Software Engineering with Microsoft Visual Studio Team Systém,* Addison-Wesley Professional, 2006, ISBN 0321278720

3^{nd} International Symposium on Multimedia—Applications and Processing

Multimedia—Applications and Processing (MMAP'10) will be organized within the framework of the International Multiconference on Computer Science and Information Technology.

MMAP'10 will be organized by Software Engineering Department, Faculty of Automation, Computers and Electronics, University of Craiova, Romania ("Multimedia Applications Development" Research Centre).

Conference Background and Goals

Multimedia 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 2010 (MMAP 2010) 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 of interest

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

Costin Badica, University of Craiova, Romania
Thomas M. Deserno, Aachen University, Germany
Harald Kosch, University of Passau, Germany
Mohammad S. Obaidat, Monmouth University, USA
Ioannis Pitas, University of Thessaloniki, Greece
Vladimir Uskov, Bradley University, USA

Organizing Committee

Costin Badica, University of Craiova, Romania
Marius Brezovan, University of Craiova, Romania
Dumitru Dan Burdescu, University of Craiova, Romania
Mihai Mocanu, University of Craiova, Romania
Liana Stanescu, University of Craiova, Romania

Publicity Chair

Amelia Badica, University of Craiova, Romania

Program Committee

Reda Alhajj, University of Calgary, Canada
Christopher Barry, National University of Ireland, Ireland
Laszlo Böszörmenyi, Klagenfurt University, Austria
David Bustard, University of Ulster, U.K.
Richard Chbeir, Bourgogne University, France
Ryszard Choras, Institute of Telecommunications, Poland
Qi Chun, Xi'an Jiaotong University, P.R. CHINA
Vladimir Cretu, Politehnica University of Timisoara, Romania
Christos Douligeris, University of Piraeus, Greece
Rami Finkler, Afeka College of Engineering, Tel Aviv, Israel

Vladimir Fomichov, State University Moscow, Russian Federation

Romulus Grigoras, IRIT – University of Toulouse, France

Daniel Grosu, Wayne State University, USA

Voicu Groza, University of Ottawa, Canada

Janis Grundspenkis, Riga Technical University, Latvia

Rajkumar Kannan, Bishop Heber College, India

Valery Korzhik, State University of Telecommunications, Russian Federation

Igor Kotenko, SPIIRAS, Russian Federation

Reggie Kwan, Caritas Francis Hsu College, Hong Kong, P. R. CHINA

Michael Lang, National University of Ireland, Ireland

Rynson Lau, City University of Hong Kong, Hong Kong

Shiguo Lian, France Telecom R&D Beijing, China

Che-Chern Lin, National Kaohsiung Normal University, Taiwan

Bogdan Logofatu, University of Bucuresti, Romania

Giuseppe Mangioni, University of Catania, Italy

Cristian Mihaescu, University of Craiova, Romania

Andrea Molinari, University of Trento, Italy

Abdel-Badeeh M. Salem, Ain Shams University, Egypt

Marek Ogiela, AGH University of Science and Technology, Poland

Kazuo Ohzeki, Shibaura Institute of Technology, Japan

Toshio Okamoto, University of Electro-Communications, Japan

Enn Õunapuu, University of Technology, Tallinn, Estonia

Wilfried Philips, Universiteit Gent, Belgium

Dan Popescu, CSIRO, Sydney, Australia

José Luis Sevillano Ramos, University of Sevilla, Spain

Boris Shishkov, Delft University of Technology, Netherlands

Yoshimi Teshigawara, Soka University, Japan

George K. Thiruvathukal, Loyola University Chicago, USA

Christian Timmerer, Klagenfurt University, Austria

Stefan Trzcielinski, Poznan University of Technology, Poland

George Tsihrintzis, University of Piraeus, Greece

Miguel A. Vega-Rodriguez, University of Extremadura, Spain

Alexander Verbraeck, Delft University of Technology, Netherlands

Wita Wojtkowski, Boise State University, USA

Franz Wotawa, Technische Universitaet Graz, Austria

Jacek Zurada, University of Louisville, USA

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 695–701

ISBN 978-83-60810-27-9
ISSN 1896-7094

An Hypergraph Object Oriented Model for Image Segmentation and Annotation

Eugen Ganea
Software Engineering Department
University of Craiova
Craiova, Romania Email: ganea_eugen@software.ucv.ro

Marius Brezovan
Software Engineering Department
University of Craiova
Craiova, Romania Email: brezovan_marius@software.ucv.ro

***Abstract*—This paper presents a system for segmentation of images into regions and annotation of these regions for semantic identification of the objects present in the image. The unified method for image segmentation and image annotation uses an hypergraph model constructed on the hexagonal structure. The hypergraph structure is used for representing the initial image, the results of segmentation processus and the annotation information together with the RDF ontology format. Our technique has a time complexity much lower than the methods studied in the specialized literature, and the experimental results on the Berkeley Dataset show that the performance of the method is robust.**

I. Introduction

The hypergraph data model and object-oriented model make join to define the spatial relations between regions from images and the features of them; the composite model is called hypergraph object-oriented model ($HOOM$). Hypergraph data structure inherit the characteristics of the object model. Depending on hypergraph structure the complex spatial relations can be described easily and the attributive data can also be integrated more efficiently. Using hypergraph structures allows more advantages: using a single form for the representation of the combinations of classes and structural inheritance; allowing to avoid data redundancy (inherited values for sub-objects are taken from the super-objects) and to define complex objects (using undirected hyperedges) and functional dependencies (using direct hyperedges); support mechanism for identification through OID's and offers mechanisms for specifying multiple inheritance. In [1] was presented an overview of a hypergraph-based image representation that considered Image Adaptive Neighborhood Hypergraph ($IANH$) model.The proposed segmentation and annotation methods use a virtual graph structure constructed on the image pixels in order to determine the regions from the image and to allocate the labels for these which can give the semantic signature of the each region. Thus the image segmentation is treated as a hypergraph partitioning problem. The predicate for determining the set of nodes of connected components is based on two important features: the color distance and syntactic features [2], that are geometric properties of regions and their spatial configurations. The schema for the prototyping system is presented in Fig. 1:

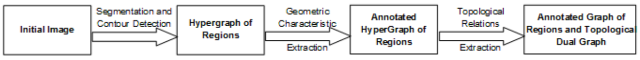

Initial Image
Segmentation and Contour Detection
Hypergraph of Regions
Geometric Characteristic Extraction
Annotated HyperGraph of Regions
Topological Relations Extraction
Annotated Graph of Regions and Topological Dual Graph

Fig. 1. The image processing system architecture

In the following sections there are discussed the next topics: in Section II we describe the image segmentation based on hypergraph structure; in Section III presents the method of the image annotation; Section IV describes the results of our experiments and Section V concludes the paper.

A. Related Work

In this section we briefly consider some of the related work that is most relevant to our approach. In the image segmentation area, the most graph-based segmentation methods attempt to search a certain structures in the associated edge weighted graph constructed on the image pixels, such as minimum spanning tree [3], or minimum cut [4]. The major concept used in graph-based clustering algorithms is the concept of homogeneity of regions. For color segmentation algorithms, the homogeneity of regions is color-based, and thus the edge weights are based on color distance. For image annotation, an approach based on graph is presented in [5], where the learning model is constructed in a simple manner by exploring the relationship among all images in the feature space and among all annotated keywords. The Nearest Spanning Chain method is proposed to construct the similarity graph that can locally adapt to the complicated data distribution. Object oriented database models [6] are based on the object-oriented techniques and their goal is representing by data as a collection of objects that are organized in hierarchy of classes and have complex values associated with them. The [7] describes the use of the $OODB$ in content-based medical images retrieval and the proposed approach accelerates image retrieval processing by distributing the workload of the image processing methods in the storing time.

II. Image Segmentation Technique Based on Hypergraph Structure

A. The Image Hypergraph Model

The construction of the initial hypergraph is based on an utilization of pixels from the image that are integrated into a network type graph. We used a hexagonal network structure on the image pixels for representation of the hypergraph $HG = (V, HE)$ and we considered two hyperedges of graph joining the pseudo-gravity centers of the hexagons belongs to hexagonal network as presented in Fig. 2.

The introduction of the hypergraph structure was in [8] and is a generalization of graph theory. The main idea refers to consideration of sets as edges and then a hypergraph is the family of these edges (called hyperedges). This concept represent more general data than graph structure and the theory of hypergraphs has proved to be of a major interest in applications to real-world problems such is image processing. The object model used for storing images, is based on the complex and different structure for each image that does not allow a simple data model using predefined data structures such as those used in relational databases. Relational databases have several limitations in representing an image: from the perspective of data representation model, in the relational database, links between two records are achieved through attributes primary key and foreign key. The records have the same values for foreign keys, primary that are logically related, although they are not physically linked (logical references). 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.

In an object-oriented database, each object in the real world can be modeled directly as an instance of a class; each instance has an OID and is associated with a simple or complex object. The OID stored in the database is not changed, while other fields associated object can be modified. This identity provides a good support for object sharing updates and simplifies management. Inheritance offered by object-oriented paradigm provides a powerful mechanism for organizing data, it allows to the user to define classes in an incremental way by specializing existing classes. As you can see, is achieved a triangulation of the image, which is in fact a decomposition of the image in a collection of triangles whose edges form the set V of nodes of the hypergraph HG. The condition for achieving a triangulation is satisfied, namely collection of triangles is mutually exclusive (no overlapping triangles) and fully exhaustive (all triangles meeting covers the original image). If it considered the edges which join the gravity pseudo-centers of the hexagons, it obtain a Delaunay triangulation [9]; the grid-graph is a Delaunay graph and based on planarity graph condition ($no\_edge \leq (3 * no\_vertex - 6)$) we demonstrated that the time complexity of segmentation algorithm is $O(nlogn)$. The algorithms for segmentation and the demonstration for complexity are presented in [10]. In the hexagonal structure, for each hexagon h in this structure there exist 6-hexagons that are neighbors in a 6-connected sense and the determination of indexes for 6-hexagons neighbors having as input the index of current hexagon is very simple. The main advantage when using hexagons instead of pixels as elementary piece of information is the reduction of the time complexity of the algorithms. The list of hexagons is stored such as a vector of integers $1 \ldots N$, where N, the number of hexagons, is determined based on the formula:

$$N = \frac{H-1}{2} \times (\frac{W-(W\%4)}{4} + \frac{W-(W\%4)-4}{4}) \quad (1)$$

where H represents the height of the image and W represents the width of the image.

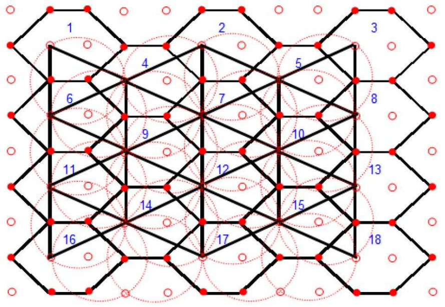


Fig. 2. The hexagonal structure on the image pixels

Each hexagon from the set of hexagons has associated two important attributes representing its dominant color and its pseudo-gravity center. For determining these attributes we use eight pixels: the six pixels of the hexagon frontier, and two interior pixels of the hexagon. For RGB space, the dominant color of a hexagon is determined as follows: is extracted three components (r, g, b) for each pixel of the hexagon; next step is sorting of the three vector components; last phase involves choosing as dominant components of media components located in positions 3 and 4 in the vectors, and dominant color calculation using the formula:

hexagon_color =

$$(r\_d\_color << 16)|(g\_d\_color << 8)|b\_d\_color \quad (2)$$

where r_d_color is the dominant color for the red component, g_d_color is the dominant color for the green component and b_d_color is the dominant color for the blue component of the RGB colorspace. We split the pixels of image into two sets, a set of pixels which represent the vertices of hexagons and a set of complementary pixels; the two lists will be used as inputs for the algorithm which construct the initial hypergraph of the initial image. The mapping of pixels network on the hexagons network is immediately and it is not time consuming in accordance with the following formula which determines the index for the first vertex of hexagon:

$$fv = 2 \times h + \frac{2 \times (h-1)}{columnNb - 1} \quad (3)$$

where fv represent the index of the first vertex, h the index of the hexagon and $columnNb$ represent the column number of the hexagon network. For representing the output of the image segmentation process we used the Attributed Relational Graph (ARG) [11]. The result of segmentation algorithm is stored as a graph where the nodes represent the regions and the edges represent the neighborhood relations: G = (V_r, E_r), where V_r is the set of vertices corresponding regions detected and E_r is the set of edges that describes the neighborhood relations. The spatial relations between regions are divided into 3 categories: distance relations, direction relations and topological relations. For determining these types of relations we choose for each region the following relevant geometric features: the pseudo-center of gravity; the distance between two neighboring regions; the length of common boundary of two regions and the angle which is formed by two regions.

B. The Image Segmentation Methods

In this subsection there is shown the algorithms for determining the initial hypergraph and the segmentation method. By using as unit element a hexagon, any two neighboring elements have in common two vertexes (an edge). The function $createHG$ which is describes in algorithm 1 produce the initial hypergraph for the initial image.

Algorithm 1: Create the initial hypergraph

```
Input: The list of color pixels from the hexagonal
       network: $L = \{p_1, \ldots, p_{6n}\}$
Output: The initial hypergraph $HG$ which correspond to
        the initial image
Procedure createHG($L$ ; $HG$);
* init $crtHyperEdge$;
for $i \leftarrow 1$ to $sizeof(L)$ do
    $hcrt \leftarrow L(i)$;
    * init $indexN$ as indexes from $L$ for the neighbors
    of $hcrt$;
    $indexM \leftarrow -1$;
    for $k \leftarrow 1$ to $6$ do
        $distRef \leftarrow colorDist$ ($L[indexN[k]]$, $L[hcrt]$);
        if $distRef$ are minimum value then
            $indexM \leftarrow k$;
        end
    end
    if !mark(edge[$L[indexN[indexM]]$, $L[hcrt]$]) then
        * add ($crtHyperEdge$,
        $edge[L[indexN[indexM]]$, $L[hcrt]]$];
        * mark $edge[L[indexN[indexM]]$, $L[hcrt]]$];
    end
    else
        * add $crtHyperEdge$ to $HG$;
    end
end
```

The output of segmentation algorithm 2 is the hypergraph corresponding to the segmentation image and it is computed based on the hexagonal grid-graph and the distance between the two colors for the RGB color space. The variable $colorsHexagon$ represents the vector colors of the hexagon and the others attributes of the class $Hexagon$ are $indexHexagon$ in the hexagonal structure of the virtual graph and $visitedHexagon$ which is used in the crossing network algorithms.

Algorithm 2: Determination of the segmentation hypergraph

```
Input: The total number of hexagons from hexagonal
       grid $n$
The hypergraph representation for an image ($HG$),
obtained with algorithm 1
Output: The hypergraph with the hexagons of regions
        $HG = \{\{he_1\}, \ldots, \{he_k\}\}, he_i =$
        $\{color, \{r_1, \ldots, r_p\}\}$
Procedure HGSegmentation ($N$, $HG$; $HG$);
* initialize the stack of hyperdges;
* get $indexH$ as index of unmarked hexagon
while ($indexH$ != -1) do
    * mark as visited the hexagon $indexH$;
    * push ($hyperdge[indexH]$);
    $HERegionItem \leftarrow newHyperEdgeRegion$;
    while !empty(stack) do
        $crtColorHyperedge \leftarrow pop()$;
        for each $crtColorHexagon$ from the set of
        $crtColorHyperedge$ do
            if (!mark($crtColorHexagon$) then
                * add $crtColorHexagon$ to
                $HERegionItem$;
                * push ($hyperdge[crtColorHexagon]$);
                * mark as visited the hexagon
                $crtColorHexagon$;
            end
        end
    end
    * add object $HERegionItem$ to list $HG$;
    * get $indexH$ as index of unmarked hexagon
end
```

The procedure $HGSegmentation$ returns the hypergraph with the hexagons of regions; the elements of hypergraph (the hyperedges) are determined for each distinct color from the input image as a list of regions that contain hexagons which have the same dominant color. The $HERegionItem$ is an instance of the class $HyperEdgeRegion$ and represents the data structure corresponding to an item from the output list. The attributes of the class $HyperEdgeRegion$ are: the color of the region and the list of hyperedges which represent the hexagons with the same color.

III. Image Annotation Technique Based on Hypergraph Structure

The management of ontologies, used for annotate of images, has two hierarchical levels that are closely associated. On the one hand, the low-level image contains specific properties as color, texture, shape, and the second level, which contains sematic of image that can be perceived human user. An ontology management system should model the low level that supports retrieval and inference level of their content. One scenario of using such a system can be that a user loads all ontologies in a given area, and allows selection of different objects in images and their correlation with the concepts of ontologies.

A. Ontologies and RDF format

For specifying the ontologies and the corresponding graph structure of the images segmented and annotated format we used RDF (Resource Description Framework). The RDF is a specification defined $metadata$ processing, providing interoperability between different applications such as an exchange of information, the purpose of understanding the semantics. To use the method of reasoning described in the ontology-based knowledge bases, we used $RDF2Jess$ model, a hybrid model that can be used to fill the gap between RDF and $Jess$. Based on domain knowledge and using Protégé [12] ontology editor, this method turns the RDF format in $Jess$ facts using XSL transformations on XML syntax and additional rules in Jess. For these rules redefined based on RDF semantics $Jess$ inference system is used to implement the reasoning. Predefined rules are used to check consistency and to determine the characteristics of RDF vocabulary. Deducted $Jess$ assertions are helpful for phase domain ontology modeling to assess and refine ontology. According to different levels of expressiveness, $RDF2Jess$ could be extended to new $SWRL2Jess$ where $SWRL$ extends the set of axioms to include Horn rules. The conversion of syntactic and semantic RDF in $Jess$, allows replacement of ontology in RDF format using $Jess$ reasoning engine. Ontology conversion into Jess facts and rules is done in four steps:

- first step is the ontology construction, ontology editor used $Protege$, provides a plug-in RDF ontology to support development. The taxonomy of knowledges into classes, features, restrictions was accepted by experts in the field and by software developers, because this paradigm is very similar to object-oriented modeling (UML). Lately, most ontologies have been formalized using standardized $RDF(S)$ and can be reused to extend the rules, if necessary;
- the second step is to represent the transformation of RDF syntax in $Jess$ syntax using $XSLT$, the output file consists of $Jess$ facts. If support ontology language semantics is specified as $Jess$ rules, matching specific keywords is no longer necessary in the transformation;
- the third step is to combine file $Jess$, including $XSLT$ transformation result, and RDF predefined rules. Moreover, external queries and $Jess$ rules can also be added to the composition similar properties;
- the last step is running the system of rules $Jess$ inference system. The rules defined, is the classification and consistency checking characteristics. The output containing erroneous messages indicating the presence of incorrect syntax in RDF ontology processed.

To implement the RDF, semantics are defined to represent the additional facts and their relations with $RDFS$ primitives such as $rdfs : Class$. This approach prevents duplication of semantic information in the database, and an overview of how to store links between visual concepts/concepts domain and related regions is given by Fig. 3.

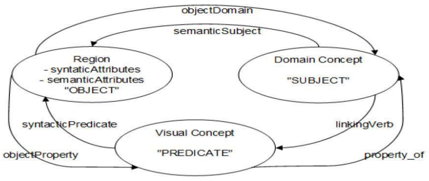


Fig. 3. Triple link: visual concept/semantic concept/Region

In Fig. 4 are presented the classes which are used for specification of the $HOOM$ in the image annotation processus:

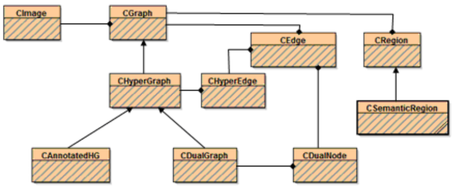


Fig. 4. The hypergraph object data model

B. The Image Query Specification

To specify queries we have expanded the query language $GOQL$ using data structures of type hypergraph - $HGOQL$ (Object Query Language Hyper-Graph). In [13] is proposed $GROOVY$ model (Graphically Represented Object-Oriented Data Model with Values) which represents a proposal to formalize the object-oriented model based on hypergraph structures type. It defines a set of data structures for object model data: (I) Value diagram that defines the attributes that contain a class of objects. Items can be atomic or multi-value, (ii) The court: defines a lot of items under the diagram. An object is a pair $O = < id, v >$, where id is the identifier of the object and v is the value of the object (his properties), (III) Dependency of functional values: are used in the scheme to check that the value of a set of attributes determine in a unique way the value of another attribute; (IV) Object diagram: is a triplet $< N, F, S >$, N - diagram value, F a lot of dependency of functional values for N and S - a lot of subsets of N. This structure is defined using hypergraphs by establishing a corresponding one-to-one between object diagram $< N, F, S >$ and an interpretation where N is the oriented nodes of a hypergraph, F is its direct hyperedges, and S, undirect hypeedges. A manipulation language of hypergraph (HML - Hypergraph Manipulation Language) for query and update them has with two operators querying based on identifier or value, eight operators for inserting and deleting the hypernodes and the hyperedges. During the developed language ($HGOQL$) we implemented only operators who refers to the query. The grouping of the indexes attributes (which includes enable/disable attributes through facet $index - propagation$) is achieved by Algorithm 3 based on a function implemented in the states of each class. The function $hyperGraphGroup$ that does clustering indexes, used as support for storing and linking indexes a $hypergraph$ [14], implemented by the $CHypergraph$ class, subject to the previous result that output algorithm ($indexThis$) is an instance of $CHypergraph.$ Choosing $hypergraph$ type structure to represent indexes was made because this type of structure is very good for browsing and retrieving images corresponding graph processed. The function $HyperGraphGroup$ properly algorithm is presented in Algorithm 4.

Algorithm 3: Grouping the attributes index

```
Input: The current instance
Output: The index of current instance
Procedure groupIndexObject ($thisObject$;$indexThis$);
$attributeSet \leftarrow atributesOf$ ($thisObject$);
initialize $attributeIndexSet$;
for $attribute \leftarrow attributeSet$ do
    if $attribute.pattern\text{-}match\text{-}facet$ $is$ $reactive$ then
        add $attribute$ to $attributeIndexSet$;
    end
end
for $attribute \leftarrow attributeIndexSet$ do
    if $attribute.index\text{-}propagation\text{-}facet$ $is$
    $non - inherit$ then
        remove $attribute$ to $attributeIndexSet$;
    end
end
initialize $indexThis$;
for $attribute \leftarrow attributeIndexSet$ do
    $indexThis \leftarrow hyperGraphGroup(indexThis,$
    $attribute)$;
end
```

Algorithm 4: Function hyperGraphGroup

```
Input: The index of current instance, the current attribute
Function hyperGraphGroup ($indexThis$, $attribute$);
initialize $currentHEdge$;
if $isSyntactic$ $(attribute)$ then
    $currentHEdge \leftarrow$ addToSyntacticHyperEdge
    ($attribute$)
end
else
    $currentHEdge \leftarrow$ addToSemanticHyperEdge
    ($attribute$)
end
$hyperEdgeSet \leftarrow hyperEdges$ ($indexThis$) U
$currentHEdge$;
clear $indexThis$;
for $hyperEdge \leftarrow hyperEdgeSet$ do
    if $isNonTopological$ $(hyperEdge)$ then
        add $hyperEdges$ to $indexThis$
    end
    else
        add $hyperEdge$ to $indexThis$
    end
end
return $indexThis$
```

IV. Experiments

In this section experimental results are highlighted demonstrating that the method presented produces a good image segmentation and annotation, and extraction of outlines for visual objects from different images without the need for parametrization of the method depending on image processing. For this purpose, we used human segmentation for color image from Berkeley Segmentation dataset ($BSDB$).

A. Image Segmentation Experiments

After problem examination of validating the quality of segmentation, we proposed an effective evaluation system which shows the comparative results of segmentation both obtained with the algorithm implemented and the three alternatives segmentation methods: the "Mean-Shift" method (MS) [15], the "Local-Variation" method (LV) [3] and the "Normalized-Cuts" method (NC) [4]. In the last part is presented an analysis performed for a set of synthetic images. The $BSDB$ database contains two sets of images: a lot of training with 200 images and a lot of test 100 images. For each image is available a set of human segmentations. This set contains between four and eight segmentations specified in the form of images labeled using a special format. In [16] showed that human segmentation, although vary in detail, are according with each other for the segmented regions by humans are exposed to a finer level of detail and in this situation can merged so as to appear as to extracted regions a coarse level of detail. To assess performance of segmentation method were segmented images of crowds test and were used two measures of quality: the value function harmonic mediation ($F - measure$) and performance value ($P - measure$) [17]. Value of $F - measure$ [18] determined by a combination of precision values (P) and re-call (R) is calculated for each segmentation and number the correct segmentation comparing to the wrong. In Fig. 5 are the results of contour detection for a group images, considering the four methods of segmentation. In the three methods chosen for comparison was considered when the result value is the measure of $F - measure$ is the maximum, namely: MS method, DW = 10 and PW = 8, for the LV method, the value used was k = 900, for NC method, the value used was $nTresh$ = 25. In the second part of experiments was considered a set of synthetic images generated so the RGB colorspace to be presented with non-null value a single channel of the three. In figure Fig. 6 are examples of such images generated. The purpose of generate and use this set of images was to determine the optimal formula for calculating dominant color of a hexagon (formula

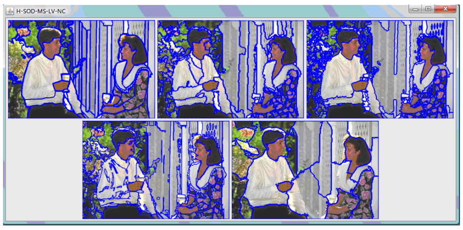


Fig. 5. Results for image segmentation: human segmentation, SOD segmentation, MS segmentation, LV segmentation and NC segmentation

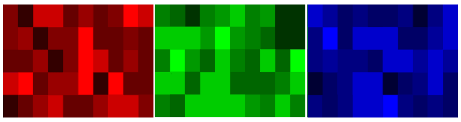


Fig. 6. Samples of synthetics images

2) and on the other side validation formulas used to determine threshold values used in the two stages of segmentation. To achieve these goals we generated for each image, an attached file in XML format. File picture represented in XML as a list of rectangles, for each rectangle specified coordinates of points on the left/top, that right/down, channel number and the value for that channel with values between [0 . . . 255]. After segmentation a synthetic image is comparing the list of regions obtained with the rectangles list extracted from the XML file by using a parser. As a result of comparison the aim is to modify channel values after considering the dominant color from the level of a hexagon. The closest values were obtained for two experiments that have used the formula 2 for calculation of hexagon color. threshold values used in the two stages of segmentation.

B. Image Annotation Experiments

The following phase after the segmentation step involves the addition of labels to the semantic regions. There are two phases of this stage, manual annotation (training phase) and automatic annotation. For the first phase, representative images are selected for the considered domain. Manual annotation scenario assumes that the first step, loading the domain ontology. In the next step, the user selects the regions of the segmented image and put them in correspondence with the ontology domain concepts. While the regions are manually annotated, an XML file with the annotated information is generated. To identify a region in space of the image is stored in the XML file the hexagons list region which are founded on its border. This information is necessary to establish a visual correlation between the XML node for a region and this; this is done by double selection: when selecting a region already annotated in the tree view of XML corresponding node is selected automatically. Fig. 7 presents the corresponding GUI of the annotation phase.

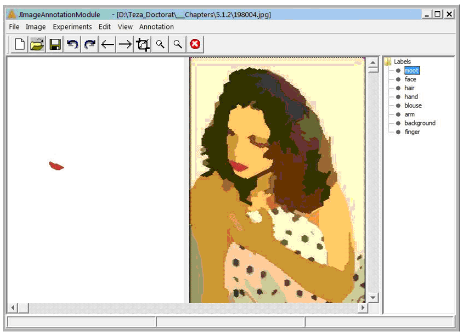


Fig. 7. Graphical user interface subsystem for image annotation

Based on syntactic and semantic properties of manually annotated images is constructed a decision tree which is translated into a set of rules used for automatic annotation of image regions segmented test. We used to add semantic tags the decoration technique of the hypernodes of the hypergraph in accordance with the RDF format representation. The obtained hypergraph as a result of the processing steps is serialized as a system of objects, which have links in accordance with existing relationships (inheritance or composition) on the all classes which are implemented.

V. Conclusion

In this paper presents an hypergraph object-oriented model for image segmentation and annotation when is considered as the input the information extracted from the image. The unified method for image segmentation and image annotation uses an hypergraph model constructed on the hexagonal structure. The hypergraph structure is used for representing the initial image, the results of segmentation processus and the annotation information together with the RDF ontology format. The hypergraph representation of images is the output of all the phases of system: the segmentation phase and the annotation phase. Our technique, which combines the hypergraph model with the object oriented model, has a good time complexity and the experimental results showed that the method can be yielded with good results regardless of the area of the images that come.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 CNCSIS IDEI 535 is gratefully acknowledged.

References

[1] Bretto, A. and Gillibert, L., *Hypergraph Based Image Representation*, Graph Based Representations in Pattern Recognition, 2005.

[2] Bennstrom, C.F. and Casas, J.R., *Binary-partition-tree creation using a quasi-inclusion criterion*, Proc. of the Eighth International Conference on Information Visualization, 2004.

[3] Felzenszwalb, P.F. and Huttenlocher, W.D., *Efficient Graph-Based Image Segmentation*, International Journal of Computer Vision, 2004.

[4] Shi, J. and Malik, J., *Normalized cuts and image segmentation*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 2000.

[5] J. Liu, M. Li, W.-Y. Ma, Q. Liu, H .Lu, *An adaptive graph model for automatic image annotation*, Multimedia Information Retrieval, 61–70, 2006.

[6] W. Kim, *Object-Oriented Databases: Definition and Research Directions*, IEEE Transactions on Knowledge and Data Engineering, 2(3):327–341, 1990.

[7] C.Jr. Traina, A.J.M. Traina, R.A. Ribeiro, E.Y. Senzako *Content-based Medical Images Retrieval in Object Oriented Database*, Proceedings of 10th IEEE Symposium on Computer-Based Medical System - Part II, 67–72, 1997.

[8] Berge, C., *Hypergraphs*, North-Holland Mathematical Library, 1989.

[9] Delaunay, B., *Sur la sphére vide*, Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk, 7:793–800, 1934.

[10] Burdescu, D.D., Brezovan, M., Ganea, E., Stanescu, L., *A New Method for Segmentation of Images Represented in a HSV Color Space*, Advanced Concepts for Intelligent Vision Systems, Bordeaux, France, 2009.

[11] Hong, P. and Huang, T. S., *Spatial pattern discovery by learning a probabilistic parametric relational graphs*, Discrete Applied Mathematics, 139:113–135, 2004.

[12] *Protégé project*, http://protege.stanford.edu/ (consulted 17/06/2009).

[13] Levene, M. and Poulovassilis A., *An Object-Oriented Data Model Formalised Through Hypergraphs*, Data and Knowledge Engineering (DKE), 1991.

[14] B. Goertzel *Patterns, Hypergraphs and Embodied General Intelligence*, IJCNN, Neural Networks International Joint Conference on, 451–458, 2006.

[15] Comaniciu, D. and Meer, P., *Robust analysis of feature spaces: Color image segmentation*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1997.

[16] Martin, D., Fowlkes, C., Tal, D., and Malik, J., *A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics*, IEEE International Conference on Computer Vision, 2001.

[17] Grigorescu, C., Petkov, N., and Westenberg, M., *Nonclassical receptive field inhibition*, IEEE Transactions on Image Processing, 2003.

[18] Fowlkes, C., Martin, D., d anMalik, J., *Learning affinity functions for image segmentation: combining patch-based and gradient-based approaches*, IEEE Conference on Computer Vision and Pattern Recognition, 2003.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 703–708

ISBN 978-83-60810-27-9
ISSN 1896-7094

Classification of Image Regions Using the Wavelet Standard Deviation Descriptor

Sönke Greve, Marcin Grzegorzek, Carsten Saathoff and Dietrich Paulus
Department of Computer Science, University Koblenz-Landau
Universitätsstraße. 1, 56070 Koblenz, Germany
Emails: {sgreve,marcin,saathoff,paulus}@uni-koblenz.de

***Abstract*—This paper introduces and comprehensively evaluates a new approach for classification of image regions. It is based on the so called *wavelet standard deviation descriptor*. Experiments performed for almost one thousand images with region segmentation given provided reasonable results for a very general application domain: "holiday pictures".**

I. Introduction

In the field of research dealing with image classification and understanding we follow a multi-layered concept. Image data is described in high level and low level semantics. High level semantics are represented by ontologies and low level semantics in feature comparison techniques. This paper is focused on wavelet features following the proposal of the *Wavelet Standard Deviation Descriptor (WSD)* [1] used for the classification of texture patterns. Here it is used in order to classify regions of previously segmented images. The task at hand is to find a label for each region in any given holiday photo that can later be used in image management applications or retrieval systems e.g. to automatically tag inserted photos. The major motivation for this research work lies in the two following statements:

1) finding adequate region labels can improve the research on high-level semantics (narrowing the semantic gap)
2) the problem of image region classification can be reduced to a problem of texture pattern classification

Indications on the first assumption can be found in [2] but in general it is task of the high-level domain. The second assumption is analyzed in detail. Therefore two implementations using the WSD are presented each tested under different modalities. Both classification algorithms compare the WSD of an image region to the WSD of each pre-defined concept. The concepts used are shown in table I. They were trained using a subset of the available images. The region-labels of those images were manually annotated beforehand providing ground truth.

This introduction is followed by a brief overview of similar research in II. Section III introduces the environment of the performed tests followed by the implementation of the mathematical constructs in IV. The results are shown in V with a view on issues, proposals and future developments in VI

II. Related Work

The field of image region classification offers many approaches. In [3] seven feature extraction and comparison methods are compared in their performance of correct image retrieval out of the WANG[1] and the IRMA[2] databases. The evaluated feature extraction methods are image features ($\widehat{=}$ pixel values), color histograms [4], invariant feature histograms [5], Gabor feature histograms [6], Tamura texture feature histograms [7], local features [8] and region based features [9]. As the IRMA database contains too specific medical images, the images of the WANG database get close to the sources of this paper though it is not limited to the loosely defined class of holiday photos. The provided ground truth is also more focused on semantic classes, e.g. "Africa," "food," "monuments," than on elementary texture patterns like "sand," "sky" or "foliage."

In many cases features are generated using an entire image without providing information about specific regions in the image. As soon as regional features are provided the focus lies on object detection. A task very similar to this paper though using different features is presented in [2]. Also similar is [10] using Gabor features [11] on segmented satellite images which provide optimal conditions for texture analysis as they barely suffer from perspective distortion. The idea of this study is to ignore the perspective distortion and reduce the problem of identifying image regions to a problem of identifying textures. The publication closest to this work is [12]. It also faces the task of concept similarity measures in an even larger scale and therefore affiliation estimation. The features used are two-dimensional hidden Markov models.

III. Design

A set of 922 RGB-images with a resolution of 800 by 600 pixels was provided to represent "holiday images". They

TABLE I
The pre-defined concepts to be found in images

- building
- foliage
- mountain
- person
- road
- sailing boat
- sand
- sea
- sky
- snow

[1] http://wang.ist.psu.edu/docs/related.shtml
[2] http://ganymed.imib.rwth-aachen.de/irma/index_en.php

originate in the available Database of the K-Space Project[3]. To avoid an *over-fitting* of the (to be generated) concept descriptors the data was separated into three subsets. One for the training of the concept classifiers, a second one to validate parameter changes in the algorithm and a third one to test the final classification performance. For the training- and validation-subset 230 images were each randomly chosen out of the provided images. The remaining 462 photos were used as test-subset.

A. *Concept Training*

The aim of this study is to enable a comparison of image regions against a set of pre-defined concepts using WSDs. Therefore a WSD-representation of the concepts must be developed. As a WSD is a feature vector of invariant length (for identical image resolutions) it was chosen to represent each of the features by mean and standard-deviation. To generate the concept-descriptors the following steps were taken:

1) Compute the WSDs of each image region within the training-set and select one representative per region.
2) Group the representative WSDs of all the images within the training-set by their concept label using a manually annotated affiliation list.
3) Calculate the mean and standard-deviation of each feature value of a WSD over the given concept.
4) Store mean and standard-deviation as representative for the corresponding concept in a concept descriptor.

A concept is now defined by a label (e.g. sailing-boat) and a corresponding descriptor which is a mean and standard-deviation value for each computable feature value.

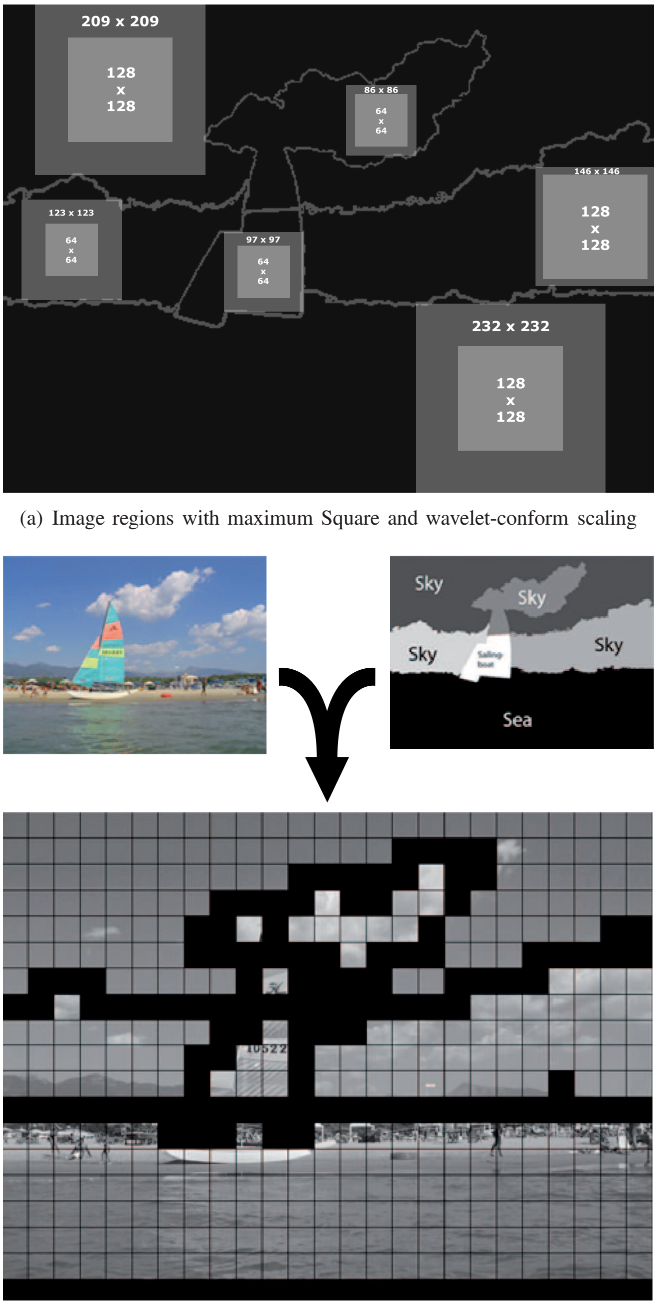
B. *Modalities & Validation*

When dealing with texture data there's usually no need to provide color information as it multiplies the processing time at least by the number of color-channels used. There is also the problem that color information is strongly dependant on the environment's illumination. However color channels can include patterns that supply additional information of the structural character. During the validation it showed that e.g. the concepts snow and sand tend to be very similar in their structural consistency. Here a separate analysis of the color channels can add additional precision to the results. Therefore three different color-spaces were tested - gray-scale, RGB, and YC$_r$C$_b$. The gray-scales are obviously representing structure-only information. RGB was chosen in regard to the simplicity of the approach as the available images are provided in this format and it is to be expected that most of the sources of ”holiday images” (e.g. cameras) have the same output. In case of satisfying results without color-space transformations the run-time performance would benefit. YC$_r$C$_b$ was chosen as proposed in [1] with the argument of describing structural information in the Y-component and the distribution of color information in the C$_r$- and C$_b$-components.

When working with wavelet transformations the image or region to be processed must fulfill the conditions

$$(x = y) \wedge (x = 2^n : n \in \mathbb{N}_{>0}) \tag{1}$$

where x and y define the pixel resolution. As image regions mostly don't fit these requirements it is necessary to choose a representative for every region in order to compute the WSD-features. To meet these conditions two approaches were tested.



(a) Image regions with maximum Square and wavelet-conform scaling

(b) Determination of valid patches using a 32 by 32 pixels grid

Fig. 1. maximum Square 1(a) and grid overlay 1(b) feature selection methods

a) maximum square: The first approach finds the largest square fitting in an image region and scales it down to the

[3] http://kspace.qmul.net:8080/kspace/

next valid wavelet-conform resolution as shown in fig. 1(a). Then the WSD was computed stopping after the third level of the wavelet transformation (see IV-A) in order to assure a common vector size for later comparison. This limits the patch resolution from 8^2 up to 512^2 pixels using 800 by 600 pixel images.

b) grid overlay: The second approach uses a square grid overlay to create a pattern with wavelet-conform patches. The size of the grid's patches was set to 32^2 pixels. When using YC$_r$C$_b$ color-space a resolution of 8^2 pixels was additionally tested. Only patches lying completely within a single region were considered for the WSD computation. The process of finding out the valid patches ist depicted in figure 1(b). To determine a single representative for an image region different methods were used:

- *first patch:*
  when processing the image, the first found patch of an image region was selected as representative (dummy implementation).
- *mean value:*
  the WSD of each patch was computed. All WSD features of an image region were merged into a representative WSD for the image region using mean for every feature (per filter and level).
- *maximum affiliation:*
  the WSD of each patch was computed. Then they were compared to each concept resulting in 10 affiliation values[4] per patch. The patch with the largest affiliation value[4] was selected as representative.
- *largest difference:*
  similar to *max. affiliation*. Though the representative was selected by the largest difference of the highest two affiliation values[4] per patch.

Table II lists the modes and combinations the validation was performed in. The rows show the representative selection method, the columns show the colorspace. Content is the used patch size of the specific method. The *maximum square* entry refers to the first selection approach and therefore doesn't use a static patch size. All parts of the algorithm were validated against the validation-set of images concerning error handling and parameter optimization.

TABLE II
VALIDATION PERFORMED REGARDING COLOR-SPACE AND REGION REPRESENTATIVE SELECTION METHOD. THE CONTENTS SHOW THE APPLIED PATCH RESOLUTION OF THE GRID OVERLAY IN PIXELS

	gray-scales	RGB	YC$_r$C$_b$
first patch	32^2	-	-
mean value		32^2	8^2 and 32^2
maximum affiliation			
largest difference			
maximum Square	8^2 to 512^2	-	-

C. Algorithm Testing

In order to avoid bias due to overtraining of the algorithms on a specific set all presented results in this paper were created using the images of the test-set with the unaltered parameters or fixes created during the validation process.

IV. Implementation

The implementation of this study covers four partial tasks.

- create the wavelet-transformation for the input images
- compute the WSD out of the wavelet-transformed images
- describe any pre-defined concept using WSDs
- a *concept-to-WSD* comparison method must be developed

As the descriptor for the WSD values was decided to be the mean and standard-deviation over the computed features in the training set, there won't be a more detailed description for point three. The implementation of the wavelet-transformation, the WSD computation and the selected comparison method are presented in the following.

A. The Wavelet Standard Deviation Descriptor

The wavelet-transformation computes the frequencies within different levels of a signal. In case of image processing it is interpreted as the frequencies of a single channel's discrete values within the input image—most likely gray-scale images. The implementation uses the Haar wavelet [14] following the proposal of [1]. It results in a transformed image like the example in figure 2. Every level of the wavelet transformation a partial HL-, LH-, HH- and LL-image is created where H is a high-pass filter and L a low-pass filter. The partial images are created following:

1) Apply the horizontal filter to the input image (e.g. H).
2) Eliminate every second column.
3) Apply the vertical filter (e.g. L).
4) Eliminate every second row.
5) In case of HL-, LH- and HH-images store the result in the specific position shown in figure 2 (e.g. top-right for HL). in case of an LL-image compute the next wavelet-level using the LL-image as input.
6) Repeat this process until the desired wavelet-level is reached or the LL-image has a resolution of 1 by 1 pixels.

The *wavelet standard deviation descriptor* (WSD,[1]) is a vector that uses the weighted standard deviation of each partial HL-, LH- and HH-image as feature values. The weighted standard-deviation of the LL-image and its mean are used as final feature values in the vector. The *maximum Square* approach uses a depth of three wavelet-levels as its WSD. In the *grid overlay* approach all level-features were calculated and used as WSD. In the second case the size of the feature vector is therefore always:

$$\varepsilon = 3 \cdot \#_k + 2 \tag{2}$$

Where $\#_k$ is the number of computed levels.

[4] described in IV-B

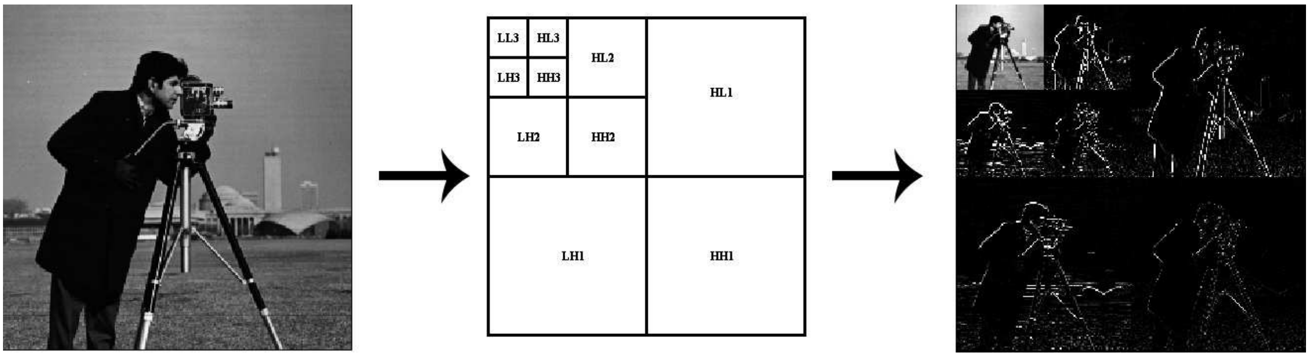


Fig. 2. Wavelet transformation with three levels on an image by [13]. The intensities in the LL-image on the right were normalized for visualization purposes.

The WSD vector is then defined as:

$$WSD = \{\frac{\sigma(LH_k)}{2^{k-1}}, \frac{\sigma(HL_k)}{2^{k-1}}, \frac{\sigma(HH_k)}{2^{k-1}}, \dots, \frac{\sigma(A)}{2^{k-1}}, \mu(A)\} \quad (3)$$

$LH_k, HL_k, HH_k =$ the partial image of the k-th level
$\sigma(image) =$ standard deviation of the image
$\mu(image) =$ mean value of the image
$k =$ the index of the specific level $[1..(\varepsilon - 2)/3]$
A = the approximation image (the lowest LL-image)

B. Comparison Method

In [1] a similarity measure for WSDs is offered based on the weighted difference of the corresponding features. It was decided to implement a probabilistic measure similar to the offered approach. As after section III-A each feature of a concept is represented by mean μ and standard-deviation σ, it is possible to compute a relative similarity ρ for each feature f of a WSD using the probability distribution:

$$\rho_k = exp(-\frac{(f_k - \mu)^2}{2\sigma^2}), k \in [1..\varepsilon] \quad (4)$$

The final concept affiliation was tested with two different aggregation modes - the sum of the single values and their product.

$$\gamma_{sum} = \Sigma_1^{\varepsilon} \rho_k \quad (5)$$

$$\gamma_{product} = \Pi_1^{\varepsilon} \rho_k \quad (6)$$

V. Results

Within the 462 images of the test-set a manual annotation was provided for 2960 of the contained regions. Any result presented in this paper is referencing to this number of regions providing ground truth. Applying the grid presented in section III-B with a patch size of 32^2 pixels results in 2521 classifiable regions. The remainder of the regions didn't fit the grid in so far as none of its patches were completely located within its boundaries and therefore no features could be extracted. A patch size of 8^2 pixels could cover each region. Anyway these circumstances make it necessary to distinguish between two types of results. First there is the *feature classification rate* (feature rate) which represents the classification performance only in regard to all computable features. And second there is the *task classification rate* (task rate) which shows the overall classification performance treating ignored regions as falsely classified.

The computed classification rates are presented in Table III. It shows that a region representative is best selected by creating a mean WSD over all the region's patches. The feature aggregation to be favored is summing up the feature values (compare equation 5). Using color information doesn't strongly influence the classification rates though using RGB shows slight advantages. Applying the smaller sized grid improves the results by less than one percent when comparing the best classification rates of each size. Instead the algorithm's run time increases a lot as there are 16 times as many features to be computed, merged and compared. The *maximum squares* approach delivered results worse than the task rates of a 32^2 pixel sized grid, both using gray-scale converted images.

It showed up that none of the patches provided any data in the partial images after the fourth level of the wavelet-transformation. And also the fourth level only contained data in a very few cases. This was not tested with patches larger than 32^2 pixels but the computation speed of implementations as used in this study can be enhanced when stopping feature computation after the third wavelet level. Table II includes an entry for a *first patch* method on 32^2 pixel gray-scale patches. This method was only implemented as an early placeholder for better founded methods and is therefore not listed in the results table.

VI. Conclusion

This paper is based on the wavelet standard-deviation descriptor [1] that uses texture images from the Brodatz texture database [15] containing 1856 samples. Their algorithm reaches an average classification rate of up to 70.30% when used in an image retrieval scenario. Compared to the extended classification approach in this paper the maximum task rate of 36.45% is to be rated rather moderate. The maximum feature rate of 42.12% also shows that a classification system (e.g. an automated image tagging system) can't solely be based on this algorithm. Nevertheless in 76.74% of the cases the

TABLE III
CLASSIFICATION RESULTS

colorspace & patch size	patch selection	feature aggregation	regions		feature rate			task rate		
			total	validated	1st	2nd	3rd	1st	2nd	3rd
GRAY max. squares	first patch	$\sum$	2960	all	identical to task rate			32.20	44.46	55.20
		$\prod$						23.41	31.76	41.39
YCrCb 8x8	maximum affiliation	$\sum$						19.06	31.38	35.94
		$\prod$						14.64	28.26	34.28
	mean value	$\sum$						**36.45**	**53.77**	61.88
		$\prod$						32.25	51.88	59.57
	largest difference	$\sum$						4.78	15.22	21.96
		$\prod$						5.36	16.32	22.75
YCrCb 32x32	maximum affiliation	$\sum$		2521	40.16	58.54	59.89	34.20	49.86	51.01
		$\prod$			6.13	14.04	25.61	5.22	11.96	21.81
	mean value	$\sum$			38.29	**62.28**	**76.74**	32.61	53.04	**65.36**
		$\prod$			5.27	16.51	25.01	4.49	14.06	21.30
	largest difference	$\sum$			23.65	41.94	57.60	20.14	35.72	49.06
		$\prod$			6.13	14.12	25.61	5.22	12.03	21.81
RGB 32x32	maximum affiliation	$\sum$			37.01	52.33	70.03	31.52	44.57	59.64
		$\prod$			8.68	15.31	26.97	7.39	13.04	22.97
	mean value	$\sum$			**42.12**	60.49	75.81	35.87	51.52	64.57
		$\prod$			7.91	16.25	26.21	6.74	13.84	22.32
	largest difference	$\sum$			29.61	45.44	58.87	25.22	38.70	50.14
		$\prod$			8.85	15.15	26.97	7.54	12.90	22.97
GRAY 32x32	maximum affiliation	$\sum$			35.91	51.81	68.83	30.58	44.13	58.62
		$\prod$			7.66	15.57	26.47	6.52	13.26	22.54
	mean value	$\sum$			39.57	59.39	75.04	33.70	50.58	63.91
		$\prod$			6.55	15.40	25.69	5.58	13.12	21.88
	largest difference	$\sum$			25.61	41.86	55.48	21.81	35.65	47.25
		$\prod$			7.49	15.49	26.29	6.38	13.19	22.39

- *feature rate* represents the classification performance in regard to all validated regions
- *task rate* represents the classification performance in regard to the total number of regions
- 1st / 2nd / 3rd ≙ searched concept is contained in best / best two / best three affiliation result(s)
- all classification rates are percentage values, bold text shows the column's maximum
- $\sum$, $\prod$ reference to equations 5 and 6

correct concept can be found within the best three affiliated concepts out of ten. This can be used as a weighting factor for algorithms dealing with similar tasks. The results show two major problems when using texture descriptors on segmented images of natural scenarios. First the shape or size of the segments can strongly counter-act the requirements of the texture descriptor - in case of wavelets this is the square-sized base shape. And second texture patterns in natural images underly a perspective distortion as you most likely don't watch top-down onto the surface (like in satellite photography). The extension of the original approach brings up one more issue. Instead of comparing two images directly, an image is compared to a trained descriptor representing a group of texture patterns. This affects the capabilities of the algorithm as it adds additional noise to the results. Considering those problems the results appear sufficient as base for upcoming studies and the presented knowledge gaps require further investigation. The following paragraphs propose approaches to be validated in order to improve the classification performance.

A. Segment WSD selection

The determination of a segment's WSD can be dealt with in very different ways. The grid-based process used in this paper (see section IV-B) was chosen to make sure the input for the wavelet-transformation has always the same resolution. Another aspect was a predictable computation duration. However this demands for a selection or aggregation of data as most likely many WSDs can be computed for a single segment. The larger the grid's patch size is compared to the segment size the more data is lost in the border regions of the segments. It also increases the possibility to find no single patch being completely located within the segment. The smaller the grid's patch size is chosen the less representing the data found in a single patch becomes. This also increases the computation time a lot as much more wavelet-transformations and comparisons to the concepts must be performed per segment. Upcoming studies can therefore focus on the problem of finding an optimal patch size depending on the image resolution, the segment size and shape.

The selection or creation of a representative patch currently favors the mean value methods presented in section III-B. A method yet unevaluated would be to count the appearances the patches' concept affiliations favor a specific concept within a single segment. Saying a segment consisting of 10 patches favors 2 times ”sand”, 3 times ”water” and 5 times ”snow”. Then you can select the ”snow”-patch with the highest affiliation value to the concept snow as representative. An extension to this idea would be to connect positioning values to a patch's concept affiliation values. Saying the highest value scores 10 points (when having 10 concepts), the next highest value scores 9 points and so on. In the end you sum up the concept affiliation scores and pick the concept with the highest score as representative.

Another Idea that was tested is the *maximum square* approach. This approach increases the computation speed a lot as only one feature vector has to be calculated as soon as the maximum square area was identified. Unfortunately the results show a worse performance than a grid sized approach on gray-scales.

An even more sophisticated approach would be to identify subregions within the image segments that are oriented almost planar to the camera's viewpoint. Those subregions provide data that is just slightly altered due to perspective distortion. You can also think of finding a segment's world plane in order to create a planar representation through perspective back projection (e.g. an ocean layer or a wall). The assumption here is that the distortion a perspective back projection creates is smaller than the information value generated in terms of WSD similarity.

B. Similarity Measure

The similarity measure presented in this paper is based on a probabilistic approach using mean and standard deviation to create a similarity value for each of the feature's levels. Finally the similarities of each level were either summed up or multiplied to create a value expressing the patch's affiliation to a concept.

In literature there are a lot more similarity measures offered for vector- and histogram-like comparisons. Implementations using (e.g.) the vector angle, a support vector machine, the earth mover's distance or the kullback-leibler divergence might improve the results a lot.

C. Usage of color values

The implementation as proposed in [1] currently only uses the standard deviation of each color channel as color information. This means only the structural character of the color is taken into account when creating a concept representation. A more promising approach would be to use the color intensities and a comparison method based on their mean value as an additional factor in the process of calculating a concept affiliation. This could be realized e.g. using the HSV or Lab color space. A WSD-conform vector on the S,V or L channels is created to represent the structure of a segment and additionally the H or a and b channels create a similar vector including the color mean values. This should at least improve distinguishing concepts like snow, sand or sea from each other.

References

[1] Sitaram Bhagavathy and Kapil Chhabra, “A wavelet-based image retrieval system,” University of California, Santa Barbara, Tech. Rep., an ECE 278A Project Report.

[2] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, and J. Malik, “Blobworld: a system for region-based image indexing and retrieval,” Berkeley, CA, USA, Tech. Rep., 1999. [Online]. Available: http://portal.acm.org/citation.cfm?id=893714

[3] T. Deselaers, D. Keysers, and H. Ney, “Features for image retrieval - a quantitative comparison,” in *In DAGM 2004, Pattern Recognition, 26th DAGM Symposium*, 2004, pp. 228–236.

[4] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, “Efficient and effective querying by image content,” *J. Intell. Inf. Syst.*, vol. 3, pp. 231–262, July 1994. [Online]. Available: http://dx.doi.org/10.1007/BF00962238

[5] D. ing Sven Siggelkow, D. Prof, D. T. Ottmann, P. Dr, T. Ottmann, B. Haasdonk, L. Bergen, O. Ronneberger, C. B. S. Utcke, and S. Siggelkow, “Feature histograms for content-based image retrieval,” 2002.

[6] C. Palm, D. Keysers, T. Lehmann, and K. Spitzer, “Gabor filtering of complex hue/saturation images for color texture classification,” 2000.

[7] H. Tamura, T. Mori, and T. Yamawaki, “Textural features corresponding to visual perception,” vol. 8, pp. 460–473, June 1978.

[8] T. Deselaers, “Features for image retrieval,” December 2003, diploma Thesis.

[9] J. Z. Wang, J. Li, and G. Wiederhold, “Simplicity: Semantics-sensitive integrated matching for picture libraries,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 947–963, 2001.

[10] B. Manjunath and W. Ma, “Browsing large satellite and aerial photographs,” 1996, pp. II: 765–768.

[11] ——, “Texture features for browsing and retrieval of image data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 837–842, 1996.

[12] J. Li and J. Z. Wang, “Automatic linguistic indexing of pictures by a statistical modeling approach,” *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 25, no. 9, September 2003.

[13] Michael Clemens, “Wavelet tutorial,” http://nt.eit.uni-kl.de/wavelet/dwt_2d.html (11. Mars 2010).

[14] A. Haar, “Zur theorie der orthogonalen funktionssysteme,” *Mathematische Annalen*, vol. 69, no. 3, pp. 331–371, 1910.

[15] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. Dover Publications, 1999.

ISBN 978-83-60810-27-9
ISSN 1896-7094

High Capacity Colored Two Dimensional Codes

Antonio Grillo*, Alessandro Lentini*, Marco Querini * and Giuseppe F. Italiano*
*Department of Computer Science, Systems and Production
University of ”Tor Vergata”
Via del Politecnico 1, 00133 Rome Italy
Email: grillo;lentini;italiano@disp.uniroma2.it

***Abstract*—Barcodes enable automated work processes without human intervention, and are widely deployed because they are fast and accurate, eliminate many errors and often save time and money. In order to increase the data capacity of barcodes, two dimensional (2D) code were developed; the main challenges of 2D codes lie in their need to store more information and more character types without compromising their practical efficiency. This paper proposes the High Capacity Colored Two Dimensional (HCC2D) code, a new 2D code which aims at increasing the space available for data, while preserving the strong reliability and robustness properties of QR. The use of colored modules in HCC2D poses some new and non-trivial computer vision challenges. We developed a prototype of HCC2D, which realizes the entire Print&Scan process. The performance of HCC2D was evaluated considering different operating scenarios and data densities. HCC2D was compared to other barcodes, such as QR and Microsoft's HCCB; the experiment results showed that HCC2D codes obtain data densities close to HCCB and strong robustness similar to QR.**

I. Introduction

Barcodes have become widely popular because of their reading speed, accuracy, and functional characteristics. As barcodes became popular and their convenience universally recognized, the market began to call for codes capable of storing more information, more character types, and that could be printed in smaller space. As a result, various efforts were made to increase the amount of information stored in barcodes, such as increasing the number of barcode digits or laying out multiple barcodes. However, these improvements have some negative effects, such as enlarged barcode areas, complicated reading operations, and increased printing costs. Barcodes may be referred to as linear or one-dimensional (1D) codes. However, barcodes are available also in patterns of square, dots, hexagons and other geometric patterns within the image; such a kind of barcodes are referred to as matrix or bidimensional (2D) codes. Although 2D systems use symbols other than bars, they are generally referred to as barcodes as well. In order to increase the available data space, 2D codes introduce the capability of storing information in two directions. 2D codes contains information in both the vertical and horizontal dimensions, whereas 1D codes contains data in one dimension only. Figure 1 shows examples of 2D (a) e 1D (b) codes.

Available 2D codes solutions span from repeating a single 1D code over multiple rows to exploiting bidimensional shapes to represents data. Figure 2 illustrates the evolution of barcode technology. In particular, Figure 2 (a) shows the multiple barcode layout: the main disadvantage related to this simple 2D layout is the need of multiple scans in order to get all the information contained in the barcode. Figure 2 (b) illustrates the stacked barcode layout: in this case one single scan is enough to obtain the stored information but the scanning equipment must be carefully aligned with the code orientation. Finally, in Figure 2 (c), the matrix barcode layout is presented: this layout enables to acquire information with one single scan and does not require the accurate alignment of the scanning equipment.

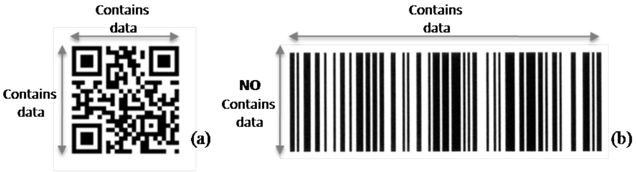


Fig. 1. Dimension for storing data in 2D (a) e 1D (b) codes

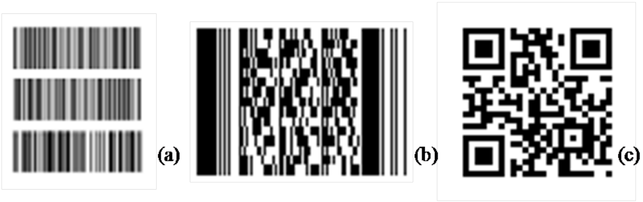


Fig. 2. Evolution of barcodes: multiple barcode layout(a), stacked barcode layout (b) and matrix barcode layout (c)

There are more than 20 types of conventional 2D codes. Figure 3 illustrates some examples of 2D codes; the main difference among the presented codes is in terms of the amount of data which can be stored in a single code. For example, Quick Response (QR)[1] code is able to store more than 7,000 decimal digits, while Maxi Code is able to store only 138 decimal digits. QR Code is a type of 2D codes developed by Denso Wave, a division of Denso Corporation at the time, and released in 1994 with the primary aim of being easily interpreted by scanner equipment.

The capability of storing more data in the same space taken by a classical two dimensional code represents one of the main challenges of the next generation barcodes. In order to increase

[1] QR Code is registered trademarks of Denso Wave Incorporated in Japan and other countries.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

	QR Code	PDF417	DataMatrix	Maxi Code
Developer *(country)*	DENSO (Japan)	Symbol Technologies (USA)	RVSI Acuity CiMatrix (USA)	UPS (USA)
Code Type	Matrix	Stacked Bar Code	Matrix	Matrix
Numeric Data	7,089	2,710	3,116	138
Alphanumeric Data	4,296	1,850	2,355	93
Binary Data	2,953	1,018	1,556	
Kanji Data	1,817	554	778	
Main features	Large capacity, Small printout size, High speed scan	Large capacity	Small printout size	High speed scan
Standardization	AIM International, JIS, ISO	AIM International, ISO	AIM International, ISO	AIM International, ISO

Fig. 3. Characteristics of some types of 2D codes

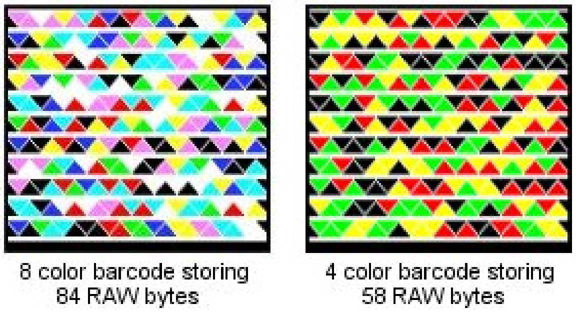


Fig. 4. An example of the Microsoft High Capacity Color Barcode (HCCB) (Viewed better in color)

data capacity, some 2D barcodes use colors to create more symbols, resulting in larger data capacity within the same size. Examples of such barcodes are the Color Bar Code System of Imageid Ltd [1] and the more widely diffused Microsoft's High Capacity Color Barcode (HCCB) [2], [3], [4]. While HCCB may be used for a variety of applications, its most immediate application is for marking univoquely commercial medias such as motion pictures, video games, broadcasts, digital video recordings, etc... We now describe briefly the main features of HCCB (see Figure 4). It consists of rows of strings of symbols (triangles) of four different colors: black, red, green and yellow, and consecutive rows are separated by a white line. While the number of rows in a HCCB code may vary, the number of modules in each row is always a multiple of the number of rows. A module represents the basic entity for storing information in a 2D code. HCCB has a black boundary around it, further surrounded by a thick white band. These patterns are designed to act as visual landmarks in order to locate the barcode in an image. The black boundary at the bottom of HCCB is thicker than the boundaries on the other three sides: the bottom boundary acts as an orientation landmark, as barcodes may be at an arbitrary orientation in the image. The last 8 symbols on the last row are always in the fixed order of black, red, green and yellow (2 symbols per color) and can be used as a color palette during the scan. The main limitation of the HCCB code is related to the fragility of the detection and alignment mechanisms. Indeed, the detection process works as follows: it starts from a point which is supposed to be at the interior of the code and proceeds on squares of larger sizes until it recognizes the white border around the code; after the white border has been located, it starts the alignment process by looking for the thick bottom boundary. The fragility of the detection process derives from the fact that not all the images inside a white border are necessarily codes (thus giving rise to delayed failures, which will be explained later), while the weakness of the alignment process derives from the facts that different slopes in the scan phase might result in failures to properly recognize the thick bottom boundary [5].

The increased data density obtained with the usage of colors comes at an additional cost. Today a Print&Scan process is commonly used for image reproduction and distribution. Indeed, often images are converted between printed and digital formats. A rescanned image may look similar to the original, but it may have been distorted during the process. Indeed, reading 2D color codes poses significant computer vision challenges [5], [6], [7]. This is due to several factors, and we cite only few of them in the following. First, the color balance may be drastically different in different code readers. Second, the images containing codes may be taken by unexperienced users, and thus the location of the barcode in the image, its orientation, its slope, etc. can be mostly unconstrained. Furthermore, possible transformations in the prospective can distort the geometry of the barcode. Last but not least, the light conditions under which the images are taken can vary dramatically.

Not all the scenarios where black and white 2D codes are currently exploited may benefit from the introduction of colors; in many scenarios 2D codes have to be copied or transmitted through fax, and fast color printers and color fax machines are not yet widely diffused in today's offices. On the other hand, the availability of new low cost hardware may solve some of the problems that arise in the Print&Scan process. Since in many cases the Print operation is executed once while the Scan operation is likely to be repeated many times, we can consider that replacing ad-hoc hardware for scanning 2D codes by inexpensive mobile phones equipped with a megapixel camera may dramatically boost the adoption of colored 2D codes.

The contribution of this work is a new 2D code technology, named HCC2D (High Capacity Colored Two Dimensional), which use colors to increase the code data density. The introduction and recognition of colored modules in HCC2D poses some new and non-trivial computer vision challenges, such as handling the color distortions introduce by the hardware equipment that realizes the Print&Scan process. The HCC2D codes presented in this paper are able to support different types and sizes of data input, and adapt smoothly the code dimension to the actual input size. In order to support all those scenarios in which the Print&Scan process imposes the usage of only two colors (i.e., Black&White) HCC2D considers Black&White codes as codes with exactly two colors. In particular, HCC2D has been designed so as to be fully compatible with the standard QR code, which is currently the most widespread 2D code technology (a standard QR code represents the simplest case of our 2D colored code). The main advantage of HCC2D over QR is that HCC2D is able to store substantially more data than QR, while preserving the strong reliability and robustness properties of QR.

We developed a prototype of HCC2D, which realizes the entire Print&Scan process, tested this prototype in many experiments considering different operating scenarios and data densities, and compared it to QR and HCCB. In our experiments, HCC2D codes obtained data densities close to HCCB and strong robustness similar to QR. In particular, HCC2D resulted to be very robust, and capable of resisting to dirt, damage and distortion, and introduced little computational overheads compared to QR.

II. Standard QR Code

Standard QR codes as well as other black and white 2D codes store data using a graphical representation; the core of this representation is based on the arrangement of multiple simple geometric shapes over a fixed space. A generic 2D code is required to perform efficiently at least the following three functions:

- the position detection function is critical; elements that serve as position detection function give to the acquisition process the capability of identifying the presence of a 2D Code in the acquired image;
- the alignment function is required to synchronize the Scan process on the right position of a 2D code. This function exploits some alignment patterns placed by the Print process in a well known position. Hence, the Scan process focuses on retrieving these well known patterns in order to position correctly the 2D code.
- the data function is required to encode the input data in a specific graphical representation. Some additional goals may be reached by the data function; error correction and data masking are examples of these functions for strengthening the 2D code.

In particular standard QR codes adopt an arrangement of black and white squares of different sizes for all the required functions. In the following, some of the features provided by QR codes will be discussed.

A. High Capacity Encoding of Data

While conventional 1D codes store up to 20 decimal digits, QR Code is able to store from several dozen to several hundred times more information. QR Code can handle a large variety of data, such as binary, numeric and alphabetic characters, Kanji, Kana and Hiragana (Japanese) symbols, and control codes. If the input is represented by decimal digits, one code can encode up to 7,089 decimal digits (see Figure 3).

B. Small Printout Size

Since QR code carries information both horizontally and vertically, it is capable of encoding the same amount of data in approximately one-tenth the space of a traditional 1D code.

C. Dirt, Damage and Distortion Resistant

QR code has error correction capability. Data can be restored even if the symbol is partially dirty and damaged (see Figure 5). A maximum of 30% of the codewords can be restored. A codeword is a unit that constructs the data area. In the case of QR code, one codeword is equal to 8 bits. Thanks to its alignment function, QR Code is resistant to distorted acquisitions.

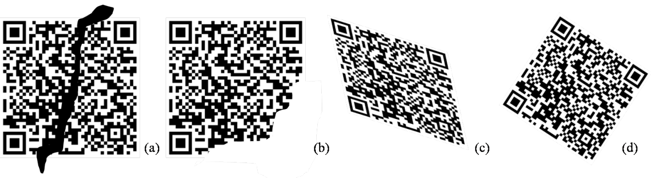


Fig. 5. Samples of QR Code dirty (a), damaged (b), distorted (c), and rotated(d).

D. Readable from any direction in 360 degrees

QR code is capable of 360 degrees (omni-directional) reading. This task is accomplished through position detection patterns located at the three corners of the symbol. These position detection patterns guarantee stable high-speed reading, circumventing the negative effects of background interference.

E. Structured Append Feature

If a single QR code is too large for the print space available, a splitting function may be applied to obtain smaller QR codes containing the same data. One data symbol can be divided into up to 16 codes, which can be printed in smaller areas (see Figure 6).

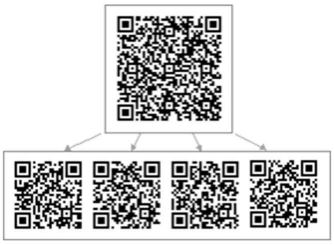


Fig. 6. Single QR Code splitted in four smaller QR Code

F. Standardization Process

QR codes have become widely used for two reasons: QR code specifications are clearly defined and made public and the QR codes can be freely usable. QR code is open in the sense that the specification of QR code is disclosed and that the patent right owned by Denso Wave is not exercised.

III. High Capacity Colored QR Code

The main goals of our HCC2D code are to increase the space available for data and to preserve strong robustness and error correction properties similar to the original QR standard. HCC2D increases the storage space by generating each module of the data area with a color selected from a color palette. Figure 7 illustrates samples of HCC2D with 4 colors HCC2D (Figure 7 (a)) and 16 colors (Figure 7 (b)).

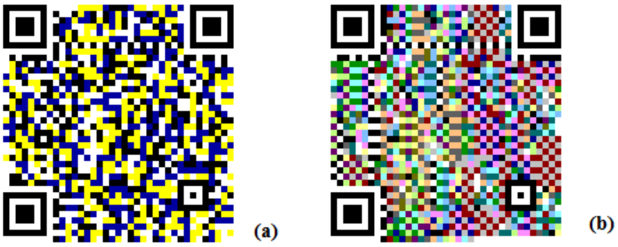


Fig. 7. Samples of the High Capacity Colored Two Dimensional Code (HCC2D) version 5 and error correction level H: 4 colors (a) and 16 colors (b) (Viewed better in color).

In the standard QR code each module represents a single bit following a simple rule: black squares store 1 and white squares store 0. To ensure robustness, we have designed similar mechanisms to those available in the standard QR code; in particular, we have applied the Reed Solomon error correction codes for correcting code modules that represent more than one single bit. This way, HCC2D defines a superset of the standard QR code set, and thus it is able to maintain fully compatibility with QR.

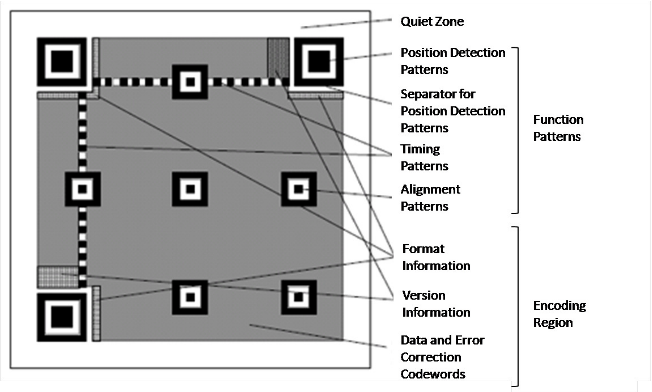


Fig. 8. Structure of a generic QR Code

A standard QR Code can be represented as shown in Figure 8; the structure is composed of some elements that perform the various functions. The available space in each symbol may serve as *Function Patterns* or as *Encoding Region*. The *Position Detection Patterns*, the *Alignment Patterns*, the *Timing Patterns*, and the *Separators for Position Detection Patterns* support the Scan process in detecting the presence, the right orientation and the correct slope of a QR code into an image. The *Format Information* describes the error correction level used in the code; it is possible to use four different error correction levels in the HCC2D code: the lower level (i.e., Level L) is able to correct about 7% of the data, the Level M restores about 15% of the data, the Level Q is able to fix about 25% of the data, and the higher level (i.e., Level H) corrects approximately 30% of the data. The *Version Information* contains the real size of the code; it is possible to generate QR codes starting from Version 1 (i.e., 21x21 modules) to Version 40 (i.e., 177x177 modules). Finally the *Data and Error Correction Codewords* contains input and error correction data.

We designed the HCC2D code preserving all the *Function Patterns*, the *Format Information* and the *Version Information* defined in the standard QR code. Saving the structure and position of such critical information allows HCC2D code to preserve compatibility with the standard QR code. Furthermore, the space required by all this information is small, so we did not reduce this space to increase the data density. Any modification to such information may led to failures in the recognition process. The most important changes are gathered in the Data and Error Correction Codewords area. The most noticeable difference with a standard QR code is that the modules may be of different colors; in a code with a palette composed by at least 4 colors each module is able to store more than one bit. Introducing colors in the data and error correction area requires to address some issues, which will be analyzed next.

A. *HCC2D Code Tables*

The HCC2D Code Tables contain some information such as the total codewords count, the symbol version, the error correction level, the Reed-Solomon block type, etc. The aim of these tables is to support users in selecting the best code once the size and kind of the input data and the desired error correction level are known. In order to define the table it is possible to refer to those published in the ISO/IEC 18004 document that contains the definition of the standard QR code. The Bits per Module (BpM) can be defined as the number of bits that a single module is able to store:

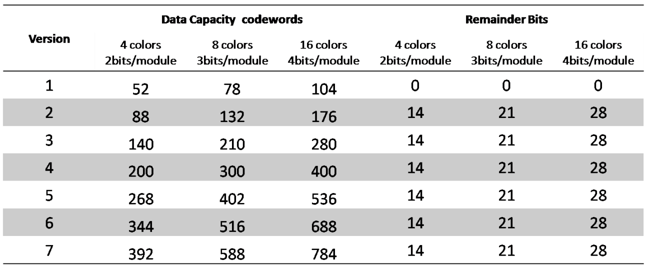
$$BpM = \log_2(\text{number of colors})$$

The more colors available, the more data can be stored into the code. In Figure 9 the data capacity for the smaller versions of QR Code are detailed.

Version	No. Of Modules/Side	Function Patterns Modules	Format and Version Information Modules	Data Modules	Data Capacity (codewords)	Remainder Bits
1	21	202	31	208	26	0
2	25	235	31	359	44	7
3	29	243	31	567	70	7
4	33	251	31	807	100	7
5	37	259	31	1079	134	7
6	41	267	31	1383	172	7
7	45	390	67	1568	196	0

Fig. 9. Data capacity for smaller version of standard QR Code

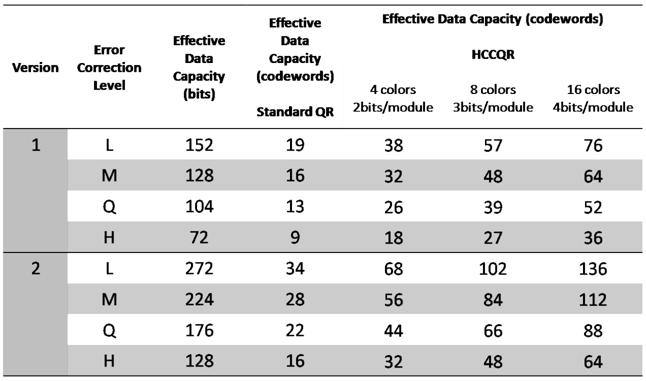
When considering values for BpM greater than one, values contained in the Data Capacity column vary. In particular, since each module is able to store exactly BpM bits each value has to be multiplied by the BpM value as shown in Figure 10. The *Remainder Bits* are bits used to fill empty positions of the symbol encoding region after the final symbol character in the standard QR codes. They follow the same simple rules: the new value of *Remainder Bits* is obtained by multiplying the old value by the BpM value.



Version	Data Capacity codewords			Remainder Bits		
	4 colors 2bits/module	8 colors 3bits/module	16 colors 4bits/module	4 colors 2bits/module	8 colors 3bits/module	16 colors 4bits/module
1	52	78	104	0	0	0
2	88	132	176	14	21	28
3	140	210	280	14	21	28
4	200	300	400	14	21	28
5	268	402	536	14	21	28
6	344	516	688	14	21	28
7	392	588	784	14	21	28

Fig. 10. New values for data capacity for smaller version of HCC2D codes

Figure 11 illustrates the effective data capacity for version 1 and version 2 of QR and HCC2D when considering the use of a specific error correction level. The new effective capacity is obtained by multiplying the old values by the BpM value. Starting from these values and choosing the data type that will be encoded in the HCC2D code we can compute the real code capacity.

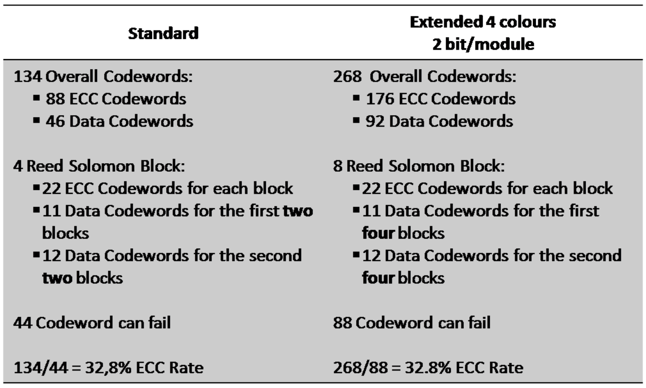


Version	Error Correction Level	Effective Data Capacity (bits)	Effective Data Capacity (codewords)	Effective Data Capacity (codewords)		
				HCCQR		
			Standard QR	4 colors 2bits/module	8 colors 3bits/module	16 colors 4bits/module
1	L	152	19	38	57	76
	M	128	16	32	48	64
	Q	104	13	26	39	52
	H	72	9	18	27	36
2	L	272	34	68	102	136
	M	224	28	56	84	112
	Q	176	22	44	66	88
	H	128	16	32	48	64

Fig. 11. Error Correction Level for version 1 and version 2 of standard QR codes and HCC2D codes

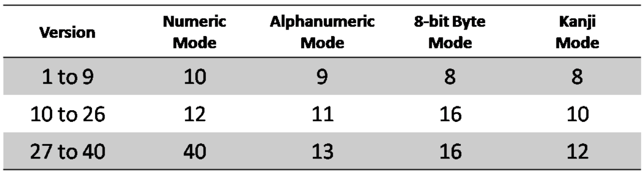
Figure 12 illustrates the situation for version 5 of HCC2D with the highest error correction level (i.e., level H). The *Character Count Indicator* value gives the number of elements of a specific data type that are encoded in the code. Conversely, the *Mode Parameter* expresses the specific data type contained in the code. Since the main goal of HCC2D is to increase the number of elements that can be encoded in a single symbol, it is important to verify if this value fits in the space reserved by the standard.

Note that QR code version from 1 (i.e., 21x21 modules) to 9 (i.e., 53x53 modules) in alphanumeric mode reserves 9 bits for the *Character Count Indicator*, thus allowing for at most 511 (i.e., $2^9 - 1$) characters. However, with the increase of data density, HCC2D codes may contain more than 511 elements, and thus 9 bits are no longer sufficient. Hence, the space reserved for the *Character Count Indicator* in a HCC2D code is defined according the following rules:



Standard	Extended 4 colours 2 bit/module
134 Overall Codewords: ▪ 88 ECC Codewords ▪ 46 Data Codewords	268 Overall Codewords: ▪ 176 ECC Codewords ▪ 92 Data Codewords
4 Reed Solomon Block: ▪ 22 ECC Codewords for each block ▪ 11 Data Codewords for the first **two** blocks ▪ 12 Data Codewords for the second **two** blocks	8 Reed Solomon Block: ▪ 22 ECC Codewords for each block ▪ 11 Data Codewords for the first **four** blocks ▪ 12 Data Codewords for the second **four** blocks
44 Codeword can fail	88 Codeword can fail
134/44 = 32,8% ECC Rate	268/88 = 32.8% ECC Rate

Fig. 12. Comparison between standard QR code and HCC2D code considering version 5 and error correction level H



Version	Numeric Mode	Alphanumeric Mode	8-bit Byte Mode	Kanji Mode
1 to 9	10	9	8	8
10 to 26	12	11	16	10
27 to 40	40	13	16	12

Fig. 13. Number of bits reserved for the *Character Count Indicator* for the various standard QR Code version

- 16 bit are reserved in the 8-bit Byte mode, regardless of the HCC2D Code version;
- for the remaining modes, a simple rule that combine the old *Character Count Indicator* length, $Length_{old}$, and the bit per module, BpM value is applied:
  - $Length_{old} + (BpM/2)$ if the number of bits per module is even
  - $Length_{old} + ((BpM + 1)/2)$ if the number of bits per module is odd

Those rules are valid if and only if the color palette is composed by at least four colors; otherwise, the space for the *Character Count Indicator* can be defined as in QR codes (see Figure 13).

B. The Color Palette

If the *Encoding Region* is composed of colored modules, the Scan process needs to know the complete color palette in order to decode the symbol. In the standard QR code during the Scan process only the brightness information is taken in account. A simple solution is to consider the color palette as an *a priori* shared knowledge between the Print and the Scan processes; in such a scenario handling the distortions introduced by the specific hardware (e.g., scanner, camera, ...) represents a critical issue. Hence, in the HCC2D code we have introduced an additional field (i.e., the color palette) to ensure that the Scan process is able to know how many and which colors are used in the scanned code. Encoding the color palette directly in the HCC2D code helps in reducing failures in the acquisition process due to the color distortions related to the specific hardware.

In image processing, it is important to define some quick failure criteria that avoid unnecessary computations if the Scan process delays a successful recognition. Forcing the start of a new Scan process instead of trying to recognize low quality images can reduce the delayed failure. Sorting the colors that compose the color palette according to the brightness value may represent a simple criterion to overcome the delayed failure problem. The Scan process starts recognizing only the color palette, if colors in the palette are not sorted as expected the process terminate with a quick failure, otherwise the Scan process can continue.

Furthermore, if the color palette is replicated around the bidimensional code, the quick failure rate can be further reduced. Each color palette that does not respect the expected color sorting should be discarded. If a minimum number of color palettes is successfully recognized, the Scan process can build the color palette for decoding the code by computing the average of the valid color palettes. In HCC2D the repeated color palettes are placed on the right side of the code; this single line will be adjacent to the *Encoding Region* for preserving compatibility. Considering version 1 of HCC2D code we introduce an overhead of about 4,76 %, i.e., one line of 21 modules over 441 modules. The overhead is reduced to 0,56% for version 40, i.e., one line of 177 modules over 31329 modules.

Analyzing the color palette extra field in the HCC2D code, the Scan process is able to build a model for evaluating each module in the *Encoding Region*. In the standard QR code the problem of distinguishing between light and dark modules is addressed by handling only the brightness information; exploiting the *Quiet Zone* to calibrate the Scan process, an appropriate threshold is chosen. In HCC2D a more complex similarity function is needed for handling colored modules. Since colors in computer graphics are usually represented as vectors in multidimensional space (e.g., the Red Green Blue model, the Cyan Yellow Magenta Key model, the Hue Saturation Lightness model, etc.), we solve the identification problem by using Euclidean distances between such vectors. In particular, modules in the encoding region are considered as vectors according to the specific color model used by the scanning equipment; the color recognized is given by the color in the palette which minimizes the vector distance.

C. Data Masking

The main purpose of data masking in the standard QR code is to reach an appropriate balance between dark and light areas and to avoid that patterns similar to those used for *Position Detection*, *Alignment* and *Timing* appear in the *Encoding Region*. The HCC2D needs a similar function for preserving the Scan process in recognizing the *Function Patterns* and increasing the code robustness. The standard QR code defines eight different masks; each mask is generated according to a simple rule that mimics the *Function Patterns*. The Print process has to select the mask that obtains the best score according to a scoring rule that is based on some bitwise XOR operations. Switching the brightness of some modules (i.e., light modules become dark modules and vice versa) is the result of data masking.

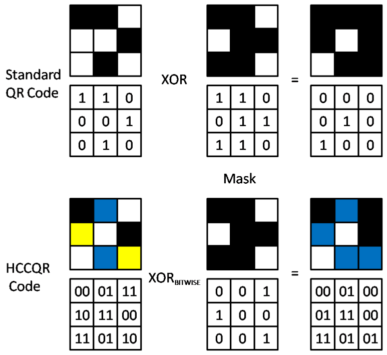


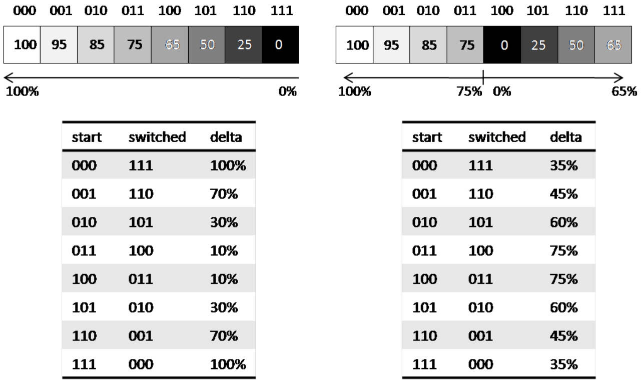
Fig. 14. Sample application of data masking in standard QR code and HCC2D code

Since HCC2D increases the number of bits per module (BpM), the standard scoring rules are no longer useful and we need to define new bitwise operations that are able to handle more than one bit per module. As far as the mask selection step is concerned, the HCC2D code is considered as a binary matrix composed simply of dark (i.e., the darkest colors) and light (i.e., the lightest colors) modules; the score for each mask is computed as in the standard QR code without taking in account the chromatic information of modules which is disjoint by the brightness information. Once the best mask is identified, each bit of the mask has to be used for switching the color of the module in the *Encoding Region*. For each module a bitwise operation between the bit in the mask and each bit that is stored in the module has to be performed. Figure14 shows an example of how the data masking process works for a 3x3 modules grid with 4 colors, considering both the standard QR code and the HCC2D code.

To preserve the balance between dark and light areas, we need to properly organize the palette. The colors are sorted in descending order according to the brightness information, the lighter colors first and the darker colors last; the color palette is then splitted in two halves and the second part is sorted in reverse order. As shown in Figure 15, this simple reorganization of colors in the palette results in increasing the minimum brightness distance between switched colors from 10% up to 35%.

IV. Experimental Results

We developed a prototype that implements the Print&Scan process for HCC2D codes. In particular, we implemented two different applications, the encoder and the decoder, for generating and acquiring HCC2D codes. The HCC2D code encoder was realized with the help of *libqrencode* [8], a C library for encoding data in standard QR code symbols, while the decoder was built with the help of *zxing* [9], an opensource Java project for improving the processing of 1D/2D barcodes. In our implementation, the decoder is



start	switched	delta
000	111	100%
001	110	70%
010	101	30%
011	100	10%
100	011	10%
101	010	30%
110	001	70%
111	000	100%

start	switched	delta
000	111	35%
001	110	45%
010	101	60%
011	100	75%
100	011	75%
101	010	60%
110	001	45%
111	000	35%

Fig. 15. Color palette organization for data masking

able to recognize standard QR codes as well; some of the operations executed in the acquisition process are preserved: recognizing the position detection pattern, recognizing and exploiting the alignment patterns, and reading the version and format information. After all these operations have been carried out, the decoder tries to detect the color palette. If the color palette is succesfully detected, the decoder tries to process a HCC2D code. Otherwise, it tries to decode a standard QR code. The image processing phase ends by returning a matrix representation of the scanned code; in the standard QR code a bitmatrix is sufficient to represent the modules whose brightness is greater (i.e., bit 0) or lower (i.e., bit 1) than the threshold value determined by analyzing the *Quiet Zone*. In HCC2D a bitmatrix is not adequate to represent the chromatic information of the modules. The acquisition process ends with a matrix of vectors that describe the scanned modules according to the color model used by the scanning equipment. In the HCC2D code design data are unmasked using the brightness information only; each scanned module is represented as a vector in the color model: a color distance rule is applied, the most similar (i.e., the closest) color in the palette is found and the associated bitstream is updated. Once the complete bitstream is reconstructed, the Reed-Solomon correction may be executed and the decoding process can retrieve the original input.

We now turn to the experimental results. We executed some performance tests using our prototype. We aimed at measuring the increase of the space available for data and the time computational overhead introduced by the use of colors, and thus we measured the time needed to convert all modules in a binary representation according to the color palette considered. Finally we reproduced the Print&Scan process for different code versions and different print qualities using widely diffused and low-cost print and scan equipment.

The first issue we address is the increase of data density, and a general comparison is illustrated in Table I. Like most 2D codes, the QR Code data-density depends on several factors. The module size depends heavily on the printing and scanning resolutions. Microsoft offers its HCCB in black and white, with four and eight different colors. Currently Microsoft laboratory tests have yielded using eight colors, 16,000 bits per square inch in its highest density form using a 600dpi business card scanner (cfr. [2]). Conversely, if a standard QR code symbol is printed with a resolution of 600 dpi, 4-dot printer, the module size is 0.17mm and will therefore require a scanner resolution of less than 0.17 mm (cfr. [10]). Using Version 19 of the standard QR code and the M correction level (i.e. about 15%) it is possibile to store 5,016 bits per square inch. Introducing a color palette composed of 8 colors, a HCC2D code of Version 19 and with the M correction level, is able to store 15,048 bits per square inch. HCC2D data density is slightly lower than Microsoft's HCCB data density but while our solution preserves similar robustness in detection, alignment, and error correction of the standard QR code HCCB has no patterns that strongly support the detection and alignment process. Our solution increases the data density of standard QR code by a factor proportional to the BpM as mentioned in Section III-A and reaches a similar capacity to Microsoft HCCB.

TABLE I
DENSITY OF BARCODE SYMBOLOGIES AT 600 DPI. THE DATA FOR HCCB IS TAKEN FROM [2]

Barcode Type	Data Density [KB in square inch]
QR Code	0.627
HCCB	2.0
HCC2D	1.881

TABLE II
RESULTS OF THE SCAN PROCESS TIME IN MSEC FOR QR AND HCC2D

Version	QR	HCC2D 4 color	HCC2D 16 color
1L	122	132 (7.57%)	135 (9.62%)
1M	123	136 (9.55%)	138 (10.87%)
1Q	129	133 (3.01%)	135 (4.45%)
1H	131	135 (2.97%)	136 (3.68%)
5L	143	171 (16.38%)	206 (30.59%)
5M	151	174 (13.21%)	208 (27.40%)
5Q	163	181 (9.95%)	208 (21.63%)
5H	161	182 (11.54%)	209 (22.97%)
10L	176	209 (15.79%)	246 (28.45%)
10M	188	208 (9.61%)	249 (24.49%)
10Q	189	210 (10.0%)	273 (30.77%)
10H	190	212 (10.37%)	282 (32.62%)
20L	240	349 (31.23%)	387 (37.98%)
20M	253	334 (24.25%)	398 (36.43%)
20Q	250	337 (25.81%)	389 (35.73%)
20H	257	323 (20.43%)	395 (34.93%)
30L	333	447 (25.50%)	474 (29.75%)
30M	350	430 (18.60%)	460 (23.91%)
30Q	347	437 (20.59%)	478 (27.40%)
30H	338	416 (18.75%)	506 (33.20%)
40L	430	483 (10.97%)	550 (21.81%)
40M	415	481 (13.72%)	540 (23.15%)
40Q	420	500 (16.0%)	566 (25.79%)
40H	373	485 (23.09%)	552 (32.42%)

Afterwards, we address the computational overhead for processing HCC2D codes by comparing the average time taken by the scan process. As far as standard QR codes are concerned, the time required for the automatic recognition of a code may be considered as a measure for the Scan process. Selection of a proper binarization method is critical to the performance of barcode recognition system. Binarization of gray scale images is the first and important step to be carried out in pre-processing system. In binarizing an image, a simple and popular method is thresholding. Sahoo et al. [11] concluded that, among more than 20 thresholding methods, the Otsus method [12] which chooses the threshold that minimizes within-group variance, gives better results. A goal-directed performance evaluation of eleven popular locally adaptive thresholding algorithms were performed in [13] for map images. The experimental results indicated that Niblacks method with postprocessing step appears to be the best. Since our solution preserves the binarization for recognizing position detection and alignment patterns, the results mentioned in [11], [12], [13] are still valid.

In Table II we report results of an experiment focused on the recognition of more than 100 barcode images. To measure the overhead introduced by colors, times were taken after the alignment and detection phase. The experiment was run on a machine equipped with a Linux Slackware 13.0 operating system running on 1.73 GHz Intel dual core with 1 GB of RAM. The results report the average time in milliseconds for QR codes, HCC2D with 4 colors and HCC2D with 16 colors, and the overhead (in parenthesis, percentual values) introduced by HCC2D over QR. Our experiments show that, although the overhead introduced by HCC2D over QR tends to increase, as expected, with the number of colors and the code size, it seems to remain always within reasonable values (ranging from a minimum of about 3% for HCC2D version 1H with 4 colors to a maximum of about 38% for HCC2D version 20L with 16 colors). In particular, the average overhead introduced by HCC2D with 4 colors is about 15%, while the average overhead introduced by HCC2D with 16 colors is about 25%. The dependendence of the overhead on the error correction levels appears to be more complicated: in HCC2D with 4 colors the overhead (for the same code size) tends to decrease with the error correction levels for all versions except for version 40, while our experiments did not show a very strong correlation between the overhead of HCC2D with 16 colors and the error correction levels. In any case, the slowdown implied by the use of higher levels of error correction appears always limited (within about 5% of the total time).

Finally, in order to evaluate the usability of HCC2D code in a real setting, we realized a common Print&Scan scenario using a widespread inkjet multifunction equipment with print and scan capabilities. The equipment is able to print and to scan at different resolutions; in the test scenario we decide to fix the scan resolution while varying the print resolution. We have considered four different print resolutions: Draft Mode (i.e., 180 dpi), Text Mode (i.e., 360 dpi), Text and Photo Mode (i.e., 720 dpi), Text and Photo Mode (i.e., 720 dpi) and Photo (i.e., 1440 dpi). We experimented with a small print size (1 square inch) for different versions of the 4-colors HCC2D codes with the L error correction level; the Print&Scan process is repeated for the code versions: Version 5 (i.e., 37x37 modules), Version 10 (i.e., 57x57 modules) and Version 15 (i.e., 77x77 modules). Table III shows that only codes printed at poor quality levels (i.e., Draft Mode with a resolution of 180 dpi) fail to complete successfully the Print&Scan process. When the print resolution increases, the success rate depends on the HCC2D version: at 360 dpi only Version 15 (77x77 modules) failed on print sizes of 1 square inch.

TABLE III
USABILITY OF DIFFERENT VERSIONS OF THE HCC2D CODE VARYING THE PRINT RESOLUTION

Print resolution	180dpi	360dpi	720dpi	1440dpi
Version 5	No	Yes	Yes	Yes
Version 10	No	Yes	Yes	Yes
Version 15	No	No	Yes	Yes

V. CONCLUSIONS

In this paper we have proposed High Capacity Colored QR codes, a new 2D code which aims at increasing the space available for data, while preserving similar robustness, error correction and without loosing compatibility with the original QR standard. Our results show that HCC2D leads to larger data density compared to QR at the price of a small computational overhead. Though the data density is slightly lower than in HCCB, HCC2D does not suffer from the problems in detection and alignment of the 2D code.

REFERENCES

[1] E. Sali and D. M. Lax, "Color bar code system," US Patent 7210631, February 2006.

[2] "High capacity color barcodes," http://research.microsoft.com/en-us/projects/hccb/, Microsoft Research, May 2010.

[3] T. Bishop, "Software notebook: Color is key to Microsoft's next-generation bar code," http://www.seattlepi.com/business, April 2007.

[4] I. Fried, "Microsoft gives bar codes a splash of color," http://news.cnet.com, April 2007.

[5] D. Parikh and G. Jancke, "Localization and segmentation of a 2d high capacity color barcode," in *Proceedings of the 2008 IEEE Workshop on Applications of Computer Vision*. IEEE Computer Society, January 2008, pp. 1–6.

[6] O. Bulan, V. Monga, and G. Sharma, "High capacity color barcodes using dot orientation and color separability," in *Proceedings of Media Forensics and Security*. SPIE, January 2009.

[7] K. O. Siong, D. Chai, and K. T. Tan, "The use of border in colour 2d barcode," in *2008 International Symposium on Parallel and Distributed Processing with Applications (ISPA'2008)*, December 2008, pp. 999–1005.

[8] K. Fukuchi, "Libqrencode, a c library for encoding data in a qr code symbol," http://megaui.net/fukuchi/works/qrencode/, October 2010.

[9] "Zxing, multi-format 1d/2d barcode image processing library," http://code.google.com/p/zxing/, Google Inc., May 2010.

[10] "Quick response code - printer head density and module size," http://www.denso-wave.com/qrcode/qrgene3-e.html, Denso Wave, May 2010.

[11] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Y. Chen, "A survey of thresholding techniques," *Computer Vision, Graphics, and Image Processing*, vol. 41, pp. 233–260, February 1988.

[12] I. J. Kim, "Multi-window binarization of camera image for document recognition," in *Proceedings of the 9th International Workshop on Frontiers in Handwriting Recognition*. IEEE Computer Society, October 2004, pp. 323–327.

[13] O. D. Trier and A. K. Jain, "Goal-directed evaluation of binarization methods," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17. IEEE Computer Society, December 1995, pp. 1191–1201.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Region-based Measures for Evaluation of Color Image Segmentation

Andreea Iancu, Bogdan Popescu, Marius Brezovan and Eugen Ganea
University of Craiova
Software Engineering Department
Craiova, Bd. Decebal 107, Romania
{andreea.iancu, bogdan.popescu}@itsix.com, {brezovan_marius, ganea_eugen}@software.ucv.ro

***Abstract*—The present paper is aimed to compare the efficiency of a new segmentation method with several existing approaches. The paper addresses the problem of image segmentation evaluation from the error measurement point of view.We are introducing a new method of salient object recognition with very good results relative to other already known object detection methods. We developed a simple evaluation framework in order to compare the results of our method with other segmentation methods. The experimental results offer a complete basis for parallel analysis with respect to the precision of our algorithm, rather than the individual efficiency.**

I. Introduction

The problem of segmentation is an important research field and many segmentation methods have been proposed in the literature so far ([1],[3],[4],[7]). The aim of image segmentation is the domain-independent partition of the image into a set of regions which are visually distinct and uniform with respect to some property, such as grey level, texture or color.

Image segmentation is one of the most important operations performed on acquired images. Image segmentation evaluation [2] focuses on two main properties: objectivity and generality. Objectivity means that all the test images in the benchmark should have an unambiguous ground-truth segmentation so that the evaluation can be conducted objectively. Generality means that the test images in the benchmark should have a large variety so that the evaluation results can be extended to other images and applications.

The main objective of this paper is to emphasize the very good results of image segmentation obtained by our segmentation technique, $Graph-Based\ Salient\ Object\ Detection$, and to compare them with other existing methods. The algorithms that we use for comparison are: $Normalized\ Cuts$, $Efficient\ Graph-Based\ Image\ Segmentation$ $(Local\ Variation)$ and $Mean\ Shift$. All of them are complex and well known algorithms, with very good results in this area and building the knowledge based on their results represents a solid reference.

The experiments were completed using the images and ground-truth segmentations in the Berkeley segmentation dataset [8]. Since the ground-truth segmentation may not be well and uniquely defined, each test image in the Berkeley benchmark is manually segmented by a group of people.

The segmentation accuracy is measured taken into consideration the global consistency error and the local consistency error. We will provide comparative results that reflect a well-balanced behavior of the algorithm we propose.

The paper is organized as follows. In Section III we briefly present previous studies in the domain of image segmentation and the segmentation method we propose. The methodology of performance evaluation is presented in Section IV. The experimental results are presented in Section V. Section VI concludes the paper and outlines the main directions of the future work.

II. Related Work

Image segmentation evaluation is an open subject in today's image processing field. The goal of existing studies is to establish the accuracy of each individual approach and find new improvement methods. The segmentation methods require ground truth image segmentations as reference. The main drawback of providing such reference is represented by the resources that are needed. However, after analyzing the differences between the image under study and the ground truth segmentation, a performance proof is obtained.

Region-based segmentation methods can be broadly classified as either model-based [13] or visual feature-based [14] approaches. A distinct category of region-based segmentation methods that is relevant to our approach is represented by graph-based segmentation methods. Most graph-based segmentation methods attempt to search a certain structures in the associated edge weighted graph constructed on the image pixels, such as minimum spanning tree [3], or minimum cut [15].

Berkeley image segmentation benchmark is the reference that we use for our study. Using the same input, the provided image dataset, we are developing a customized methodology in order to efficiently evaluate our algorithm.

The closest work to ours is [3], in which an image segmentation is produced by creating a forest of minimum spanning trees of the connected components of the associated weighted graph of the image. The novelty of our contribution concerns two main aspects: (a) in order to minimize the running time we construct a hexagonal structure based on the image pixels, that is used in both color-based and syntactic-based segmentation algorithms, and (b) we propose an efficient method for segmentation of color images based on spanning trees and both color and syntactic features of regions.

III. Segmentation Methods

We will compare four different segmentation techniques, the Mean Shift-Based segmentation algorithm [4], Efficient Graph-Based segmentation algorithm [3], Normalized Cuts segmentation algorithm [7] and our own region-based segmentation method. We have chosen Mean Shift-Based segmentation because it is generally effective and has become widely-used in the vision community. The Efficient Graph-Based segmentation algorithm was chosen as an interesting comparison to the Mean Shift. Its general approach is similar, however, it excludes the mean shift filtering step itself, thus partially addressing the question of whether the filtering step is useful. Due to its computational efficiency, Normalized Cuts represents a solid reference in our study. We use all these algorithms as terms of comparison for the evaluation we performed.

A. Graph-Based Salient Object Detection

We present an efficient segmentation method that uses color and some geometric features of an image to process it and create a reliable result [10]. The used color space is RGB because of the color consistency and its computational efficiency.

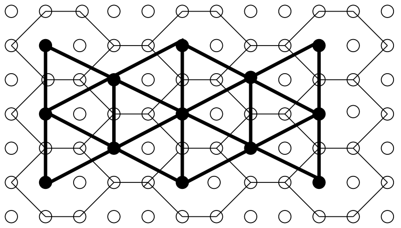


Fig. 1. The grid-graph constructed on the hexagonal structure of an image

What is particular at this approach is the basic usage of hexagonal structure instead of color pixels. In this way we can represent the structure 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 hexagon has six neighbors and each neighborhood connection is represented by an edge in the set E of the graph. For each hexagon on the structure two important attributes are associated: the dominant color and the coordinates of the gravity center. Basically, each hexagonal cell contains eight pixels: six from the frontier and two from the middle.

Image segmentation is realized in two distinct steps. The first step represents a pre-segmentation step when only color information is used to determine an initial segmentation. The second step represents a syntactic-based segmentation step when both color and geometric properties of regions are used.

The first step of the segmentation algorithm uses a color-based region model and will produce a forest of maximum spanning trees based on a modified form of the Kruskal's algorithm. In this case 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. The color-based segmentation algorithm builds a maximal spanning tree for each salient region of the input image.

The second step of the segmentation algorithm uses a new graph, which has a vertex for each connected component determined by the color-based segmentation algorithm. In this case the region model contains in addition some geometric properties of regions such as the area of the region and the region boundary. The final segmentation step produces a forest of minimum spanning trees based on a modified form of the Borůvka's algorithm. Each determined minimum spanning tree represents a final salient region determined by the segmentation algorithm.

B. Efficient Graph-Based Image Segmentation

Efficient Graph-Based image segmentation [3], is an efficient method of performing image segmentation. The basic principle is to directly process the data points of the image, using a variation of single linkage clustering without any additional filtering. A minimum spanning tree of the data points is used to perform traditional single linkage clustering from which any edges with length greater than a given threshold are removed [6].

Let $G = (V, E)$ be a fully connected graph, with m edges $\{e_i\}$ and n vertices. Each vertex is a pixel, x, represented in the feature space. The final segmentation will be $S = (C_1, ..., C_r)$, where C_i is a cluster of data points. The algorithm [3] can be shortly presented as follows:

1) Sort $E = (e_1, ..., e_m)$ such that $|e_t| \leq |e'_t| \forall t < t'$
2) Let S^0=$(\{x_1\}, ..., \{x_n\})$ in other words each initial cluster contains exactly one vertex.
3) For $t = 1, ..., m$
    a) Let x_i and x_j be the vertices connected by e_t.
    b) Let $C^{t-1}_{x_i}$ be the connected component containing point x_i on iteration $t-1$ and $l_i = max_{mst} C^{t-1}_{x_i}$ be the longest edge in the minimum spanning tree of $C^{t-1}_{x_i}$. Likewise for l_j.
    c) Merge $C^{t-1}_{x_i}$ and $C^{t-1}_{x_j}$ if:

$$|e_t| < min\left\{ l_i + \frac{k}{C^{t-1}_{x_i}}, l_j + \frac{k}{C^{t-1}_{x_j}} \right\} \quad (1)$$

4) $S = S^m$.

C. Normalized Cuts

Normalized Cuts method models an image using a graph $G = (V, E)$, where V is a set of vertices corresponding to image pixels and E is a set of edges connecting neighboring pixels. The edge weight $w(u, v)$ describes the affinity between two vertices u and v based on different metrics like proximity and intensity similarity. The algorithm segments an image into two segments that correspond to a graph cut (A, B), where A and B are the vertices in the two resulting subgraphs.

The segmentation cost is defined by:

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \qquad (2)$$

where $cut(A,B) = \sum_{u \in A, v \in B} w(u,v)$ is the cut cost of (A,B) and $assoc(A,V) = \sum_{u \in A, v \in V} w(u,v)$ is the association between A and V. The algorithm finds a graph cut (A,B) with a minimum cost in $Eq.(1)$. Since this is a NP-complete problem, a spectral graph algorithm was developed to find an approximate solution [7]. This algorithm can be recursively applied on the resulting subgraphs to get more segments. For this method, the most important parameter is the number of regions to be segmented. Normalized Cuts is an unbiased measure of dissociation between the subgraphs, and it has the property that minimizing normalized cuts leads directly to maximizing the normalized association relative to the total association within the sub-groups.

D. Mean Shift

The Mean Shift-Based segmentation technique [4] is one of many techniques dealing with "feature space analysis". Advantages of feature-space methods are the global representation of the original data and the excellent tolerance to noise [9]. The algorithm has two important steps: a mean shift filtering of the image data in feature space and a clustering process of the data points already filtered. During the filtering step, segments are processed using the kernel density estimation of the gradient. Details can be found in[4]. A uniform kernel for gradient estimation with radius vector $h = [h_s, h_s, h_r, h_r, h_r]$ is used. h_s is the radius of the spatial dimensions and h_r the radius of the color dimensions. Combining these two parameters, complex analysis can be performed while training on different subjects.

Mean shift filtering is only a preprocessing step. Another step is required in the segmentation process: clustering of the filtered data points $\{x'\}$. During filtering, each data point in the feature space is replaced by its corresponding mode. This suggests a single linkage clustering that converts the filtered points into a segmentation.

Another paper that describes the clustering is [5]. A region adjacency graph (RAG) is created to hierarchically cluster the modes. Also, edge information from an edge detector is combined with the color information to better guide the clustering. This is the method used in the available EDISON system, also described in [5]. The EDISON system is the implementation we use in our evaluation system.

IV. Region-based performance evaluation

We present comparative results of segmentation performance for our region based segmentation method and the three alternative segmentation methods mentioned above.

Our evaluation measure is mainly related to the consistency between segmentations. We use segmentation error measures that provide an objective analysis of the segmentation algorithms.

A potential problem for a measure of consistency between segmentations is that there is no unique human segmentation of an image, since each human perceives the scene differently. In this situation you could declare the segmentations inconsistent. However, if one segmentation is a refinement of the other, then the error should be small. Therefore, the measures are designed to be tolerant to refinement. Some other aspects to be taken into account are that error measure should not depend on the pixelation level and should be tolerant to noise along region boundaries.[12]

We used two metrics in order to provide an objective comparison between the four segmentation methods and the human segmentation. The two error measures are described below. We applied the measures to the Berkeley segmentation database and the segmentation results of the four algorithms.

In order to describe the segmentation errors, we considered two different segmentations S_1 and S_2 and calculated a value in the range $[0..1]$ where 0 represents no error. For a given p_i we considered segments S_1 and S_2 that contain the pixel. If one segment is a proper subset of the other, then the pixel lies in an area of refinement, and the local error should be zero. Otherwise, the two regions overlap in a inconsistent manner and we should calculate the corresponding error. We use \to denote the set difference and $|x|$ for cardinality of set x.If $R(S, p_i)$ is the set of pixels corresponding to the region in segmentation S that contains pixel p_i, the local refinement error is defined as:

$$E(S_1, S_2, p_i) = \frac{|R(S_1, p_i) n R(S_2, p_i)|}{|R(S_1, p_i)|} \qquad (3)$$

This local error measure is not symmetric. It encodes a measure of refinement in one direction only: $E(S_1, S_2, p_i)$ is zero precisely when S_1 is a refinement of S_2 at pixel p_i, but not vice versa. Considering this local refinement error in each direction at each pixel, there are two methods to combine the values into an error measure for the entire image. We apply two error measures as follows: $Global\ Consistency\ Error$ (GCE) that forces all local refinements to be in the same direction and $Local\ Consistency\ Error$ (LCE) that allows refinement in different directions in different parts of the image.[11] For a given n as the number of pixels we have:

$$GCE(S_1, S_2) = \frac{1}{n} min \left\{ \sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i) \right\} \qquad (4)$$

$$LCE(S_1, S_2) = \frac{1}{n} \sum_i min\{E(S_1, S_2, p_i), E(S_2, S_1, p_i)\} \qquad (5)$$

We will evaluate the performance of our algorithm on the Berkeley Segmentation Database (BSD) [8]. We will refer the characteristics of the error metrics previously defined by Martin et al. [2], explore potential problems with these metrics in order to evaluate the quality of each segmentation and to characterize its performance over a range of parameter values.

The current public version of the Berkeley Segmentation Database is composed of 300 color images. The images have a size of 481×321 pixels, and are divided into two sets, a training set containing 200 images that can be used to tune the parameters of a segmentation algorithm, and a testing set containing the remaining 100 images on which the final performance evaluations should be carried out.

We built a custom benchmark framework, that processes the Berkeley dataset, converts it to our proprietary format and preforms parallel analysis. Additionally, we adapted the other mentioned algorithms to the same evaluation format for unitary purposes.

The human segmented images provide the ground truth boundaries. Therefore, any boundary marked by a human subject is considered to be valid. Since there are multiple segmentations of each image by different subjects, it is the collection of these human-marked boundaries that constitutes the ground truth. Based on the output of the previously presented algorithms for a set of images, we will determine how well the ground truth boundaries are approximated.

In order to determine an algorithm's efficiency by comparing it to the ground truth boundaries, a threshold of the boundary map is needed.

We are providing an additional evaluation based on histogram representation of the error density characteristic for each algorithm.

V. Experimental Results

Our study of segmentation quality is based on experimental results and uses the Berkeley segmentation dataset provided at [8].

A. GCE and LCE Metrics

In order to proper evaluate the segmentation method we propose, we first need to better understand how the GCE and LCE error metrics work. Given two extreme cases:an under-segmented image, where every pixel has the same label (i.e. the segmentation contains only one region spanning the whole image), and a completely over-segmented image in which every pixel has a different label.

From the definitions of the GCE and LCE we can see that both measures evaluate to 0 on both of these extreme situations regardless of what segmentation they are being compared to. The reason for this can be found in the tolerance of these measures to refinement. Any segmentations is a refinement of the completely under-segmented image, while the completely over-segmented image is a refinement of any other segmentation.

In order to have a better analysis result and a more complete description for the errors we considered, we have performed 10 different tests for each subject per algorithm - Fig. 2.

More precisely, by varying several key parameters, we have obtained 10 distinct points that define the errors for each approach. For $Normalized\ Cuts$ [7] we have modified the $number\ of\ segments$ in the range of $\{5, 10, 12, 15, 20, 25, 30, 40, 50, 70\}$. The variable parameter for $Efficient\ Graph - Based\ Image\ Segmentation$ [3] was the scale of observation, k , in range {100, 200, 300, 400, 500, 600, 700, 800, 900, 1000}. For Mean-Shift [4] we have made 10 combinations from $Spatial\ Bandwidth$ {8, 16} and $Range\ Bandwidth$ {4, 7, 10, 13, 16}.

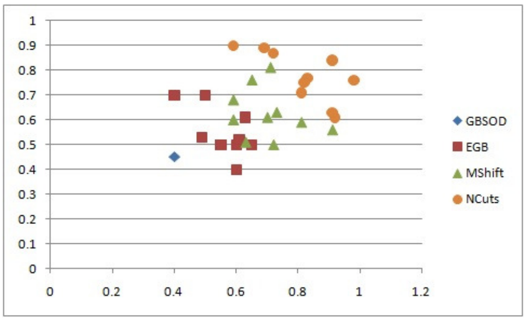


Fig. 2. Average GCE vs. LCE for Berkeley test images

We calculated the GCE and LCE average values for the 100 test images provided by Berkeley. Figure 2 illustrates the GCE vs. LCE graphic representation.

In the resulting diagram (Fig. 2) we can see that the GCE vs. LCE error metric for our proposed method, denoted $GBSOD - Graph\ Based\ Salient\ Object\ Detection$ is situated below the values for the other algorithms indicating a better performance result, a smaller average error and a balanced algorithm. Analyzing the set of results for each parameter per algorithm, it's easy to distinguish which algorithm is generating better results; the smaller the error it is, the better is the accuracy of the respective algorithm.

B. Histogram based evaluation

We elaborated a histogram-based evaluation mechanism aimed to compare the segmentation results for the studied algorithms via the errors metrics.

In order to achieve this, we considered the human segmentation as the ground-truth segmentation and compared each algorithm with it, measuring the error metrics GCE and LCE.

For each algorithm we analyzed the 100 test images from Berkley and calculated the corresponding GCE and LCE. The histograms presented below illustrate this approach (Fig.3 - Fig. 10).

For a better description of the histogram based analysis, in Fig.3 and Fig.4 we have depicted the distribution of the the values of GCE respectively LCE for the 100 images processed using $GBSOD - Graph\ Based\ Salient\ Object\ Detection$ algorithm. It is very important that these value are more concentrated on smaller error values, which gives us the confidence that the presented method has good results. In Fig.5 and Fig.6 it can be seen the same analysis for $Normalized\ Cuts$, in Fig.7 and Fig.8 for $Efficient\ Graph - Based\ Image\ Segmentation$ and in Fig.9 and Fig.10 for $Mean - Shift$ algorithm.

Figures Fig.11 and Fig.12 are illustrating a comparison between all the four presented algorithms; it gives a good

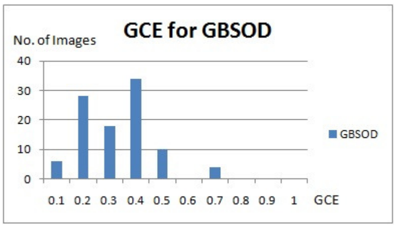


Fig. 3. GCE for Graph Based Salient Object Detection

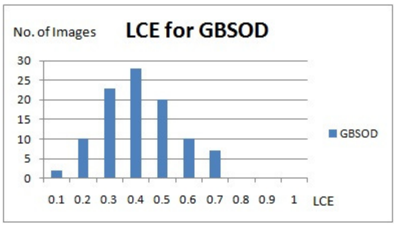


Fig. 4. LCE for Graph Based Salient Object Detection

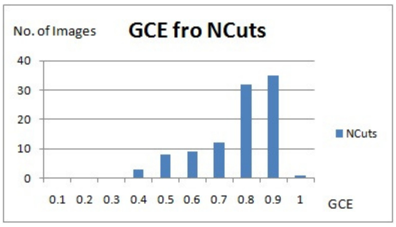


Fig. 5. GCE for Normalized Cuts

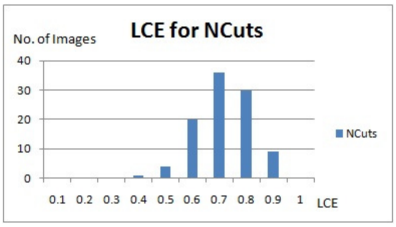


Fig. 6. LCE for Normalized Cuts

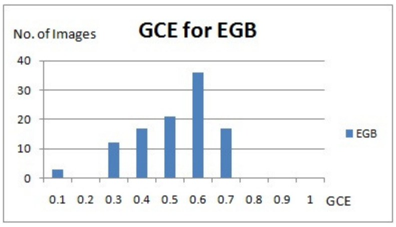


Fig. 7. GCE for Efficient Graph-Based Image Segmentation

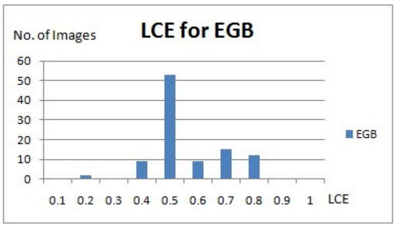


Fig. 8. LCE for Efficient Graph-Based Image Segmentation

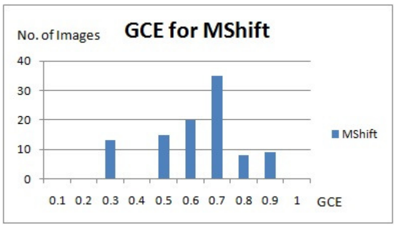


Fig. 9. GCE for Mean-Shift

perspective on the what error values generates each studied algorithm.

VI. Conclusion

In this paper we presented a new graph-method for image segmentation and extraction of visual objects. Starting from a survey of several segmentation strategies, we aimed at performing an image segmentation evaluation experiment.

Our segmentation method and other three segmentation methodologies were chosen for the experiment, and the complementary nature of the methods was demonstrated in the results. The study results offer a clear view of the effectiveness of each segmentation algorithm, trying in this way to offer a solid reference for future studies.

Future work will be carried out in the direction of integrating syntactic visual information into a semantic level of a semantic image processing and indexing system.

Acknowledgment

This work was supported by $CNCSIS - UEFISCSU$, project number $PNII - IDEI$ code/2008.

References

[1] K. Fu and J. Mui, “A survey on image segmentation. Pattern Recognition”, 1981.

[2] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics”, Proc. Int. Conf. Comp. Vis., vol. 2, pp. 416-425, 2001.

[3] P. Felzenszwalb and D. Huttenlocher, “Efficient Graph-Based Image Segmentation”, Intl J. Computer Vision, vol. 59, no. 2, 2004.

[4] 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.

[5] C. Christoudias, B. Georgescu, and P. Meer, “Synergism in Low Level Vision”, Proc. Intl Conf. Pattern Recognition, vol. 4, pp. 150-156, 2002.

[6] R. Unnikrishnan, C. Pantofaru, and M. Hebert, “Toward Objective Evaluation of Image Segmentation Algorithms”, IEEE Transactions on pattern analysis and machine inteligence, Vol. 29, No. 6, 2007.

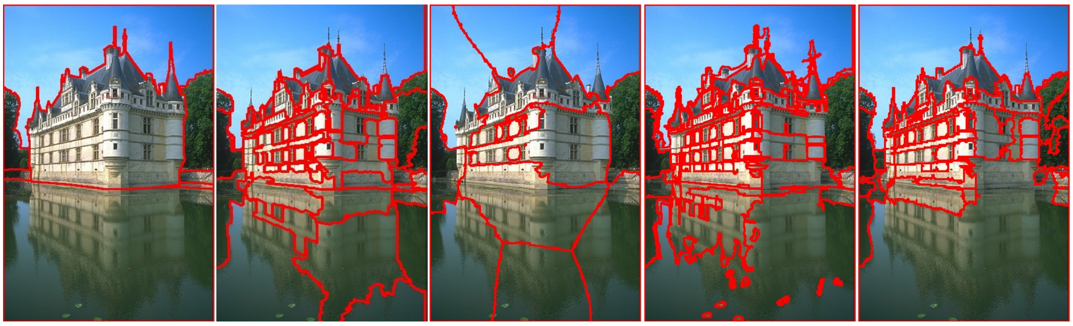


Fig. 13. Comparative segmentation results: Human Segmentation, Graph-based Salient Object Detection, Normalized Cuts, Efficient Graph-based, Mean-Shift

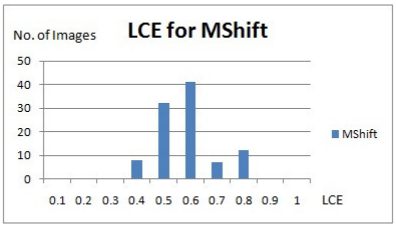


Fig. 10. LCE for Mean-Shift

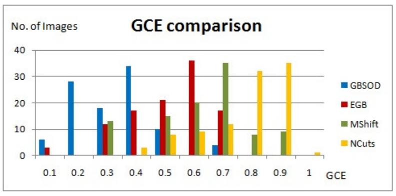


Fig. 11. GCE overall comparison

[7] J. Shi and J. Malik, “Normalized Cuts and Image Segmentation”, IEEE Transactions on pattern analysis and machine intelligence, Vol. 22, No. 8, 2000.

[8] “Berkeley Segmentation and Boundary Detection Benchmark and Dataset”, 2003, http://www.cs.berkeley.edu/projects/vision/grouping/segbench.

[9] R.O. Duda, P.E. Hart, and D.G. Stork, “Pattern Classification”, John Wiley & Sons, New York, 2000.

[10] 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.

[11] D. Martin, C. Fowlkes, D. Tall, and J. Malik, “A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics”, ICCV Vancouver, 2001.

[12] R. Unnikrishnan, C. Pantofaru, and M. Hebert, “A Measure for Objective Evaluation of Image Segmentation Algorithms”,

[13] C. Carson, S. Belongie, H. Greenspan, and J. Malik, “Blobworld: Image segmentation using expectation-maximization and its application to image querying and classification”, IEEE Trans. on Pattern Analysis and Machine Intelligence, 24(8),1026–1037, 2002.

[14] J. Fauqueur and N. Boujemaa, “Region-based image retrieval: Fast coarse segmentation and fine color description”, Journal of Visual Languages and Computing, 15(1), 69–95, 2004.

[15] J. Shi and J. Malik, “Normalized cuts and image segmentation”, Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 731–737, 1997.

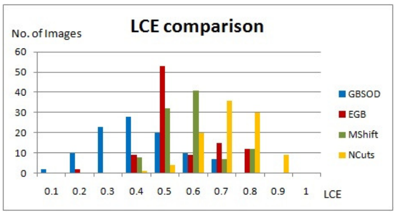


Fig. 12. LCE overall comparison

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 723–728

ISBN 978-83-60810-27-9
ISSN 1896-7094

Undetectable Spread-time Stegosystem Based on Noisy Channels

Valery Korzhik (Member, IEEE)
State University
of Telecommunications
St. Petersburg, Russia
Email: korzhik@spb.lanck.net

Guillermo Morales-Luna
Computer Science
CINVESTAV-IPN
Mexico City, Mexico
gmorales@cs.cinvestav.mx

Ksenia Loban
State University
of Telecommunications
St. Petersburg, Russia

Irina Marakova-Begoc
Bretagne Telecom, France
marakova.irina@gmail.com

***Abstract*—We consider a scenario where an attacker is able to receive a stegosignal only over a Gaussian channel. But in order to provide security of this channel noise–based stegosystem under the very strong condition that an attacker may know even the cover message, it is necessary to establish a very low signal-to-noise ratio in the channel. The last requirement is very hard to be implemented in practice. Therefore we propose to use spread-time stegosystem (STS). We show that both security and reliability of such STS can be guaranteed and their parameters can be optimized with the use of error correcting codes. We show some simulation results with an own STS implementation for digital audio cover messages presented in WAV format.**

***Index Terms*—Digital audio signal, error correcting codes, noisy Gaussian channel, relative entropy, stegosystems.**

I. Introduction

Steganography (SG) is the information hiding technique that embeds the hidden information into an innocent *cover message* (CM) under the conditions that the CM is not corrupted significantly and that the presence of the additional information into the CM may not be detected.

In order to prevent statistical detecting attacks on SG systems, it should be guaranteed the following principle: the statistics of the CM and the SG signal have to be indistinguishable for the time limited analysis.

But in order to implement this principle, the designer of the SG system should know at least the statistics of the CM. At the same time, it is a rather hard problem to study completely the CM distribution. In order to be successful within this risky situation (which is, indeed, a bottleneck of any SG system), it has been proposed in [1] to move into another concept of SG system setting, namely to SG system *based on noisy channels*.

This setting can be justified only if there exists in a natural manner a noisy channel and the attacker is able to receive the stegosignal just over this channel, and nothing else. Then the attacker's problem consists in statistically distinguishing the CM after its passing over the noisy channel and the SG signal passing over the same noisy channel. It should be emphasized that such model is even stronger than conventional SG systems since CM can be publicized. Thus the steganalysis problem reduces to channel noise recognition within the sum of the channel noise and the embedded signal. Since the channel noise distribution is, as a rule, known much better than the CM distribution, the problem to design SG systems which are resistant to their detection is simplified.

In the current paper we adopt only a Gaussian channel from the two models given in [1]. The embedding of an information bit b can be provided as

$$\forall n=1,\ldots,N: \quad C_W(n)=C(n)+(-1)^b\sigma_W\pi(n) \tag{1}$$

where $C=(C(n))_{n=1}^N$ is the CM, $\pi=(\pi(n))_{n=1}^N$ is a zero-mean Gaussian pseudorandom i.i.d. reference sequence with variance 1, N is the length of both sequences and σ_W is the depth of embedding. After a passing of the watermarked signal through the Gaussian channel we get

$$\forall n=1,\ldots,N: \quad C'_W(n)=C_W(n)+\varepsilon(n)$$

where $\varepsilon=(\varepsilon(n))_{n=1}^N$ is a zero-mean Gaussian i.i.d. noise sequence with variance σ_ε^2. It has been proved in [1] that, under the condition that an attacker knows even the CM, the *relative entropy* D (introduced in [2]) can be expressed, for the current SG-system model, as

$$D=0.72\ N\left[\ln\left(1+\frac{1}{\eta_W}\right)-\frac{1}{1+\eta_W}\right] \tag{2}$$

with $\eta_W=\frac{\sigma_\varepsilon^2}{\sigma_W^2}$. In order to provide a good hiding of secret information into the channel noise, η_W should be taken large. Hence, the relative entropy given by (2) is approximated as

$$D=0.36\ \frac{N}{\eta_W^2}. \tag{3}$$

We recall that in line with Information Theory [2], for any hypothesis testing rule, the following inequality should hold:

$$P_{fa}\ \ln\frac{P_{fa}}{1-P_m}+(1-P_{fa})\ \ln\frac{1-P_{fa}}{P_m}\le D, \tag{4}$$

where P_{fa} is the probability of *SG signal false alarm* and P_m is the probability of *SG signal missing*. Let us assume, for simplicity, $P_{fa}=P_m=P$. Then, by (4), we get $(2P-1)\ \ln\frac{P}{1-P}\le D$. From eq. (3) it follows that

$$\eta_W=0.6\ \sqrt{\frac{N}{D}}. \tag{5}$$

The optimal decision rule for the embedded bit b, in the case of a Gaussian channel and decoder's CM knowledge (*informed decoder*), is

$$\Lambda = \sum_{n=1}^{N} (C'(n) - C(n))\, \pi(n) \;\Rightarrow\; \tilde{b} = \begin{cases} 0 & \text{if } \Lambda \geq 0 \\ 1 & \text{otherwise} \end{cases} \qquad (6)$$

It is easy to show that for a Gaussian reference sequence $\pi = (\pi(n))_{n=1}^{N}$ the error probability of the decision rule (6) is

$$P_e = Q\left(\sqrt{\frac{N}{\eta_W + 2}}\right) \leq \exp\left(-\frac{N}{2(\eta_W + 2)}\right) \qquad (7)$$

where $Q : x \mapsto Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{+\infty} e^{-\frac{u^2}{2}}\, du$. If $\eta_W >> 1$ (which is a rather common situation in SG systems) both security and reliability can be pooled together in one expression:

$$P_e = Q\left(1.29\,(ND)^{\frac{1}{4}}\right) \leq \exp\left(-0.83\,(ND)^{\frac{1}{2}}\right). \qquad (8)$$

From (8) there follows that for any security level D there can be chosen an appropriate N such that the SG system provides any given reliability P_e.

But this apparent good design of the SG system has one defect. Namely, if one wants to embed many secret bits into the CM in such a way that they can be reliably decoded by a legal user, it is necessary to increase the parameter η_W and this may not be possible in practical implementation. In order to see sharply this negative property, let us consider the following:

Example 1: Let $D = 0.1$ (that provides an acceptable level of security) and let $m = 10$ be the number of secure embedded bits. For multiple bit embedding, (8) should be posed as

$$P_e \leq \exp\left(-0.83\,\frac{(ND)^{\frac{1}{2}}}{m}\right). \qquad (9)$$

Let us choose then $N = 10^5$. From (9), the probability of error is bounded as $P_e \leq 2.5 \times 10^{-4}$ which is acceptable. But $\eta_W = 600$, by (5). Now, if the CM signal-to-noise ratio has been taken also within an acceptable, say $\frac{\sigma_C^2}{\sigma_\varepsilon^2} = 10^2$, where $\sigma_C^2 = \text{Var}\left((C(n))_{n=1}^{N}\right)$, then $\frac{\sigma_\varepsilon^2}{\sigma_W^2} = 6 \times 10^4$ which is indeed unacceptable for the most practical digital applications.

In order to overskip this unfortunate situation, we propose in section 2 the so called *spread-time stegosystem* (STS). The security and reliability of STS are proved in that section jointly with an optimization of parameters. An improvement of the cost of using error correcting codes is also given there. Section 3 presents the results of STS simulation for digital audio signal with a CM in the WAV format. Section 4 consists of some conclusions and open problems in this direction.

II. Description of STS and its performance evaluation

Let us consider initially an uncoded stegosystem. Let us embed secret bits b as a random modification of the embedding rule (1), namely, $\forall n = 1, \ldots, N$:

$$\begin{array}{rcl} \Pr\left[C_W(n) = C(n) + (-1)^b \sigma_W \pi(n)\right] & = & P_0 \\ \Pr\left[C_W(n) = C(n)\right] & = & 1 - P_0 \end{array} \qquad (10)$$

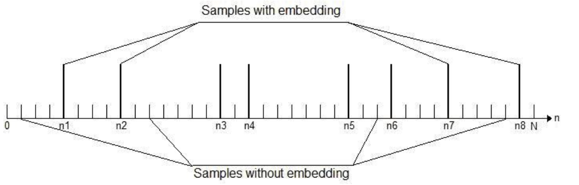


Fig. 1. Pseudorandom samples for STS system embedding, with $N_s = 8$, $N = 38$, $P_0 = 8/38 = 4/19$.

In practical implementation of the modified embedding rule (10), we can simply use a pseudorandom subsequence of samples. Let $(n_m)_{m=1}^{N_s}$ be an increasing sequence of indexes, $N_s \leq N$, generated also as a secret stegokey K, determining the samples in which the WM's are to be embedded (see Fig. 1). Then for a large value of N we may assume that $P_0 = N_s/N$. For uncoded SG, the same secret bit b is used at N_0 consecutive chosen samples for embedding. Hence the total number of secret bits embedded into N_0 samples for STS is $N_t = N_s/N_0$. Any legal user should know the stegokey K, hence he knows exactly the samples with embedding, and is able to extract one-by-one all the N_t secret bits using the decision rule (6). The error probability can be found by (7) (by considering the N appearing there as the current N_0).

An attacker A ignores the stegokey and hence the samples with the WM embedding. In order to take a decision about presence or absence of the SG system under the condition of a known $C = (C(n))_{n=1}^{N}$, the attacker A has to perform a testing of two hypothesis. Let

$$\delta = \left(\delta(n) = C'_W(n) - C(n)\right)_{n=1}^{N} \qquad (11)$$

and $\sigma_s^2 = \sigma_\varepsilon^2 + \sigma_W^2$. Then the two hypothesis to be tested are:

$$H_0 : \left[\delta \in \text{N}(0, \sigma_\varepsilon^2) \text{ and is an i.i.d}\right] \qquad (12)$$

$$H_1 : \begin{cases} \Pr\left(\delta \in \text{N}(0, \sigma_s^2) \text{ and is an i.i.d}\right) = P_0 \\ \Pr\left(\delta \in \text{N}(0, \sigma_\varepsilon^2) \text{ and is an i.i.d}\right) = 1 - P_0 \end{cases} \qquad (13)$$

The hypothesis testing can be done using the *maximum likelihood ratio*

$$\Lambda\left(\Lambda_1 | \Lambda_0\right) = \frac{P\left(\delta | H_1\right)}{P\left(\delta | H_0\right)}$$

where $P\left(\delta | H_j\right)$ is the probability distribution of the random variables $\left(\delta(n)\right)_{n=1}^{N}$ under the condition that hypothesis H_j is valid, $j = 0, 1$. Namely, the *optimal hypothesis testing* based on maximum likelihood ratio [3] is

$$\begin{array}{lcl} \Lambda\left(\Lambda_1 | \Lambda_0\right) \geq \lambda & \Longrightarrow & H_1 \\ \Lambda\left(\Lambda_1 | \Lambda_0\right) < \lambda & \Longrightarrow & H_0 \end{array} \qquad (14)$$

where λ is some fixed threshold. By substituting into (14) the probability distributions (12)-(13) we get after simple transforms

$$\Lambda\left(\Lambda_1 | \Lambda_0\right) = \prod_{n=1}^{N} \left[P_0 \sqrt{\frac{\sigma_\varepsilon^2}{\sigma_s^2}} \exp\left(\frac{\sigma_W^2}{2\sigma_s^2 \sigma_\varepsilon^2} \delta(n)^2\right) + (1 - P_0)\right]$$

By changing λ in (14), it is possible to pass to the logarithmic likelihood ratio, $\Lambda_L\left(\Lambda_1|\Lambda_0\right) = \log\Lambda\left(\Lambda_1|\Lambda_0\right)$, and it equals

$$\sum_{n=1}^{N}\log\left[P_0\sqrt{\frac{\sigma_\varepsilon^2}{\sigma_s^2}}\exp\left(\frac{\sigma_W^2}{2\sigma_s^2\sigma_\varepsilon^2}\delta(n)^2\right)+(1-P_0)\right] \qquad (15)$$

The application of the transformed decision rule based on $\Lambda_L\left(\Lambda_1|\Lambda_0\right)$ using (15) is rather hard. We will consider it again something later.

But so far let us consider a suboptimal decision rule based on some further reasonable conditions. First of all let us assume, in line with a good security guarantee, $\sigma_W^2 << \sigma_\varepsilon^2$. Then, a normalization of (15) expresses $\Lambda_L\left(\Lambda_1|\Lambda_0\right)$ as

$$\frac{1}{N}\sum_{n=1}^{N}\log\left[P_0\exp\left(\frac{1}{2\eta_W^2\sigma_\varepsilon^2}\delta(n)^2\right)+(1-P_0)\right] \qquad (16)$$

The series expansion of $x\mapsto\log(1+x)$ up to its linear term produces

$$\Lambda_L\left(\Lambda_1|\Lambda_0\right) = P_0\left[\sum_{n=1}^{N}\exp\left(\frac{1}{2\eta_W^2\sigma_\varepsilon^2}\delta(n)^2\right)-N\right].$$

The series expansion of $x\mapsto\exp(x)$ up to its linear term renders the following decision rule:

$$\left[\tilde{\Lambda}\geq\tilde{\lambda}\implies H_1\right] \quad ; \quad \left[\tilde{\Lambda}<\tilde{\lambda}\implies H_0\right] \qquad (17)$$

where $\tilde{\Lambda} = \frac{1}{N}\sum_{n=1}^{N}\delta(n)^2$ and $\tilde{\lambda}$ is some new threshold. The decision rule (17) is sufficiently reasonable because $E[\delta^2|H_1] > E[\delta^2|H_0]$ as we will show later.

Let us estimate the missing and false alarm probabilities, P_m and P_{fa} respectively, for the hypothesis H_1 (presence of the SG system) against hypothesis H_0 (absence of the SG system) under the decision rule given by (17).

For enough large N, by the Central Limit Theorem [4], $\tilde{\Lambda}\in\mathrm{N}(\mu_j,\sigma_j^2)$ for H_j, where $\mu_j = E[\tilde{\Lambda}|H_j]$ and $\sigma_j^2 = \mathrm{Var}\left(\tilde{\Lambda}|H_j\right)$, for $j=0,1$. Since $\sigma_1^2 > \sigma_0^2$ we get:

$$P_m \geq \frac{1}{\sqrt{2\pi\sigma_0^2}}\int_{-\infty}^{\tilde{\lambda}}\exp\left(-\frac{(x-\mu_1)^2}{2\sigma_0^2}\right)dx \qquad (18)$$

$$P_{fa} \geq \frac{1}{\sqrt{2\pi\sigma_0^2}}\int_{\tilde{\lambda}}^{+\infty}\exp\left(-\frac{(x-\mu_0)^2}{2\sigma_0^2}\right)dx \qquad (19)$$

Let us select the threshold $\tilde{\lambda}$ in such a way that the condition $P_m = P_{fa} = P$ is fulfilled. After simple transforms of eq's (18)-(19) it is obtained

$$P\geq Q\left(\frac{\mu_1-\mu_0}{2\sigma_0}\right). \qquad (20)$$

Necessarily the following identities should hold:

$$\mu_0=\sigma_\varepsilon^2\ ;\ \mu_1=\sigma_\varepsilon^2+P_0\sigma_W^2\ ;\ \sigma_0^2=\frac{2}{N}\sigma_\varepsilon^4. \qquad (21)$$

By substituting (21) into (20), we get

$$P\geq Q\left(\sqrt{\frac{N}{2}}\frac{P_0}{2\eta_W}\right),$$

or equivalently,

$$P\geq Q\left(\frac{N_s}{2\sqrt{2N}\eta_W}\right),$$

where, as introduced at the beginning of the current section, N_s is the number of samples with embedding. Consequently if asymptotically $N_s\sim\sqrt{N}$, then $P\sim\frac{1}{2}$ and an undetectable stegosystem results.

N	η_W	N_0	N_s	m	$P_0=\frac{N_s}{N}$
10^4	20	210	1431	6	0.1431
	50	496	3578	7	0.3578
	100	973	7156	7	0.7156
10^5	20	210	4526	21	0.04526
	50	496	11310	22	0.1131
	100	973	22630	23	0.2263
10^6	20	210	14310	68	0.01431
	50	496	35780	72	0.03578
	100	973	71560	73	0.07156
10^7	20	210	45260	215	0.004526
	50	496	113100	228	0.01131
	100	973	226300	232	0.02263

TABLE I
SETS OF PARAMETERS FOR STS PROVIDING $P_0\geq 0.4$ AND $P_e\leq 10^{-3}$ GIVEN DIFFERENT VALUES OF N AND η_W.

In order to embed m secret bits into N_s samples, $N_0=\frac{N_s}{m}$ samples should be selected for embedding each bit. Then the error probability P_e after extraction of one bit by a legal informed decoder is expressed by (7) (with N_0 playing the role of N). It is necessary to note that in order to extract the secret bits, the legal decoder has to be synchronized with both the reference sequence π, appearing in relation (1), and the pseudorandom sequence determining the samples with embedding.

In Table I we show the calculation results for some values of parameters N_s, N_0, m, P_0 providing $P_e\leq 10^{-3}$ and $P\geq 0.4$, given some values of N and η_W. For enough large N, it is possible to provide a good undetectability ($P_0\geq 0.4$) and reliability ($P_e\leq 10^{-3}$) of the STS and embed up to 232 secure bits.

In order to improve the STS efficiency it is possible to use *coded STS*. Then an embedding procedure such as (10) has to be replaced as follows: Given a CM $C=(C(n))_{n=1}^N$, let $C_W=(C_W(n))_{n=1}^N$ be such that for each sample index n_j with embedding

$$\begin{aligned}\Pr\left[C_W(n_j)=C(n_j)+(-1)^{b_{ij}}\sigma_W\pi(n_j)\right] &= P_0\\ \Pr\left[C_W(n_j)=C(n_j)\right] &= 1-P_0\end{aligned}$$

where b_{ij} is the j-th bit in the i-th codeword of length $N_0 = N_s/\ell$, with ℓ a positive integer value.

We will restrict our attention to binary linear systematic (N_0,k,d)-codes, varying i in the interval $\{1,2,\cdots,2^k-1,2^k\}$, with d the minimal code distance. In this setting the informed decoder takes a decision about the embedding of the i-th codeword by making

$$i = \arg \max_{1 \le i' \le 2^k} \sum_{j=1}^{N_0} \left(C'_W(n_j) - C(n_j) \right) (-1)^{b_{i'j}} \pi(n)$$

The total number of secure embedded bits is $m = k\ell$ and the block-error probability P_{be}, based on well known union bound [5], can be expressed as

$$\begin{aligned} P_{be} &\le (2^k - 1)\, Q \left(\sqrt{\frac{d}{2 + \eta_W}} \right) \\ &\le \exp \left(- \frac{d}{2(2 + \eta_W)} + R\, N_0 \ln 2 \right) \end{aligned}$$

Since signal-to-noise ratio η_W^{-1} is typically small, we will restrict our consideration only to two classes of linear error correcting codes: the simplex codes (SC) and the Reed-Muller codes (RMC) [5]. For the first class the main parameters are $N_0 = 2^\nu - 1$, $k = \nu$, $d = 2^{\nu-1}$, $R = \frac{\nu}{N_0}$, where ν is some integer; whereas for the second class: $N_0 = 2^\nu$, $k = \sum_{i=1}^{r} \binom{\nu}{i}$, $d = 2^{\nu - r}$, where $\nu \ge 3$ and r is an integer, the so called *order of the* RMC.

Now we can fix the total number of samples N, the security level P, the block-error probability P_{be}, the parameter η_W and then to optimize the code parameters N_0, ν and r in order to provide the maximum possible number m of secure and reliable embedded bits.

Example 2: Let us take $N = 10^7$, $P \ge 0.4$, $P_{be} \le 10^{-3}$, $\eta_W = 20$. Then we get for the class of SC the optimal parameters $\nu = 10$, $k = 10$, the total number of secret bits $m = k\frac{N_s}{N_0} = 442$. If we require more reliable extraction then we get, for the class of RM codes, the optimal parameters $\nu = 14$, $r = 2$, $k = 105$ and for the same restrictions $P \ge 0.4$, $\eta_W = 20$, the total number m of embedded secret bits is about 290 with $P_{be} \le 10^{-9}$. So, we can conclude that the use of error correcting codes results in either an increment in the number of secure embedded bits or in an improvement of reliability.

Let us find out whether the use of the optimal decision rule (16) can provide an appreciable improvement of STS detecting in comparison with the suboptimal decision rule (17).

Since N is sufficiently large, we can apply the Central Limit Theorem to the sum in (16). Then similar to the proof of (20) we get for such a choice of the threshold λ, which provides $P_m = P_{fa} = P$ the following upper bound

$$P \ge Q \left(\frac{\tilde{\mu}_1 - \tilde{\mu}_0}{2\tilde{\sigma}_0} \right) \tag{22}$$

where, for $j = 0, 1$,

$$\tilde{\mu}_j = E \left[\left(\log \left(P_0 \exp \left(\frac{\delta(n)^2}{2\eta_W \sigma_\varepsilon^2} \right) + (1 - P_0) \right) \right)_{n=1}^{N} \middle| H_j \right]$$

and

$$\begin{aligned} \tilde{\sigma}_0 &= \frac{1}{N} \mathrm{Var} \left((s(n))_{n=1}^{N} \middle| H_j \right) \\ &= \frac{1}{N} \left(E \left[\left((s(n))^2 \right)_{n=1}^{N} \middle| H_j \right] - \tilde{\mu}_0^2 \right) \end{aligned}$$

where

$$s(n) = \log \left(P_0 \exp \left(\frac{\delta(n)^2}{2\eta_W \sigma_\varepsilon^2} \right) + (1 - P_0) \right)$$

and the random values $\delta(n)$ have the probability distributions given by (11). Since it is very hard to find analytically the values $\tilde{\mu}_0$, $\tilde{\mu}_1$ and $\tilde{\sigma}_0$, we will estimate them just by the simulation of the above described procedure.

In Table II there are presented the simulation results for $\tilde{\mu}_0$, $\tilde{\mu}_1$ and $\tilde{\sigma}_0$ and the calculation of P by (22) for typical values of σ_ε^2, η_W and P_0. It can be seen that the use of the optimal decision rule does not break undetectability of STS, hence it can be declared as a secure SG system indeed.

η_W	N_0	P_e	$\tilde{P}_e$
20	210	$5.0 \cdot 10^{-4}$	0.001
50	496	$6.0 \cdot 10^{-4}$	0.001
100	973	$5.5 \cdot 10^{-4}$	0.001

TABLE III
THE RESULTS OF CALCULATIONS FOR THE ERROR PROBABILITY P_e OBTAINED AFTER DECODING BY RULE (6) AND THE THEORETICAL ERROR PROBABILITY $\tilde{P}_e$ CALCULATED BY EQ. (7), WITH $N = N_0$ FOR DIFFERENT PARAMETERS η_W AND N_0.

III. SIMULATION OF STS FOR AUDIO COVER MESSAGES

We use an audio music file in format WAV where the sample frequency is 44.1 kHz with duration about 29 sec. The CM signal-to-noise ratio η_c has been taken as 10 dB, whereas watermark-to-noise ratio (WNR) η_W^{-1} was 20 dB. The embedding rule was taken as (10), where $P_0 = 0.1$. In Fig. 2 the wave forms of the original audio signal, audio signal after passing over a noisy channel and after secret message embedding are presented at the same time interval. One can see that the noise corrupts slightly the audio signal and this fact can also be appreciated by human ear, whilst, at the same time, the embedding procedure is not observable.

Moreover, in Fig. 3 the waveforms of channel noise are shown, as well as this noise after embedding with straining in time confirming this fact. (Of course we do not claim that the impossibility to detect the SG system either by ear or by eye, is enough to prove its security by the best statistical methods. We have proved indeed this fact in the previous section).

In Table III we present the results of simulation for the error probability P_e versus the block length N_0 and η_W. The error probability $\tilde{P}_e$ calculated by eq. (7) is also presented in this Table. There, we can see that the reliability of STS obtained by simulation is even better than the theoretical estimated bound.

IV. CONCLUSIONS

In the current paper we proposed some modification of the stegosystem based on noisy channel called spread-time stegosystem (STS). The goal of the STS is to provide such a WNR able to be implemented in practice, especially with digital cover messages. We prove that both STS security

N	σ_ε^2	η_W	$P_0 = \frac{N_s}{N}$	$\tilde{\mu}_0$	$\tilde{\mu}_1$	$\tilde{\sigma}_0$	$P = Q\left(\frac{\tilde{\mu}_1 - \tilde{\mu}_0}{2\tilde{\sigma}_0}\right)$
10^4	1	20	0.1431	0.00161414	0.00162667	0.00240398	0.401753
		50	0.3578	0.00157674	0.00158801	0.00229017	0.401759
		100	0.7156	0.00156462	0.00157585	0.00225445	0.401737
	5	20	0.1431	0.00161414	0.00162590	0.00240398	0.401754
		50	0.3578	0.00157675	0.00158821	0.00229017	0.401745
		100	0.7156	0.00156462	0.00157583	0.00225445	0.401726
10^5	1	20	0.04526	0.000512618	0.000513737	0.000767001	0.401741
		50	0.1131	0.000500332	0.000501449	0.000729712	0.401809
		100	0.2263	0.000496672	0.000497830	0.000718483	0.401737
	5	20	0.04526	0.000512618	0.000513854	0.000767001	0.401772
		50	0.1131	0.000499062	0.000500277	0.000727769	0.401806
		100	0.2263	0.000496672	0.000497795	0.000718483	0.401745
10^6	1	20	0.01431	0.000162288	0.000162393	0.000243341	0.401835
		50	0.03578	0.000158479	0.000158585	0.000231440	0.401862
		100	0.07156	0.000157686	0.000157808	0.000228397	0.401548
	5	20	0.01431	0.000162288	0.000162435	0.000243187	0.401752
		50	0.03578	0.000158479	0.000158598	0.00023144	0.401711
		100	0.07156	0.000157686	0.000157797	0.000228397	0.401461
10^7	1	20	0.004526	$5.13502 \cdot 10^{-5}$	$5.13615 \cdot 10^{-5}$	$7.69844 \cdot 10^{-5}$	0.401964
		50	0.01131	$5.01145 \cdot 10^{-5}$	$5.01246 \cdot 10^{-5}$	$7.32173 \cdot 10^{-5}$	0.401900
		100	0.02263	$4.97464 \cdot 10^{-5}$	$4.97587 \cdot 10^{-5}$	$7.20836 \cdot 10^{-5}$	0.401777
	5	20	0.004526	$5.13502 \cdot 10^{-5}$	$5.13626 \cdot 10^{-5}$	$7.69844 \cdot 10^{-5}$	0.401812
		50	0.01131	$5.01145 \cdot 10^{-5}$	$5.01245 \cdot 10^{-5}$	$7.32173 \cdot 10^{-5}$	0.401969
		100	0.02263	$4.97464 \cdot 10^{-5}$	$4.97569 \cdot 10^{-5}$	$7.20836 \cdot 10^{-5}$	0.401686

TABLE II
RESULTS OF SIMULATIONS FOR VALUES OF $\tilde{\mu}_0$, $\tilde{\mu}_1$ AND $\tilde{\sigma}_0$ VERSUS TYPICAL VALUES OF σ_ε^2, η_W AND P_0.

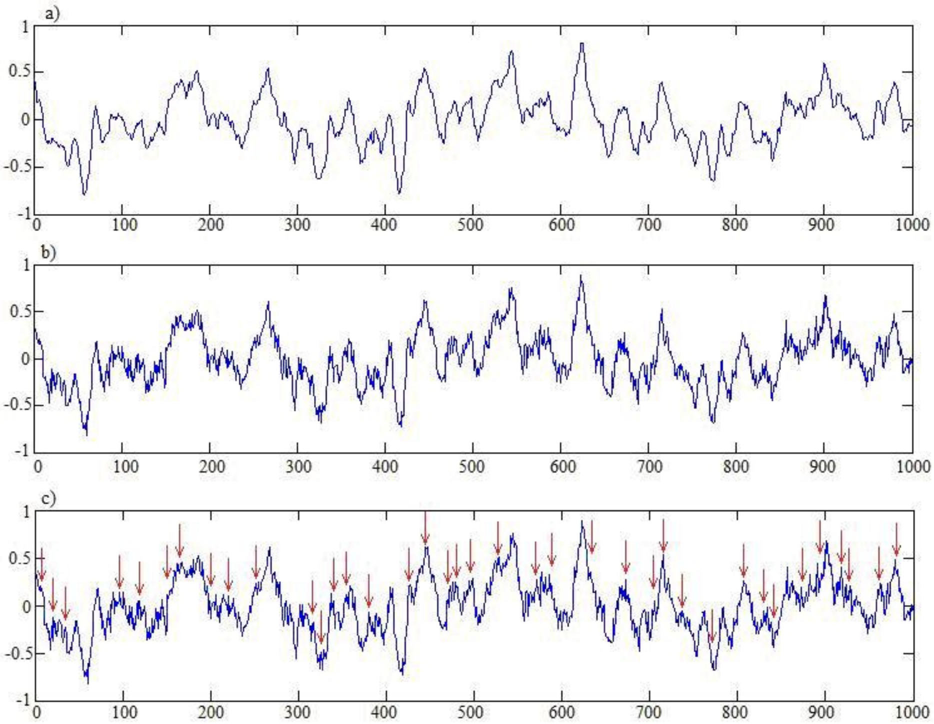


Fig. 2. (a) The waveforms of audio signal, (b) audio signal after its passing over noisy channel with CM signal-to-noise ratio $\eta_c = 10$dB, and (c) after embedding by STS algorithm with $WNR = 20$dB. The arrows show the samples with embedding.

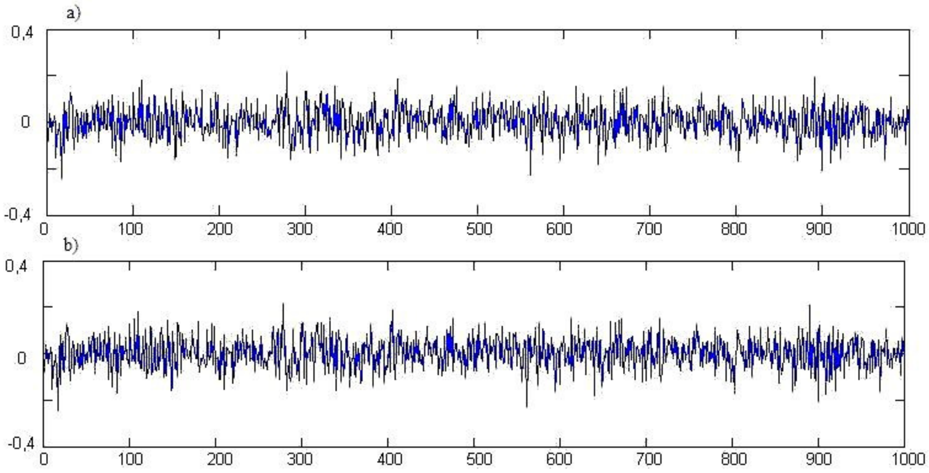


Fig. 3. (a) The waveform of channel noise and (b) the same channel noise after embedding according to rule (10).

and reliability of secure bit extraction can be provided by an appropriate selection of the system parameters. The main defect of the proposed stegosystem is its low embedding rate which entails longer times for the embedding of a limited number of secret bits. The used error correcting codes improve this situation but only slightly. However this is a generic property "sacrificed on the altar of undetectability" and an attacker's knowledge of the CM.

We show that the suboptimal SG system detection (see eq. (17) is practically as much efficient as the optimal (based on the maximum likelihood ratio). Simulation of the STS with audio CM shows that its detection by ear and eye is impossible, whereas the embedded bits can be extracted reliably.

The first of open problems which we are going to consider in the near future is to specify security of STS for digital CM and after saving the stegosignal in digital formats. The second problem considers an extraction of secret bits by a blind decoder (in particular using the improved spread spectrum modulation [6]) while keeping a good undetectability of STS.

Acknowledgment

Dr. Morales-Luna acknowledges the support of Mexican Conacyt.

References

[1] V. Korjik, M. H. Lee, and G. Morales-Luna, "Stegosystems based on noisy channels," in *Proc. IX Spanish Meeting on Cryptology and Information Security*. Univ. Aut. Barcelona, 2006, pp. 379–387.

[2] C. Cachin, "An information-theoretic model for steganography," in *International Workshop on Information Hiding 1998*. Springer LNCS, 1998, pp. 306–318.

[3] B. van der Waerden, *Mathematische Statistik*. Springer, 1957.

[4] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. Mc-Graw Hill, 1984.

[5] F. J. Macwilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, ser. North-Holland Mathematical Library. North Holland, January 1983. [Online]. Available: http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0444851933

[6] H. S. Malvar and D. Florêncio, "Improved spread spectrum: A new modulation technique for robust watermarking," *IEEE Trans. on Signal Processing*, vol. 51, no. 4, pp. 898–905, 2001.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Building Personalized Interfaces by Data Mining Integration

Mihăescu Marian Cristian
University of Craiova
Software Engineering Department,
Bvd. Decebal, Nr. 107, 200440,
Craiova, Dolj, Romania
Email: mihaescu@software.ucv.ro

***Abstract*—Building personalized high quality multimedia interfaces represents a great challenge. This paper presents a custom procedure of buiding personalized interfaces within e-Learning environments. The procedure has an interdisciplinary approach since the following domains are met: multimedia interfaces, data mining and e-Learning. A large variety of learners with possible very different background and goals may access an e-Learning system. This situation yields to the necessity that the interface to by dynamically build according with the current state of the learner. The business logic that decides which resources are available for the learner is based on Bayesian network learning.**

Introduction

THIS paper presents a custom procedure for building high quality personalized multimedia interface within an e-Learning environment.

E-Learning domain has received great amount of effort in last decade. E-learning represents a modern form of conducting education. The e-Learning domain developed greatly due to enormous development of Internet technologies. There are many areas in which e-Learning has progressed. One of the most important areas regard building storing and delivering e-Learning materials, assessment and monitoring of student progress, building recommender systems for learners. This paper is closely related with the last domain.

One of the main characteristics of traditional learning lays in the guidance offered by the professor to the learner. With time, professors gain experience and thus are able to guide learners according with their background and abilities. In education, this ability is highly appreciated and can make the difference in a context where learning resources are similar.

In same manner, e-Learning tries to emulate the experience and the ability of the real professor. Of course, human characteristics are very hard to be modeled and that is why the goal of the presented work is not an easy one yet very challenging.

The first step that needs to be accomplished represents setting up the input and the output. The input is represented by various types of data. The e-Learning context is represented by the e-Learning resources. This also regards the way e-Learning materials are structured. Another important input is represented by the actions performed by learners. All performed actions are important in the way that they will provide important information regarding the behavior of the learners. This will represent in a hard and the structured form the experience of the crowd. The core idea of the paper is represented by a custom representation of this data such that high quality personalized interface may be obtained. Thus, the output of the presented procedure has as output the obtained interface and more exactly a list of resources that need to be accessed. There will be obtained also a ranking of needed resources thus leading to a dynamic learning path that may be created for a certain learner. Under these circumstances the following issues need a great deal of attention: the employed methods, the e-Learning infrastructure, the input data and the analysis process itself.

The main analysis methods are Concept Maps [1, 2, 3] and Bayesian Network Learning [4, 5]. These methods are presented in second section. The e-Learning infrastructure is represented by Tesys e-Learning platform [6]. It is presented in third section along with the procedure of obtaining input data for the analysis process. Fourth section will present the analysis process in detail. Section five presents a sample experiment where real data are processed. Finally, in section six there will be presented conclusions and future works.

Analysis Methods

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 [7].

The usage of concept maps has a proper motivation. Using 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 [8].

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 a concept map should be presented to the students at the very first meeting. This will help them to have a good overview regarding what they will study.

Bayesian Networks

A Bayesian network [5] encodes the joint probability distribution of a set of v variables, $\{x_1, x_2, ..., x_v\}$, as a directed acyclic graph and a set of conditional probability tables (CPTs). In this paper we assume all variables are discrete. An instance is represented by a learner from the e-Learning environment. Each instance is described by a set of features which in this context represent the variables. Each node corresponds to a variable, and the CPT associated with it contains the probability of each state of the variable given every possible combination of states of its parents. The set of parents of x_i, denoted π_i, is the set of nodes with an arc to x_i in the graph. The structure of the network encodes the assertion that each node is conditionally independent of its non-descendants given its parents. Thus the probability of an arbitrary event $X = (x_1, x_2, ..., x_v)$ can be computed as

$$P(X) = \prod_{i=1}^{v} P(x_i|\pi_i)$$

In general, encoding the joint distribution of a set of v discrete variables requires space exponential in v; Bayesian networks reduce this to space exponential in $max_{i\in\{1,...,v\}}|\pi_i|$.

Bayesian networks represent a generalization of naïve Bayesian classification. In [9] it was proved that naïve Bayes classification outperforms unrestricted Bayesian network classification for a large number of datasets. Their explanation was that the scoring functions used in standard Bayesian network learning attempt to optimize the likelihood of the entire data, rather than just the conditional likelihood of the class given the attributes. Such scoring results in suboptimal choices during the search process whenever the two functions favor differing changes to the network. The natural solution would then be to use conditional likelihood as the objective function.

That is why, when using Bayesian networks conditional independence of used variables needs a great attention.

E-Learning Infrastructure

So far, e-Learning platforms are mainly concerned with delivery and management of content (e.g. courses, quizzes, exams, etc.). An important feature that misses is represented by the intelligent characteristic. This may be achieved by embedding knowledge management techniques that will improve the learning process.

For running such a process the e-Learning infrastructure must have some characteristics. The process is designed to run at chapter level. This means a discipline needs to be partitioned into chapters. The chapter has to have assigned a concept map which may consist of about 20 concepts. Each concept has assigned a set of documents and a set of quiz questions. There are three tree documents that may be attached to each concept: overview, detailed description and examples. Each concept and each quiz has a weight, depending of its importance in the hierarchy.

Figure 1 presents a general e-Learning infrastructure for a discipline. Once a course manager has been assigned a discipline he has to set up its chapters by specifying their names and their associated concept maps. For each concept managers have the possibility of setting up three documents and one pool of questions.

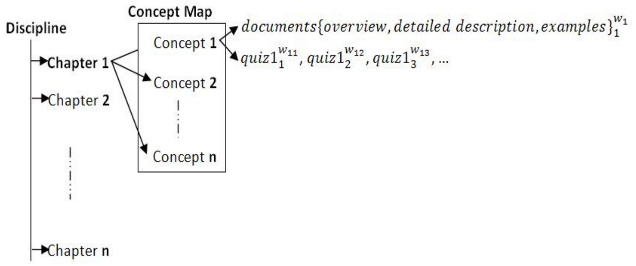

Discipline
Concept Map
Chapter 1
Chapter 2
Chapter n
Concept 1
Concept 2
Concept n
$documents\{overview, detailed\ description, examples\}_1^{w_1}$
$quiz1_1^{w_{11}}, quiz1_2^{w_{12}}, quiz1_3^{w_{13}}, ...$

Fig. 1. General structure of a discipline

When the discipline is fully set, the learning process may start for learners. Any opening of a document and any test quiz that is taken by a learner is registered. The business logic of document retrieval tool will use this data for determining the moment when it is able to determine the document (or the documents) that are considered to need more attention from the learner. The course manager specifies the number of questions that will be randomly extracted for creating a test or an exam.

Let us suppose that for a chapter the professor created 50 test quizzes and he has set to 5 the number of quizzes that are randomly withdrawn for testing and 15 the number of quizzes that are randomly withdrawn for final exam. It means that when a student takes a test from this chapter 5 questions from the pool of test question are randomly withdrawn. When the student takes the final examination at the discipline from which the chapter is part, 15 questions are randomly withdrawn. This manner of creating tests and exams is intended to be flexible enough for the professor. This means, the professor may easily manage the test and exam questions that belong to a chapter. Also, tests and exams composition may be easily managed by professors through custom settings. The difficulty of created test and exam may be controlled with the weights that were assigned to concepts and quizzes.

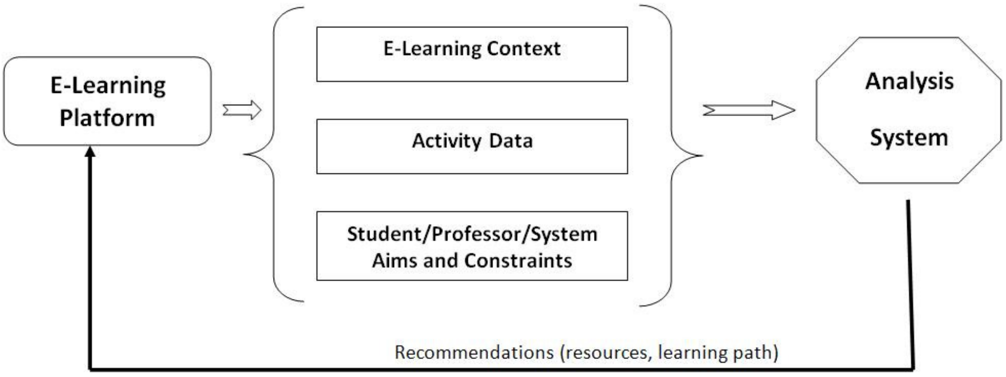

E-Learning Platform
E-Learning Context
Activity Data
Student/Professor/System Aims and Constraints
Analysis System
Recommendations (resources, learning path)

Fig. 2. General view of analysis process

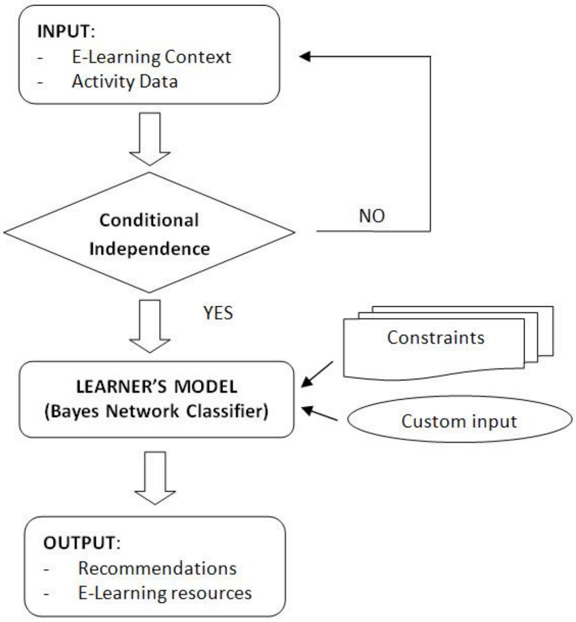


Fig. 3. Detailed view of analysis process

Analysis Process

The analysis process runs along the served e-Learning platform. The e-Learning platform is supposed to be able to provide in a standard format data regarding the context, the performed activity by learners and the aims/constraints provided by learners, professors or system administrator itself.

The e-Learning context represents the set of e-Learning resources that are available for a certain chapter of a discipline. The data that represents the context regards the concept map associated with the chapter along with resources associated to each concept or phrase from the concept map. The resources are represented by documents and quizzes as presented in section three.

The analysis system works as a service that loads the e-Learning context provided by the e-Learning platform and performs updates in a scheduled manner regarding performed activities and the constraints provided by learners, professors or administrator of the e-Learning platform. The constraints work as threshold within the analysis process.

The first step regards checking the conditional independence of attributes. If this condition does not hold than the input must be reviewed. This might mean changes regarding the attributes or even data pruning.

Once the conditional independence of attributes is met the learner's model is build. It will represent the "ground truth" against which any custom request will be evaluated. The custom input regards personal data of a certain learner. It may be regarded as the current status of the learner.

The final outcome of the analysis process is represented by the recommendations and/or a list of resources that need more attention from the learner.

The interface of the learner will be dynamically loaded with links to needed resources thus obtaining a personalized interface

Setup and Experiment

The presented experiment consists in an off-line step by step running of the analysis procedure with real data obtained from Tesys e-Learning platform.

The context has an xml representation. Below it is presented a sample of the xml file representing Computer Science program, Algorithms and Data Structures discipline, Binary Search Trees and Height Balanced Trees chapters.

```
  <module>
  <id>1</id>
    <name>Computer Science</name>
    <discipline>
     <id>1</id>
     <name>Algorithms and Data Structures</name>
     <chapter>
       <id>1</id>
       <name>Binary Search Trees</name>
       <concepts>
         <concept>
           <id>1</id>
           <name>BST</name>
         </concept>
         <concept>
           <id>2</id>
           <name>Node</name>
         </concept>
         ….
       </concepts>
       <quiz>
         <id>1</id>
         <text>text quiz 1</text>
         <visibleAns>abcd</visibleAns>
         <cotectAns>a</ cotectAns >
         <conceptId>1</ conceptId >
       </quiz>
       ……
    </chapter>
      <chapter>
       <id>2</id>
       <name>Height Balanced Trees</name>
    </chapter>
    ……
    </discipline>
</module>
```

It may be observed that each chapter has associated a set of concepts and each quiz has associated a certain concept.

Figure 4 presents the concept map associated with the Binary Search Tree chapter.

The data representing the activities performed by learners needs to be obtained. Firstly, the parameters that represent a learner and their possible values must be defined. For this study the parameters are: *nLogings* – the number of entries on the e-Learning platform; *nTests* – the number of tests taken by the learner; *noOfSentMessages* – the number of sent messages to professors; *chapterCoverage* – the weighted chapter coverage from the testing activities. Their computed values a scaled to one of the following possibilities: VF – very few, F – few, A – average, M – many, VM – very many. The number of attributes and their meaning has a great importance for the whole process since irrelevant attributes may degrade classification performance in sense of relevance. On the other hand, the more attributes we have the more time the algorithm will take to produce a result. Domain knowledge and of course common sense are crucial assets for obtaining relevant results.

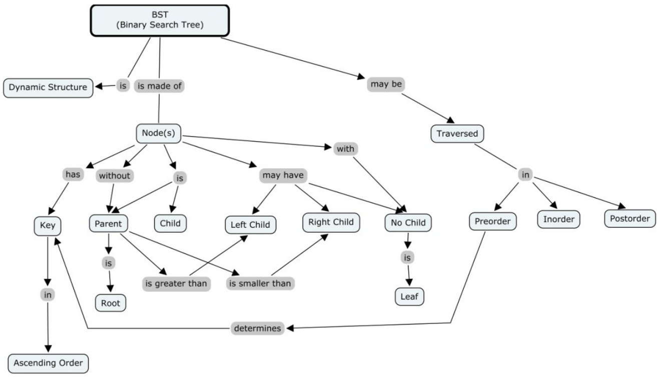


Fig. 4. Binary Search Tree Concept Map

The preparation gets data from the database and puts it into a form ready for processing of the model. Since the processing is done using custom implementation, the output of preparation step is in the form of an *arff* file. Under these circumstances, we have developed an offline Java application that queries the platform's database and crates the input data file called *activity.arff*. This process is automated and is driven by a property file in which there is specified what data/attributes will lay in *activity.arff* file.

For a student in our platform we may have a very large number of attributes. Still, in our procedure we use only four: the number of logings, the number of taken tests, the number of sent messages and the weighted chapter coverage from the testing activities. Here is how the arff file looks like:

```
@relation activity
@attribute nLogings {VF, F, A, M, VM}
@attribute nTests {VF, F, A, M, VM}
@attribute noOfSentMessages {VF, F, A, M, VM}
@attribute chapterCoverage {VF, F, A, M, VM}
@data
VF, F, A, A,
F, A, M, VM,
A, M, VM, A, V,
VM, A, VM, M,
```

As it can be seen from the definition of the attributes each of them has a set of five nominal values from which only one may be assigned. The values of the attributes are computed for each student that participates in the study and are set in the @data section of the file. For example, the first line says that the student logged in very few times, took few tests, sent an average number of messages to professors and had average chapter coverage.

In order to obtain relevant results, we pruned noisy data. We considered that students for which the number of logings, the number of taken tests or the number of sent messages is zero are not interesting for our study and degrade performance; this is the reason why all such records were deleted.

Once the dataset is obtained the conditional independence is assessed. This is necessary because the causal structure of attributes needs to be revealed. If conditional independency is identified between two variables then there will be no arrow between those two variables.

As metric regarding the conditional independence there are estimated expected utilities. This metric will specify how well a Bayesian network performs on a given dataset. Cross validation provides an out of sample evaluation method to facilitate this by repeatedly splitting the data in training and validation sets. A Bayesian network structure can be evaluated by estimating the network's parameters from the training set and the resulting Bayesian network's performance determined against the validation set. The average performance of the Bayesian network over the validation sets provides a metric for the quality of the network.

Running Bayes Net algorithm in weka [10] produced the following output:

```
    === Run information ===

Scheme:       weka.classifiers.bayes.BayesNetB -S BAYES
-A 0.5 -P 100000
    Relation:     activity
    Instances:    261
    Attributes:   4
                  nLogings
                  nTests
                  noOfSentMessages
                  chapterCoverage
    Test mode:    10-fold cross-validation
    === Classifier model (full training set) ===
    Bayes Network Classifier
    Using ADTree
    #attributes=4 #classindex=3
    Network structure (nodes followed by parents)
    nLogings(5): chapterCoverage nTests
    nTests(5): chapterCoverage
    noOfSentMessages(5): chapterCoverage nTests
    chapterCoverage(5):
    LogScore Bayes: -77.14595781124575
    LogScore MDL: -597.9372820270846
    LogScore ENTROPY: -287.4073451362291
    LogScore AIC: -511.4073451362291
    -S
    Time taken to build model: 0.12 seconds
    === Stratified cross-validation ===
    === Summary ===
    Correctly Classified Instances        228        87.5   %
    Incorrectly Classified Instances       33        12.5   %
    Kappa statistic                    0.7881
    Mean absolute error                0.0814
    Root mean squared error              0.1909
    Relative absolute error           31.9335 %
    Root relative squared error          55.2006 %
    Total Number of Instances              16
    === Detailed Accuracy By Class ===

    TP Rate  FP Rate  Precision  Recall  F-Measure  Class
     0        0        0         0        0       VF
     0        0        0         0        0       F
     0.889    0.143    0.889     0.889    0.889   A
     0.75     0        1         0.75     0.857   M
     1        0.077    0.75      1        0.857   VM
    === Confusion Matrix ===
     a   b   c    d    e   <-- classified as
     0   0   0    0    0    | a = VF
     0   0   0    0    0    | b = F
     0   0   150  0    15   | c = A
     0   0   18   50   0    | d = M
     0   0   0    0    28   | e = VM
```

The Bayesian network obtained in weka has the following graph.

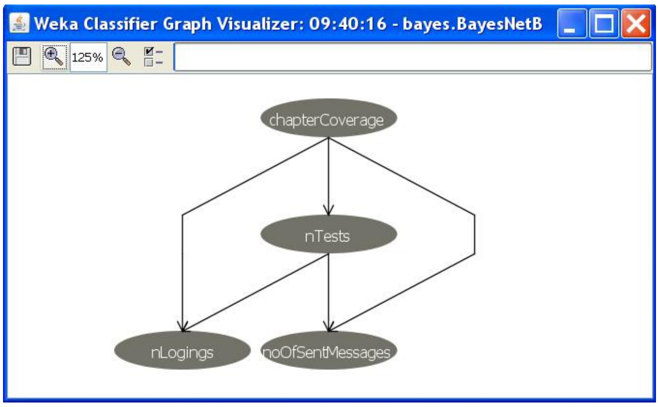

Weka Classifier Graph Visualizer: 09:40:16 - bayes.BayesNetB
125%
chapterCoverage
nTests
nLogings
noOfSentMessages

Fig. 5. Detailed view of analysis process

As it can be seen in above figure the chapter coverage is the variable with greatest conditional dependence towards all other variables. On the other hand, variables *nLogings* and *noOfSentMessages* are conditional independent which means they need to be used in further developments.

Once the Bayes Net has been obtained it may be used for obtaining the items that compose the interface for the learner. The procedure *findItems* determines the needed resources.

```
Items procedure findItems (LearnerModel LM, Constrtaints CS, Lerner l) {
    Class C = classify (l,LM);
    Class D = findClass (LM, C, CS);
    Items items = determineItems(C, D);
    return items;
}
```

Firstly, the learner is classified against the current learner model. Thus, the actual class to which the learner belongs is determined. Secondly, the destination class D is determined taking into consideration the current learner model, the class of the learner and the constraints set up by system professor or learner himself. Finally there is determined the set of items that need to be accessed by learner by analyzing classes C and D. As general idea, there are determined the items where class D is better representation than in class C. Such a metric may also rank the resources. Firstly, there are presented the resources with smaller distance between classes. It is supposed that these resources need immediate attention from the learner.

Conclusions and Future Works

This paper presents custom data analysis process which has as main outcome obtaining a personalized interface for an e-Learning platform.

The main inputs of the process are: the context of the platform, the activity data, the constraints of the involved parties and data regarding the learner for which the personalized interface is built.

The activity data managed by the analysis process is represented by actions performed by learners within the e-Learning environment. From the great variety of performed actions there were taken into consideration only four: the number of entries on the e-Learning platform, the number of tests taken by the learner, the number of sent messages to professors and the weighted chapter coverage from the testing activities.

The business logic uses Bayes Network Classifier implemented in weka for building the learner's model against which any learner is classified. For obtaining sound classification results the conditional independence is verified.

Once the conditional independence is met there may be started the procedure for obtaining the items that will be recommended. The procedure classifies the learner, finds the destination class and determines the items. Each item represents a resource (document or quiz) that needs attention from the learner.

As future works, there are some issues that need to be addressed. One issue regards the conditional independence assessment of variables. When this condition is not met the procedure for data pruning and feature selection may need improvement.

Another issue regards the granularity with which items are obtained by *findItems* procedure. Optimization of complexity calculus for determining the destination class and especially the set of items is needed.

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] Novak, J. D., Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations. Mahwah, NJ: Lawrence Erlbaum Associates, 1998.

[2] McDaniel,E., Roth, B., and Miller, M. “Concept Mapping as a Tool for Curriculum Design”, Issues in Informing Science and Information Technology.

[3] Vecchia, L., Pedroni, M., Concept Maps as a Learning Assessment Tool. Issues in Informing Science and Information Technology, Volume 4, 2007.

[4] D. Heckerman. A tutorial on learning with bayesian networks. In M. Jordan, editor, Learning in Graphical Models. MIT Press, Cambridge, MA, 1999.

[5] Pearl, J., Probabilistic reasoning in intelligent systems: Networks of plausible inference. San Francisco, CA: Morgan Kaufmann, 1988.

[6] Burdescu, D.D., Mihăescu, M.C., 2006. Tesys: e-Learning Application Built on a Web Platform. In Proceedings of International Joint Conference on e-Business and Telecommunications, pp. 315-318, Setubal, Portugal. INSTICC Press.

[7] MAC (2010), http://mac.concord.org

[8] Kolodner, J. L., Camp, P. J., Crismond, D., Fasse, B., Gray, J., Holbrook, J., Puntambekar, S., and Ryan, M. Problem-based learning meets case-based reasoning in the middle-school science classroom: Putting learning by design into practice, The Journal of the Learning Sciences, 12 (4), 2003, 495-547.

[9] Friedman, N., Geiger, D., & Goldszmidt, M., Bayesian network classiers, Machine Learning, 29, 131-163, 1997.

[10] Weka (2010), www.cs.waikato.ac.nz/ml/weka

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 735–740

ISBN 978-83-60810-27-9
ISSN 1896-7094

A Graphical Interface for Evaluating Three Graph-Based Image Segmentation Algorithms

Gabriel Mihai
University of Craiova, Faculty of
Automation, Computers and
Electronics, Craiova, Romania
Email:mihai_gabriel@software.ucv.ro

Alina Doringa, Liana Stanescu
University of Craiova, Faculty of
Automation, Computers and
Electronics, Craiova,Romania
Email:alinadoringa@hotmail.com,
stanescu_liana@software.ucv.ro

Abstract—**Image segmentation has an essential role in image analysis, pattern recognition and low-level vision. Since multiple segmentation algorithms exists in literature, numerical evaluations are needed to quantify the consistency between them. Error measures can be used for consistency quantification because are allowing a principled comparison between segmentation results on different images, with differing numbers of regions, and generated by different algorithms with different parameters. This paper presents a graphical interface for evaluating three graph-based image segmentation algorithms: the color set back-projection algorithm, an efficient graph based image segmentation algorithm known also as the local variation algorithm and a new and original segmentation algorithm using a hexagonal structure defined on the set of image pixels.**

I. Introduction

IMAGE segmentation is one of the most difficult and challenging tasks in image processing and can be defined as the process of dividing an image into different regions such that each region is homogeneous while not the union of any two adjacent regions.

The consistency between segmentations must be evaluated because no unique segmentation of an image can exist. If two different segmentations arise from different perceptual organizations of the scene, then it is fair to declare the segmentations inconsistent [3].

This paper presents a graphical interface used for an objective and quantitative evaluation of three graph-based segmentation algorithms that will be described further. For each of these algorithms three characteristics are examined [4]: correctness, stability with respect to parameter choice, stability with respect to image choice.

The evaluation of the algorithms is based on two metrics (GCE, LCE) defined in [3] which can be used to measure the consistency of a pair of segmentations. These measures allows a comparison between segmentation results on different images, with differing numbers of regions, and generated by different algorithms with different parameters. In order to establish which algorithm produces better results these are compared with manual segmentations of the same image.

For searching certain structures graph-based segmentation methods such as minimum spanning tree [6] [7], or minimum cut [8] [9] are using an edge weighted graph constructed on the image pixels. The Graph-based clustering algorithms use the concept of homogeneity of regions. For color segmentation algorithms the homogeneity of regions is color-based, and thus the edge weights are based on color distance.

For obtaining the image regions early graph-based methods have used fixed thresholds and local measures. Using these values larger edges belonging to a minimum spanning tree were beaked. The problem of fixed threshold [10] can be avoided by determining the normalized weight of an edge using the smallest weight incident on the vertices touching that edge.

Other methods presented in [6] [7] use an adaptive criterion that depends on local properties rather than global ones. The methods based on minimum cuts in a graph are designed to minimize the similarity between pixels that are being split [8], [9]. An alternative to the graph cut approach is to look for cycles in a graph embedded in the image plane [11].

The color and texture models are used by most graph-based segmentation approaches but these homogeneity criteria have some drawbacks for object extraction.

In [12] a source of additional information denoted by the term of syntactic features is presented, which represent geometric properties of regions and their spatial configurations such as homogeneity, compactness, regularity, inclusion or symmetry.

II. The color set back-projection algorithm

The color set back-projection algorithm proposed in [5] is a technique for the automated extraction of regions and representation of their color content. This algorithm is based on color sets which provide an alternative to color histograms for representing color information. Their utilization is possible only when salient regions have few equally prominent colors. Each pixel from the initial image is represented in the HSV color space. The quantized colors from 0 to 165 are stored in a matrix that is filtered by a 5x5 median filter for eliminating the isolated points. The back-projection process requires several stages:

a) Color set selection - candidate color sets are selected first with one color, then with two colors, etc., until the salient regions are extracted

b) Back-projection onto the image - a transformation from the RGB color space to HSV color space and a quantization of the HSV color space at 166 colors is performed for each segmented image.

c) Thresholding and labeling image - insignificant color information is reduced and the significant color regions are evidentiated, followed by the generation, in automatic way, of the regions of a single color, of the two colors, of three colors.

In the implementation of the color set back-projection algorithm that can be found it in [1] [13]. After processing the global histogram of the image, and the color set are provided. The process of regions extraction is using the filtered matrix and it is a depth – first traversal described in pseudo-cod in the following way:

```
Procedure FindRegions (Image I, colorset C)
    InitStack(S)
    Visited = ∅
    for *each node P in the I do
    if *color of P is in C then
        PUSH(P)
        Visited ← Visited ∪ {P}
        while not Empty(S) do
            CrtPoint ←POP()
            Visited ← Visited ∪ {CrtPoint}
              For *each unvisited neighbor S of CrtPoint do
                if *color of S is in C then
                   Visited ← Visited ∪ {S}
                   PUSH(S)
              end
            end
        end
    *Output detected region
      end
    end
```

The total running time for a call of the procedure FindRegions (Image *I*, colorset *C*) is $O(m^2 * n^2)$ where *m* is the width and *n* is the height of the image [1][13].

III. LOCAL VARIATION ALGORITHM

This algorithm described in [6] is using a graph based approach for the image segmentation process. The pixels are considered the graph nodes so in this way it is possible to define an undirected graph G = (V, E) where the vertices v_i from V represent the set of elements to be segmented. Each edge (v_i, v_j) belonging to E has associated a corresponding weight $w(v_i, v_j)$ calculated based on color, which is a measure of the dissimilarity between neighboring elements v_i and v_j.

A minimum spanning tree is obtained using Kruskal's algorithm. The connected components that are obtained represent image's regions. It is supposed that the graph has *m* edges and *n* vertices. This algorithm is described also in [14] where it has four major steps that are presented below:

1) Sort $E=(e_1,...,e_m)$ such that $|e_t|<|e_{t'}| \;\forall t<t'$

2) Let $S^o=(\{x_1\},...,\{x_n\})$ be each initial cluster containing only one vertex.

3) For $t=1,...,m$

   a) *Let* x_i *and* x_j *be the vertices connected by* e_t

   b) *Let* $C_{x_i}^{t\text{-}1}$ *be the connected component containing point* x_i *on iteration t-1 and* $l_i = max_{mst}\, C_{x_i}^{t\text{-}1}$ be the longest edge in the minimum spanning tree of $C_{x_i}^{t\text{-}1}$ .Likewise for l_j.

   c) *Merge* $C_{x_i}^{t\text{-}1}$ *and* $C_{xj}^{t\text{-}1}$ *if*

   $$|e_t| < \min\{\, l_i + \frac{k}{C_{x_i}^{t\text{-}1}}, l_j + \frac{k}{C_{x_j}^{t\text{-}1}} \} \textit{ where k is a constant.}$$

4) $S=S^m$

The existence of a boundary between two components in segmentation is based on a predicate D. This predicate is measuring the dissimilarity between elements along the boundary of the two components relative to a measure of the dissimilarity among neighboring elements within each of the two components. The internal difference of a component C ⊆ V was defined as the largest weight in the minimum spanning tree of a component MST(C, E):

$Int(C) = {}_{e\in MST(C,E)} max\, w(e)$. The difference between two components $C_1, C_2 \subseteq V$ is defined as the minimum weight edge connecting the two components:

$$Dif(C_1,C_2)=min(w(v_i,v_j))$$
$$v_i \in C_1, v_j \in C_2, (v_i,v_j) \in E$$

A threshold function is used to control the degree to which the difference between components must be larger than minimum internal difference. The pair wise comparison predicate is defined as:

$$D(C_1,C_2)=\{ \begin{matrix} \textit{true if } Dif(C_1,C_2)>MInt(C_1,C_2) \\ \textit{false otherwise} \end{matrix}$$

where the minimum internal difference Mint is defined as:

$$MInt(C_1,C_2)=min(Int(C_1)+\tau(C_1), Int(C_2)+\tau(C_2)).$$

The threshold function was defined based on the size of the component: $\tau(C)=k/|C|$.The k value is set taking into account the size of the image. The algorithm for creating the minimum spanning tree can be implemented to run in $O(m\log m)$ where m is the number of edges in the graph.

IV. IMAGE SEGMENTATION USING A HEXAGONAL STRUCTURE DEFINED ON THE SET OF PIXELS

The technique is based on a new and original utilization of pixels from the image that are integrated into a network type graph [2]. The hexagonal network structure on the image pixels, as presented in figure 1 was selected to improve the running time required by the algorithms used for segmentation and contour detection.

The hexagonal structure represents a grid-graph and for each hexagon h in this structure there exist 6-hexagons that are neighbors in a 6-connected sense. The time complexity of the algorithms is reduced by using hexagons instead of pixels as elementary piece of information. The index of each hexagon is stored in a vector of numbers $[1..N]$, where *N*, the number of hexagons, is calculated using the formula [2]:

$$N = \frac{height - 1}{2} * (\frac{width - width \bmod 4}{2} - 1)$$

where *height* and *width* represent the height and the width of the image.

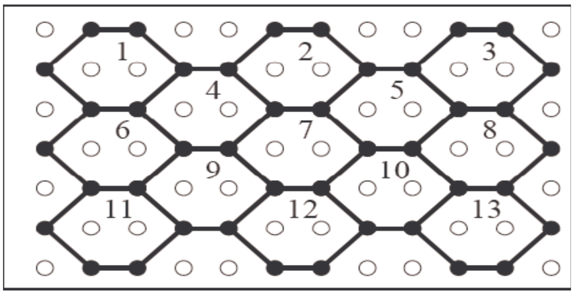


Fig.1. Hexagonal structure constructed on the image pixels

With each hexagon two important attributes are associated:

a) The dominant color - eight pixels contained in a hexagon are used: six pixels from the frontier and two from interior

b) The gravity center

The pixels of image are split into two sets, the set of pixels representing the vertices of hexagons and the set of complementary pixels. These two lists are used as inputs for the segmentation algorithm.

Based on the algorithms proposed in [2] the list of salient regions is obtained from the input image using the hexagonal network and the distance between two colors in HSV space color. The color of a hexagon is obtained using a procedure called *sameVertexColour*. The execution time of this procedure is constant because all calls are constant in time processing. The color information is used by the procedure *expandColorArea* to find the list of hexagons that have the same color. To expand the current region it is used a procedure containing as input:

a) The current hexagon h_i ,

b) L_1 and L_2 lists,

c) The list of hexagons V

d) The current region index *indexCrtRegion*

e) The current color index *indexCrtColor*.

The output is represented by a list of hexagons with the same color *crtRegionItem.* The running time of the procedure *expandColourArea* is $O(n)$ where n is the number of hexagons from a region with the same color.

The list of regions is obtained using the *listRegions* procedure The input of this procedure contains:

a) The vector V representing the list of hexagons

b) The lists L_1 and L_2

The output is represented by a list of colors pixels and a list of regions for each color.

```
Procedure listRegions ( $V, L_1, L_2$ )
colourNb ←0;
for i ←1 to n do
    initialize crtRegionItem;
    if not(visit( $h_i$ )) then
        crtColorHexagon ←sameVertexColour ( $L_1, L_2, h_i$ );
        if crtColorHexagon.sameColor then
            k ← findColor(crtColorHexagon.color);
            if k < 0 then
                add new color $c_{colourNb}$ to list C;
                k ←colourNb++;
                indexCrtRegion ←0;
            else
                indexCrtColor ←k;
                indexCrtRegion ←findLastIndexRegion(indexCrtColor);
                indexCrtRegion++;
            End
            hi.indexRegion ←indexCrtRegion;
            hi.indexColor ←k;
            add $h_i$ to crtRegionItem;
            expandColourArea( $h_1, L_1, L_2, V$ ,indexCrtRegion,
                indexCrtColor; crtRegionItem);
            add new region crtRegionItem to list of element k from C
        end
    end
end
```

The running time of the procedure list Regions is $O(n^2)$, where n is the number of the hexagons network [2].

V. Segmentation error measures

To evaluate a segmentation algorithm it is needed to measure the accuracy, the precision and the performance. When multiple segmentation algorithms are evaluated some metrics are needed to establish which algorithm produce better results.

In [3] are proposed two metrics tolerant to refinement that can be used to evaluate the consistency of a pair of segmentations. A segmentation error measure takes two segmentations S_1 and S_2 as input, and produces a real valued output in the range [0..1] where zero signifies no error. For a given pixel p_i two segments S_1 and S_2 containing that pixel, are considered. If one segment is a proper subset of the other, then the pixel lies in an area of refinement, and the local error should be zero. If there is no subset relationship, then the two regions overlap in an inconsistent manner. In this case, the local error should be non-zero. Let \ denote set difference, and $|x|$ the cardinality of set x. If $R(S,p_i)$ is the set of pixels corresponding to the region in segmentation S that contains pixel p_i, the local refinement error is defined in [3] as:

$$E(S_1,S_2,p_i)=\frac{|R(S_1,p_i)\setminus R(S_2,p_i)|}{R(S_1,p_i)}$$

This local error measure is not symmetric and it encodes a measure of refinement in one direction only. Given this local refinement error in each direction at each pixel, there are two natural ways to combine the values into an error measure for the entire image. Global Consistency Error (GCE) forces all local refinements to be in the same direction. Local Consistency Error (LCE) allows refinement in different directions in different parts of the image. Let n be the number of pixels [3]:

$$GCE(S_1,S_2)=\frac{1}{n}\min\{\quad {}_i E(S_1,S_2,p_i),E(S_2,S_1,p_i)\}$$

$$LCE(S_1,S_2)=\frac{1}{n}\quad {}_i \min(E(S_1,S_2,p_i),E(S_2,S_1,p_i))$$

$LCE \leq GCE$ for any two segmentations and it is clear that GCE is a tougher measure than LCE. In [3] are shown that, as expected, when pairs of human segmentations of the same image are compared, both the GCE and the LCE are low; conversely, when random pairs of human segmentations are compared, the resulting GCE and LCE are high. If the pixel wise minimum is replaced by a maximum it is obtained a new measure named Bidirectional Consistency Error (BCE) that is not tolerating the refinement. This measure is evaluated using

$$BCE(S_1,S_2)=\frac{1}{n}\quad {}_i \max(E(S_1,S_2,p_i),E(S_2,S_1,p_i)).$$

If an image is interpreted as a set *O* of pixels and the segmentation as a clustering of *O* we can apply measures for comparing clusters. As described in [17] we can have two types of distances available for clusters: the distance between clusters evaluated by counting pairs and the distance between clusters evaluated using set matching. For the first case are assumed two clusters C_1 and C_2 belonging to a set *O* of objects and also all pairs of distinct objects (o_i,o_j) from *OxO.*

Each pair can be found into one of the four categories:

a) in the same cluster under both C_1 and C_2 (the total number of these pairs is represented by *N11*)

b) in different clusters under both C_1 and C_2 (*N*00),

c) in the same cluster under C_1 but not C_2 (*N*10),

d) in the same cluster under C_2 but not C_1 (*N*01).

Based on these assumptions the following statement is obtained: $N_{11}+N_{00}+N_{10}+N_{01}=n(n-1)/2$.

Based on these numbers multiple distances of the first type were defined. One of the distances described in [18] is the Rand index evaluated as $R(C_1,C_2)=1-\frac{N_{11}+N_{00}}{n(n-1)/2}$.

A perfect matching between two clusters is implied by a 0 value. Another index called the Jacard index [19] is evaluated as $J(C_1,C_2)=1-\frac{N_{11}}{N_{11}+N_{10}+N_{01}}$

For the second type the comparison criteria is based on the following term $a(C_1,C_2)=\quad {}_{ci\in C_1}\max_{cj\in C_2}|c_i\cap c_j|$.

This term measures the matching degree between C_1 and C_2 has a value of *n* when the two clusters are equal. The Dongan index [20] is evaluated as $D(C_1,C_2)=2n-a(C_1,C_2)-a(C_2,C_1)$.

VI. EXPERIMENTAL RESULTS

For each image three major steps are required in order to calculate GCE, LCE, BCE, Dongen index and Rand index values:

a) Obtain the image regions by applying segmentation algorithms (color set back-projection – CS, the algorithm based on the hexagonal structure – HS, the local variation algorithm – LV) and the regions manually segmented -MS

b) Store the information obtained for each region in the database

c) Use a graphical interface to easily calculate GCE and LCE values and to store them in the database for later statistics

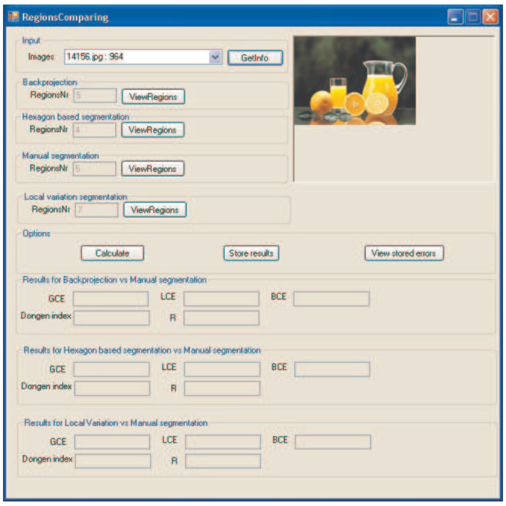
All algorithms are using a configurable threshold value to keep only the regions that have a number of pixels greater than this value. In this way only representative regions are saved in the database and used later in the experiments.

The experiments were made on a working set containing 300 images selected from IAPR-TC12 [15] and Berkeley [16] segmentation datasets. All values were calculated using the graphical interface presented in figure 1.

This application is offering to the user a list containing the name of all images that were segmented and that have regions stored in the database. After selecting from the list the name of an image the user needs to press *GetInfo* button to retrieve de information stored in the database for that image. In that moment the text boxes containing the number of regions obtained with each algorithm are filled.

For each algorithm a button named *ViewRegions* is available. By pressing this button a new form is shown containing the obtained regions as presented in image 2. This form helps the user to view the regions detected by each algorithm and to make an empirical evaluation of the performance. Because this evaluation is subjective we need a numerical evaluation.

Since the information about regions is available the user can press Calculate button. In this moment the error measures will be calculated and shown to the user. If the user decides that these values are needed later he can choose to store them in the database by pressing *Store results* button. To compare the obtained results with previous results stored in the database *View stored results* button should be pressed. In this way it is possible to evaluate what algorithm is producing better results.



Images: 14156.jpg : 964

Backprojection RegionsNr: 5

Hexagon based segmentation RegionsNr: 4

Manual segmentation RegionsNr: 5

Local variation segmentation RegionsNr: 7

Fig.1. The graphical interface used for errors calculation

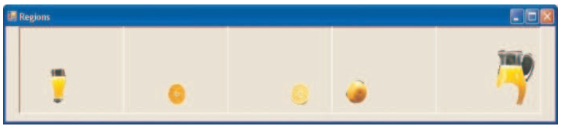
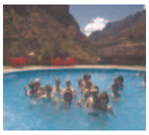
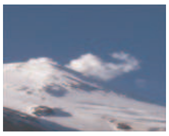


Fig.2. The form containg image regions

In table 1 are presented the images for which we will present some experimental results.

TABLE I. IMAGES USED IN EXPERIMENTS

Images		
1 	2 	3 
4 	5 	6

In table 2 can be seen the number of regions resulted from the application of the segmentation algorithms.

TABLE II .THE NUMBER OF REGIONS DETECTED FOR EACH ALGORITHM

Img.No	CS	HS	LV	MS
1	4	3	6	5
2	5	4	7	6
3	5	3	5	3
4	6	4	6	3
5	4	2	5	2
6	5	2	6	3

In table 3 are presented the GCE values calculated for each algorithm.

TABLE III .GCE VALUES CALCULATED FOR EACH ALGORITHM

Img.No	GCE - CS	GCE-HS	GCE - LV
1	0.18	0.09	0.24
2	0.36	0.10	0.28
3	0.18	0.09	0.11
4	0.10	0.09	0.09
5	0.11	0.09	0.10
6	0.04	0.02	0.03

In table 4 are presented the LCE values calculated for each algorithm.

TABLE IV. LCE VALUES CALCULATED FOR EAC H ALGORITHM

Img.No.	LCE-CS	LCE-HS	LCE -LV
1	0.11	0.07	0.15
2	0.18	0.12	0.17
3	0.17	0.07	0.08
4	0.09	0.11	0.09
5	0.05	0.08	0.10
6	0.04	0.02	0.02

Bellow is presented the regions obtained for the third image after applying manual segmentation and the three algorithms presented above.

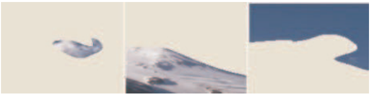


Fig.3. Regions from manual segmentation

Fig.4.Regions from the color set back-projection algorithm

Fig.5. Regions from hexagonal structure segmentation

Fig.6. Regions from the local variation algorithm

The error measures presented in tables three and four are calculated in relation with manual segmentation which is considered the ground truth. It can be observed that the values for GCE and LCE are lower in case of hexagonal segmentation. For example for the first analyzed image GCE = 0.09 and LCE =0.07 in the case of hexagonal segmentation, GCE=0.18 and LCE=0.11 for the color set back-projection algorithm, GCE=0.24 and LCE=0.15 for the local variation algorithm. The error measures, for almost all tested images, have smaller values in case of the hexagonal segmentation so this algorithm is a good refinement of the manual segmentation.

VI. Conclusion

In this paper it is proposed a graphical interface for evaluating three graph-based image segmentation algorithms: the color set back-projection algorithm, the image segmentation using a hexagonal structure defined on the set of image pixels and the local variation algorithm. Error measures like GCE, LCE are used to evaluate the accuracy of each segmentation produced by the algorithms. The values for the error measures are calculated using the described application specially created for this purpose. The proposed error measures quantify the consistency between segmentations of differing granularities. Because human segmentation is considered truth segmentations the error measures are calculated in relation with manual segmentation. The GCE and LCE demonstrate that the image segmentation based on a hexagonal structure produces a better segmentation than the other methods. In the future work the comparative study will be effectuated on medical images.

IV. References

[1] D. D. Burdescu, L. Stanescu "A New Algorithm for Content-Based Region Query in Multimedia Databases" Congres DEXA 2005 : Database and expert systems applications Copenhagen, 22-26 August 2005 , vol. 3588, pp. 124-133.

[2] D. D. Burdescu, M. Brezovan, E. Ganea, and L. Stanescu "A New Method for Segmentation of Images Represented in a HSV Color Space" ACIVS 2009: Bordeaux, France, pp. 606-617 .

[3] D. Martin, C. Fowlkes, D. Tal, J. Malik "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics " In IEEE (ed.), Proceedings of the Eighth International Conference On Computer Vision (ICCV-01), July 7-14, 2001, Vancouver, British Columbia, Canada, vol. 2, pp. 416–425.

[4] R. Unnikrishnan, C. Pantofaru, and M. Hebert. "Toward Objective Evaluation of Image Segmentation Algorithms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 6, June, 2007, pp. 929-944.

[5] 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)

[6] P.F. Felzenszwalb, W.D. Huttenlocher: "Efficient Graph-Based Image Segmentation", Intl. Journal of Computer Vision, 59(2), pp. 167–181 (2004)

[7] L. Guigues, , L.M Herve, L.-P. Cocquerez: "The hierarchy of the cocoons of a graph and its application to image segmentation." Pattern Recognition Letters, 24(8), pp. 1059–1066 (2003)

[8] J. Shi, J. Malik: "Normalized cuts and image segmentation", Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, pp. 731–737 (1997)

[9] Z. Wu ,R. Leahy: "An optimal graph theoretic approach to data clustering: theory and its application to image segmentation", IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(11),pp. 1101–1113 (1993)

[10] R. Urquhar: "Graph theoretical clustering based on limited neighborhood sets." Pattern Recognition, 15(3), pp 173–187 (1982)

[11] I. Jermyn, H. Ishikawa: "Globally optimal regions and boundaries as minimum ratio weight cycles." IEEE Trans. on Pattern Analysis and Machine Intelligence, 23(8), pp. 1075–1088 (2001)

[12] C.F. Bennstrom, J.R. Casas: "Binary-partition-tree creation using a quasi-inclusion criterion." Proc. of the Eighth International Conference on Information Visualization, London, UK, pp. 259–294, (2004)

[13] L. Stanescu "Visual information. Procecessing, Retrieval and Applications" Sitech Craiova 2008

[14] C.Pantofaru, M.Hebert: "A Comparison of Image Segmentation Algorithms". Technical Report CMU-RI-TR-05-40, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania .(2005).

[15] ImageClef (2009) http://imageclef.org/2010

[16] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html

[17] X Jiang, C Marti, C Irniger, H Bunke (2005) Distance Measures for Image Segmentation Evaluation. In: EURASIP Journal on Applied Signal Processing, vol. 2006, pp. 1—10

[18] W. M. Rand "Objective criteria for the evaluation of clustering methods" In: Journal of the American Statistical Association, vol. 66, 1971, no. 336, pp. 846–-850

[19] A. Ben-Hur, A. Elisseeff, I. Guyon "A stability based method for discovering structure in clustered data" In: Proceedings of 7th Pacific Symposium on Biocomputing (PSB '02*)*, vol. 7, 2002, pp. 6–-17, Lihue, Hawaii, USA

[20] S. van Dongen "Performance criteria for graph clustering and Markov cluster experiments" In: Tech. Rep. INS-R0012,10 EURASIP Journal on Applied Signal Processing Centrum voor Wiskunde en Informatica (CWI), 2000, Amsterdam

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 741–748

ISBN 978-83-60810-27-9
ISSN 1896-7094

Basic Consideration of MPEG-2 Coded File Entropy and Lossless Re-encoding

Kazuo Ohzeki Yuǎn yù Wei
Shibaura Institute of Technology Toyosu,
Koutouku, Tokyo, 135-8548 Japan
Email: {ohzeki,m710101}@sic.shibaura-it.ac.jp

Eizaburo Iwata
Universal Robot Inc.
Toyosu, Aoumi, Kotoku
Tokyo Japan
Email : eiza@urobot.co.jp

Ulrich Speidel
University of Auckland
Tamaki – 331, Morrin Road, Glen
Inne, Auckland New Zealand
Email: ulrich@cs.auckland.ac.nz

Abstract—**Re-encoding of once compressed files is one of the difficult challenges in measuring the efficiency of coding methods. Variable length coding with a variable source delimiting scheme is a promising method for improving re-encoding efficiency. Analyses of coded files with fixed length delimiting and with variable length delimiting are reviewed. Motion vector codes of MPEG-2 encoded files are modified as a variable-to-variable coding point of view. Length, bit-rates, and varieties of videos are examined. The largest file is 16 seconds of D1 full size at 720 ×480 among five video files. By entropy evaluation, an improvement of almost 20% in coding efficiency over the conventional MPEG-2 is obtained.**

I. Introduction

In this paper, for coded files, such as MPEG-2 encoded ones, redundancy will be evaluated to provide re-encoding information. There are four frameworks of encoding according to whether input event length and output bit length is fixed or variable. One example is fixed length input and fixed-length output that is an F-F type. In the same manner, there are three other types of F-V, V-F and V-V. Among these types, V-V is the most general and has the greatest possibilities in realizing the most efficient encoding method [1-2]. However, the V-V type has the difficult problem of how to delimit the input events into most efficient length. Universal coding and arithmetic coding are flexible methods. But they are basically still F-V types. Based on several examples of V-V trials, we will show an example of V-V re-encoding. The method improves a compression ratio of 20% compared to the MPEG-2 standard method.

Examples of V-V coding methods are reported in several papers. Jacob presented that the V-F codes are better than the F-V codes for finite alphabets of K-th order ergodic Markov sources [1]. Yamamoto et al mentioned that a VF code is called proper if the set of parse strings of the code satisfies the prefix condition [3]. He also described that non-proper VF codes should be considered in order to realize efficient VF coding. This implies that variable parsing without considering prefix condition has a large possibility to enhance coding efficiency. Abrahams presented a survey paper on the theoretical literature on fixed-to-variable-length lossless source code trees, called code trees, and on variable-length-to-fixed lossless source code trees, called parse trees [4]. He focused on Huffman coding and Tunstall V-F coding for parsing and making tree methods with a large bibliography for further investigation of algorithmic and performance perspectives. Matsui et al examined the compression ratio of the Tunstall-Huffman code. The Tunstall-Huffman code is a variable-to-variable code. They obtained better results by the Tunstall-Huffman V-V coding than with standalone coding methods such as the Tunstall V-F coding or the Huffman F-V coding [5].

All these V-V coding are theoretical not for video files. MPEG-2 coding of video files is not proved to be an optimum method. In fact one of the authors has shown reduandancy of coded MPEG-2 files using FV codes. In this paper, we will try to enhance to use VF codes for MPEG-2 coded files. Though the aplication is restricted to motion vector parts, coding efficiency of the VV codes is much larger than that of the FV case.

The proposed method is an improvement of the MPEG-2 coding. So it is not compatible with the standard methods.

As for V-V coding, there have been few papers. The reason may be that it is difficult to parse sequences at optimal delimiting points. To cope with this problem, an example of review trials will be presented in this paper. A fixed length analysis of sequences was first implemented. The length of the fixed parsing was incrementally examined. Many heuristic trials were carried out and problems with this analysis will be listed.

To conduct variable parsing of sequences, the T-code generating method [6] was taken up. T-code was introduced by Titchener in 1984 [7]. Though the generating process is only a parsing process, it may be an influential tool to optimal variable length coding. It generates codes by copying pregenerated shorter codes. T-code generating experiments were carried out for more than 50 different video sequences. Efficient parsing is itself efficient coding. How to combine efficiency conditions with a parsing algorithm will be a further problem.

According to coding of motion vectors, Yu et al presented two-dimensional motion vector coding for low bit-rate video phones. However, their Huffman coding was generated by the JPEG procedure and the number of motion vector bits were about only one-frame for full D1 digital video size, 720×480 [8]. Shimizu et al proposed a method using representation of norm and angle for motion vectors. The method was complicated and still two separate codes were used [9]. Matsuda et al proposed a lossless re-encoding scheme for MPEG-1 video. They used an arithmetic coder for both DCT and motion vector data for the MPEG-1 coder [10]. In the following sections, an exemplified review of the fixed parsing method, and a variable parsing method with T-code are

978-83-60810-27-9/09/$25.00 © 2010 IEEE

presented for long-term investigation. Then a practical variable length coding will be presented with several variations of construction parameters.

II. Re-Encoding of MPEG-2 coded files

I. Fixed Length Analysis

Before considering V-V coding, we will review F-V coding to evaluate actual entropy. There is redundancy in the MPEG coded bit-stream. There are two problems in evaluating redundancy for coded bit-streams as statistical data with a long interval. These are quantity and quality. How much coded data should be prepared? How many kinds of pictures should be prepared? For this problem, we introduce a new inverted distribution model. Based on the model, we analyze entropy space and show that the distribution is symmetrically uniform on a distorted circle at isentropic space cut out by the same entropy plane. Though the model is uniformly distributed, actual random samples of coded bit-stream statistics are not uniformly distributed as in the case of the model. They are located in a small region by our experimental results. We conclude that the MPEG coded bit-stream is not random, but is very much correlated. Based on these results, we evaluate entropy of coded bit-streams. For the interval of 20bit, the entropy value is 0.9.

If the encoder is an ideal one, the output of the encoder should be a perfect random number, which cannot be re-compressed at all into a file of smaller size. For evaluating ideal encoder performance, random number analysis is one of the most effective methods. Among several methods of random number analysis, entropy evaluation is used.

How much bit data is needed to get reliable results? A relation between the length of the bit stream and the bit pattern interval was obtained as formula (1),

$$n \simeq m \cdot 2^{L} \cdot \log_{e} 2 \ , \qquad (1)$$

where L is an interval length of the bit pattern, 2^{-m} is the level of significance, and n is the required length of the bit stream to be analyzed. Using the formula (1), we can obtain a relation between the length of a bit stream and the interval of a bit pattern to be analyzed with a probabilistic parameter "m". For a coded bit stream with a certain length, using the formula (1), we can get the length of interval "L", which assures accuracy of the result with a risk probability of $\frac{1}{2^m}$.

To consider how many kinds of pictures and coding parameters we should prepare for valid results, we introduced entropy space and cut it with an equi-entropy plane. The entropy per bit is Ent(R) =1.0. The probability distributions give a specific entropy value between 0<Entropy<1.

Fig. 1 shows isentropic curves that are plotted with probability points with the same entropy values. This figure is the case of a computational low dimension. We can utilize this idea for higher dimensional cases because we cannot exactly analyze all cases but some sample cases among all. If we take two points at random on one of these isentropic curves, these two points are not always close at hand, but rather located on symmetrically opposite each other in many cases. The two points that locate in completely symmetrical positions with each other have inverted probability distributions. If we make a sum set of the distribution of these two points with respect to their occurrence frequency, the distribution of the sum set is flat and its entropy should be 1.0. To observe these behaviors, we choose all patterns on the isentropic curves. Then, taking two samples from this pattern set at random, we put them together to form a new point. The new point locates midway between two points in Fig.1, whose entropy increases because the new point moves in an inner direction in entropy space. This method is formalized as an evaluating method 1

```
Method 1:
S1. Evaluate entropy of files, F1,F2,….
S2. Select files, Fs1,Fs2,…,Fsn, with the same entropy
  value E, e.g. E=0.9
S3. Make combined files Cp=U(Fsi,Fsj), i≠j
S4. Evaluate Entropy of Cp
S5. Evaluate increase of Entropy from S1 to S4.
```

If there is no difference between the entropy values of S1 and S4, then the original files are strongly correlated located in a narrow region. If there is a difference, the maximum value indicates the distribution of the original files. If the maximum value is not 1.0, then there should be redundancy in the set of original files.

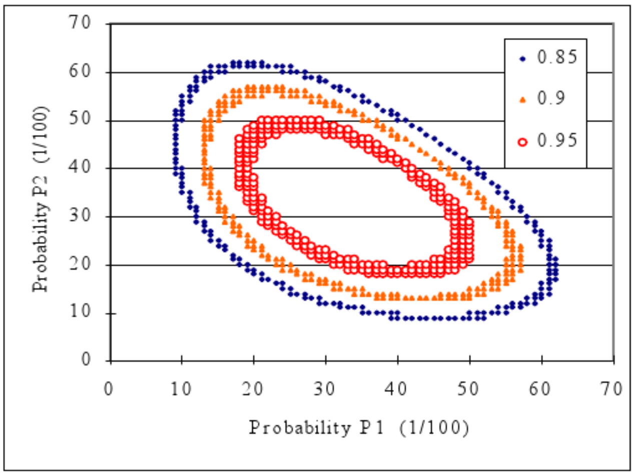


Fig. 1 Isentropic curves for all probabilistic data.

II. Variable Length Analysis

To construct an optimal V-V coding, analysis of input sequences and parsing of the sequence is important. Savari et al presented an analysis of variable-variable length codes [11]. It found that The Tunstall V-F codes can be considered to have an equi-probable code set. Then, combining the Tunstall V-F code with Huffman F-V code will provide higher efficiency. Matsui experimentally proved the concept using text files [5]. In this section, T-code analysis which was carried out in [12] will be introduced.

Fig. 2 shows an example that accomplishes better performance for the V-V encoding scheme than for the F-V encoding scheme. There is a 44-bit sequence in the number column of Fig. 2(d). Parsing this sequence by the fixed length of two bits, we get Fig. 2(a). For three events, allocating one and two bit codes, we get in total 35 coded bits. Next, parsing this sequence by the fixed length of three bits, we get Fig. 2(b). For four events, allocating two bit codes, we get in total 28 coded bits. On the other hand, in the variable coding case, Fig. 2(c) shows variable code example. There are 20 coded bits in total. Fig. 3 shows another four bit fixed code case. The total number of bits is either 21 or 22. Still the variable case has a smaller number of bits.

This example shows that there is at least a better performance variable code than that of all fixed codes within a designated code length.

An example of the T-code generation is described in Fig. 4. The result after parsing is not necessarily the prefix condition. However, the parsed codes represent the original sequence and may provide an influential tool to design a variable parsing code set. According to the T-code generation rule [6], detaching the rightmost bit "1", at first code the second bit "1" from the right. In this case, "1"appears twice and proceeds to two bits to the left. Next, for the "0", code newly "0". Further, the "0" appears three times and counts three for the "0". Then, another "0" appears, which is the fourth time. But in this case, as there is another "1", "10" becomes a newly defined code as an extended new code from the previous generated codes. Further, this "10" code appears four times. The detailed generation rule is described using recursive formulation.

2 bit pattern	freq	code length	bits	code length	bits
0001	4	1	4	2	8
0010	1	3	3	2	2
1000	4	2	8	2	8
1001	2	3	6	2	4
			21		22

Fig. 3 4 bit code case.

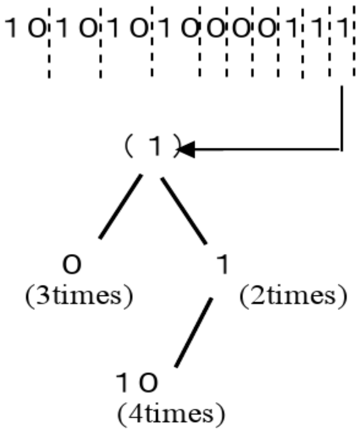


Fig. 4 T-code generation example

2bit pattern	freq	code length	bits
00	9	1	9
01	6	2	14
10	7	2	12
11	0	-	0
			35+

Fig2 (a) 2 bit code case

32bit pattern	freq	code length	bits
000	2	2	4
001	4	2	8
010	4	2	8
011	0	-	
100	4	2	8
101	0	-	
110	0	-	
111	0	-	
			28+

Fig2 (b) 3 bit code case

V-bit pattern	freq	code length	bits
1000	8	1	8
100	3	2	6
10	1	3	3
1	1	3	3
			20

Fig2(c) Variable bit case

No.	2bit	3bit	V
1	1	1	1
2	0	0	0
3	0	0	0
4	0	0	0
5	1	1	1
6	0	0	0
7	0	0	0
8	0	0	0
9	1	1	1
10	0	0	0
11	0	0	0
12	1	1	1
13	0	0	0
14	0	0	0
15	0	0	0
16	1	1	1
17	0	0	0
18	0	0	0
19	0	0	0
20	1	1	1
21	0	0	0
22	0	0	0
23	1	1	1
24	0	0	0
25	1	1	1
26	0	0	0
27	0	0	0
28	0	0	0
29	1	1	1
30	0	0	0
31	0	0	0
32	0	0	0
33	1	1	1
34	0	0	0
35	0	0	0
36	1	1	1
37	0	0	0
38	0	0	0
39	0	0	0
40	1	1	1
41	0	0	0
42	0	0	0
43	0	0	0
44	1	1	1

Fig2 (d) Input Sequence and three code case

III. Experimental Results

A. Fixed Length Coding

The authors analyzed MPEG-2 coded files to re-encode them. The length of the fixed delimiting interval is more than 20 bits. The variety of 20 bit data is about one million. The file length in which these 20 bit patterns appear once is about 2.5 Mbytes. For valid statistical evaluation, at least 10 instances of each pattern should appear on average, which means that an input file needs to be about 25MB. Table I shows the necessary sizes of file and corresponding original video lengths needed for measuring bit length by assuming a ten times occurrence for the files.

Table I
Necessary sizes of file and corresponding original video lengths needed for measuring bit length by assuming ten time occurrence for the files.

bit	Number of patterns (2^{bit})	Size of files	Video lengths（6Mbps)
20	1048576	26MB	35 Sec
30	1073741824	40GB	15 hour
40	1099511627776	55TB	848 days

B. T-code Analysis

Here, to obtain optimal delimiting methods of unknown bit streams in general, for the first step, T-code analysis is carried out. This is only half of the total coding design. But as the first step, we analyze bit streams by T-code and investigate the resultant entropy behavior. Table II shows entropies of generated T-codes for 50 different videos. The values are about half. This implies that the T-code generates codes in a balanced manner. Table III shows increase behaviors when combining two files. Table IV shows further results of combining three files and five files. These values are all normalized to input single bit, and the entropy value of 0.5 means the compressed size is 1/2 of the original size. This entropy is not the so-called T-entropy in [6].

Fig. 5 shows the increasing tendency of T-code entropy when combining a number of files. At the number of five files, the saturation tendency can be seen. The calculation time for T-code analysis takes a long time, and it is limited to evaluation for longer files.

Table II.
Entropies of T-codes for videos.

video	entropy	video	entropy
01.mpg	0.534372	26.mpg	0.524720
02.mpg	0.539007	27.mpg	0.523816
03.mpg	0.529123	28.mpg	0.526155
04.mpg	0.532101	29.mpg	0.539802
05.mpg	0.539262	30.mpg	0.535905
06.mpg	0.538966	31.mpg	0.540882
07.mpg	0.537607	32.mpg	0.536436
08.mpg	0.532928	33.mpg	0.534316
09.mpg	0.529504	34.mpg	0.537676
10.mpg	0.535544	35.mpg	0.534722
11.mpg	0.527797	36.mpg	0.537318
12.mpg	0.524526	37.mpg	0.533381
13.mpg	0.525225	38.mpg	0.536857
14.mpg	0.528234	39.mpg	0.540498
15.mpg	0.533929	40.mpg	0.540714
16.mpg	0.531385	41.mpg	0.539724
17.mpg	0.535190	42.mpg	0.538251
18.mpg	0.527166	43.mpg	0.539320
19.mpg	0.522915	44.mpg	0.531534
20.mpg	0.538911	45.mpg	0.542534
21.mpg	0.530039	46.mpg	0.528326
22.mpg	0.538085	47.mpg	0.526850
23.mpg	0.520653	48.mpg	0.532450
24.mpg	0.537217	49.mpg	0.534462
25.mpg	0.525208	50.mpg	0.533274

Table III.
T-code entropy for combined files.

videos	entropy
01+02.mpg	0.537676
11+12.mpg	0.528685
21+22.mpg	0.537929
31+32.mpg	0.543184
41+42.mpg	0.545325

Table IV
T-code entropy for combined files. Three files and five files case.

videos	entropy
01+02+03.mpg	0.541919
11+12+13.mpg	0.527247
21+22+23.mpg	0.538656
31+32+33.mpg	0.543422
41+42+43.mpg	0.547081
01+02+03+04+05.mpg	0.538796

C. Re-encoding of MPEG-2 Coded Files

Based on the concepts above, the coding efficiency of MPEG-2 coded files is shown as a V-V coding paradigm. Fig 6 shows a V-V coding design and coding execution.

According to coding of motion vectors, Yu et al presented two-dimensional motion vector coding for low bit-rate video phone. However, their Huffman coding was generated by a JPEG procedure and the numbers of motion vector bits were equivalent to about only one-frame for full D1 digital video size, 720×480 [8]. Shimizu et al proposed a method using representation of norm and angle for motion vectors. The method was complicated and still two separate codes were used [9]. Matsuda et al proposed a lossless re-encoding scheme for MPEG-1 video. They used an arithmetic coder to both DCT and motion vector data for the MPEG-1 coder [10].

To realize the whole system, we will propose a re-encoding method that integrates two existing variable codes into a single code as the first step. Based on the information theory, blocking source inputs brings efficiency to the lower bound of entropy. For the Markov source, blocking gains more efficient results. In actual MPEG-2 encoding, 2D-VLC is the only example of blocking source events with run-length of zeros and amplitude in DCT coefficient coding. We constructed block coding for motion vector coding in this paper. Motion vectors of MPEG-2 consist of a set of symmetrical 16 variable-length-codes from 1-bit to 10-bit and one bit code for a zero vector. In the following parts of this section,

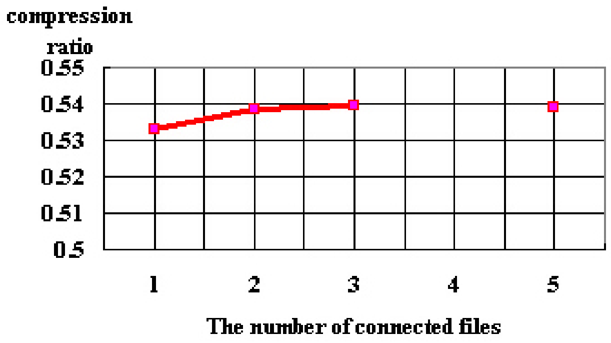


Fig.5 Increasing tendency of T-code's entropy vs. the number of connected files.

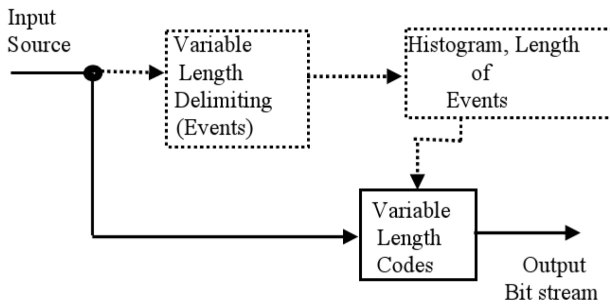

Input Source
Variable Length Delimiting (Events)
Histogram, Length of Events
Variable Length Codes
Output Bit stream

Fig.6 Design of V-V coding and re-encoding.

several experiments are carried out to improve the coding algorithm.

I. Symmetrical coding

In this subsection, 16 different non-zero events and a zero event are coded in a pair of two consecutive original VLCs. There is another sign bit to represent positive or negative for non-zero motion vectors. In this experiment, the sign bit is a fixed one bit, and is excluded for the calculation of efficiency. This scheme is called "Symmetrical coding" because positive and negative evens are coded by the same codes. The motion vectors appear as a pair of horizontal and vertical vectors as shown in Fig.7. For this format, we can block the horizontal vector and the vertical one in a single code.

Table V shows the frequency of motion vectors in encoding a video by MPEG-2 encoder, TM-5. The video size is full D1 (720×480) and the length is half a second.

Table VI shows entropy of coded bits with improving ratios of compression rates. The entropy of blocked motion vectors is reduced 27% at most from the original MPEG-2 coded bits. Table VII shows a part of the constructed codes.

Table V
Frequency of motion vectors of a video. (without sign bit consideration, video 'car' 0.5 sec)

mv	1	2	3	4	5	6
freq	50276	7784	1748	840	501	389
mv	7	8	9	10	11	12
freq	470	623	267	138	187	135
mv	13	14	15	16	17	
freq	253	126	230	257	50	

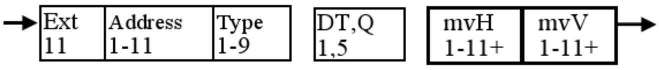


Fig.7 MPEG-2 macroblock layer structure. Ext=Macroblock_extension_code, DT=DCT_type, Q=macroblock quantization step, mvH=motion vector for horizontal direction, mvV=for vertical direction. The numerical values in the bottom sections of boxes are allocated bits for the codes.

Table VI
V-V re-encoding of motion vectors. MPEG-2 TM-5. (without sign bit consideration)

Measuring scheme	Bits	Improvement ratio
Coded bits of mv	1.62 (bit/mv)	1.0
Entropy of mv	1.30 (bit/mv)	0.80
Entropy of 2mv	2.38 (bit/2mv)	0.73
Coded bits of 2mv	2.47 (bit/2mv)	0.77
Coded bits of 2mv per mv	1.24 (bit/1mv)	0.77

Table VII
V-V codes for a pair of motion codes. (part) (without sign bit consideration)

	Code		Bit
Index	Value :	Bit pattern	
1	1 :	1	1
2	15 :	01111	5
3	33 :	0100001	7
4	79 :	01001111	8
5	436 :	0110110100	10
6	291 :	100100011	10
7	851 :	1101010011	11
8	616 :	1001101000	11
9	2743 :	0101010110111	13
10	3539 :	0110111010011	13
11	2688 :	0101010000000	13
12	2742 :	0101010110110	13
13	5378 :	01010100000010	14
14	2905 :	0101101011001	13
15	10759:	010101000000111	15
16	10758:	010101000000110	15
17	0	00	2
18	29	011101	6
19	219 :	011011011	9

The Huffman codes are generated using free software by Marcus Geelnard available at http://bcl.comli.eu/. Table VIII shows a part of the constructed codes.

II. Influence of Video Length

In this subsection, we examine the necessary length of input video for these experiments. It is better to use video test sequences as far as possible to examine the performance pre-

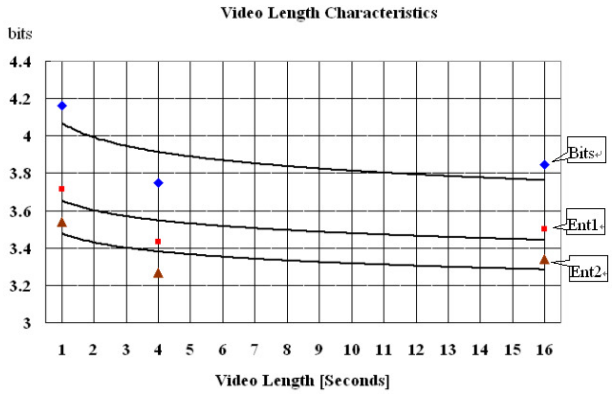


Fig.8 Video length influence of MPEG-2 re-encoding for the video No. 4, at 8Mbps.

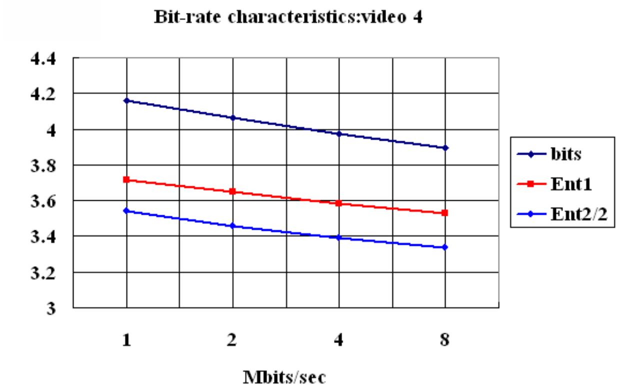


Fig. 9 Bit rate characteristics for the video No.4 with a duration of four seconds.

cisely. On the other hand, it is better to keep time and data volume to a minimum. New results of re-encoding for several cut-out partial sequences from a single video are listed in Table 8. Fig. 8 is a graph showing re-encoding. From the start of 0.5 seconds, information decreases gradually toward 16 seconds. The least squares regression lines on the graph can be gradually decreased and nearly converge. The videos used in this paper are listed in Table 9. The video used to examine of the influence of video length is No.4 "autobahn". The size of the motion picture is 720x480.

The bit-rate of encoding is 8Mbps. Sixteen different non-zero events and a zero event including the sign bit to represent positive or negative for non-zero motion vectors (MV) are coded in a pair of two consecutive original VLCs. In this experiment, the sign bit is included in newly generated VLCs and for the calculation of efficiency.

Viewing these behaviors, we choose the video length of 4 seconds for the following experiments.

III. Influence of bit-rate

In this subsection, the influence of bit-rate is examined. There are many choices of bit-rates in MPEG-2 encoding. It is important to check the effects of bit-rate to re-encoding code design. Fig. 9 shows bit-rate characteristics with log-scale for the horizontal axis. Bits mean the original MPEG-2 motion vectors. Ent1 means entropy of motion vectors of MPEG-2. Ent2/2 means entropy of motion vectors obtained by the proposed re-encoding method. In general, a decrease can be seen with bit-rates. But improving ratios from MPEG-2 to re-encoding may be the same. The entropy is larger for low bit-rates, which means a role of motion vectors is large and may require more bit-rate for describing videos. For high bit-rates, the smaller entropy means that there may be redundancy and all motion vectors are not necessarily required. These understandings coincide with the former comments that improvement of motion vector coding is effective for low bit-rates in references [8] and [9].

IV. Evaluation of a variety of videos

Table VIII and Fig 10 show overall comparison of re-encoding efficiency. The first column of Table VIII is the numbers of coded bits of motion vectors for the case of the original MPEG-2. Ent1 means entropy of motion vectors of MPEG-2. Ent2/2 means entropy of motion vectors in the case of the re-encoding method. In Fig.10, improvement ratios from MPEG-2 to re-encoding are large for videos No. 1, No.4 and No.5, but are small for videos No.2 and No.3. The characteristic of videos No.2 and No.3 is relatively smaller motion. On the other hand, videos No.1, No.4 and No.5 have large motion scenes. Ave in Fig. 10 means the average of five results. Table IX are videos used in these experiments.

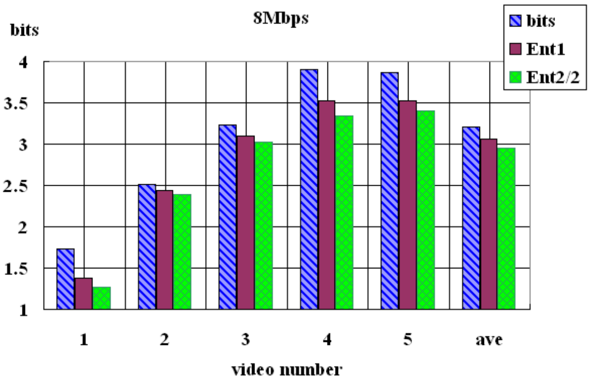


Fig. 10 Overall comparison of re-encoding efficiency.Video length is four seconds for No. 1-5. Bit-rate is 8Mbps.

V. Comparison of quantity of the number of MVs

Table X shows increased bits of motion vectors used in our experiments in this paper. A large number of video data are used to analyze the methods in detail and to improve reliability of the experiments. About 30 times more than the conventional experiments carried out by Yu et al. [8].

Fig.11 (a)-(d) are sample pictures of videos used in these experiments except (e) which is a rugby game on television. The former four videos are presented at author's homepage [13].

IV. Conclusion

A new re-encoding paradigm is reviewed and a two dimensional semi-optimization is examined. A large number of

Table VIII
Video length influence of MPEG-2 re-encoding. Video is No.4, 8Mbps.

Length of video [Sec]	bits	Ent1	Ent2/2
0.5	3.95	3.57	3.37
1.0	3.90	3.53	3.34
4.0	3.75	3.43	3.27
16.0	3.84	3.50	3.34

Table IX
Video sequences used in this paper.

No.	name	content
		original size
1	car	a taxi left to right
		SD:720x480
2	giraffe	jiggle by hand movement
		SD: 720x480
3	cherry	swinging cherry blossom in wind
		HD: 1920x1080
4	autobahn	highway driving
		HD:1920x1080
5	rugby(75)	rugby game in television
		SD: 720x480

Table X
Increased bits of motion vectors in this paper compared to the conventional paper [8].

Yu's experiments [8]		Our experiments	
video sequences	Bits of motion vectors	video sequences	Bits of motion vectors
Miss Am	6987	1.car	110464
Mother & Daughter	9135	2.giraffe	397424
Salesman	5377	3.cherry	412304
Car Phone	14961	4.autobahn	246924
Foreman	22105	5.rugby	615714
total	58565	total	1782830

video data are used to analyze the methods in detail and to improve reliability of the experiments. By two-dimensional Huffman re-encoding by V-V codes, 5-26% coding efficiency is obtained. Though the result is restricted to the motion vector parts, the efficieny improves much comparing to the conventional methods using F-V codes for MPEG-2 with 5% coding efficiency.The results may be stable as to length of video and bit-rates. However, for the variety of video contents, the result may still not be convergent. This implies that a larger variety of videos, including still scenes and large motion ones should be tried in the future.

Fig. 11(a) Video 1 car

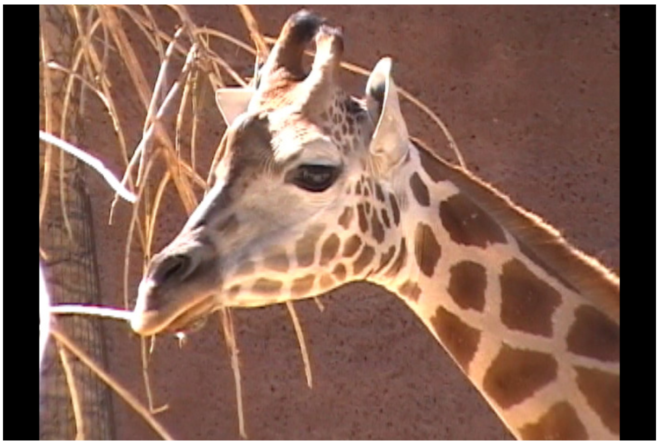


Fig. 11(b) Video 2 giraffe

Fig. 11(c) Video 3 cherry

Acknowledgment

Kazuo Ohzeki, the author, thanks Dr. Mark Titchener who provided tcalc software to calculate T-code in this paper. He also thanks Mr. T. Kato who carried out experiments of T-code analysis.

References

[1] Jacob Ziv. Variable-to-fixed length codes are better than fixed-to-variable length codes for Markov sources. IEEE Trans. IT, 36(4): pp861-863, July 1990.

Fig. 11(d) Video 4 autobahn

[2] Te Sun Han, Kingo Kobayashi , 2001. *Mathematics of Information and Coding*. American Mathematical Society, Boston, MA, USA. **Book**.

[3] H. Yamamoto and H.Yokoo, “Average-Sense Optimality and Competitive Optimality for Almost Instantaneous VF Codes”, IEEE Trans. IT, 47(6): pp.2174-2184, Sept. 2001.

[4] J. Abrahams, "Code and parse trees for lossless source encoding ", Proceedings of Compression and Complexity of Sequences pp.145 - 171 , Jun. 1997.

[5] Yudai Matsui and Takuya Kida, "Study on Efficiency of Tunstall-Huffman Code", Proc. of Data Engineering and Information Management 2009 Demi Forum i1-28 (in Japanese) http://db-event.jpn.org/deim2009/proceedings/files/i1-28.pdf

[6] Ulrich Gunther et al., “Representing Variable-Length Codes in Fixed-Length T-Depletion Format in Encodes and Decoders”, J. of Universal Computer Science Vol. 3 No. 11 pp.1207-1225, Springer. 1997.

[7] Mark Titchener, "Digital encoding by means of new T-codes to provide improved data synchronization and message integrity", Technical Note, IEE Proceedings, Volume: 131, Pt. E, Number: 4 , July 1984, Page(s): 51 –53.

[8] Guo Yao Yu, and Cheng-Tie Chen, "Two-dimensional motion vector coding for low bit rate videophone applications", Proc. ICIP 1995, vol. 2, pp. 2414.

[9] Astushi SHIMIZU, Astushi SAGATA, Kazuto KAMIKURA,Naoki KOBAYASHI, " Motion Vector Coding by Using Representation of Norm and Angle Components", IEICE Trans. J84-D-II(11), 2379-2386, 2001. (in Japanese).

[10] Ichiro Matsuda, Kei Wakabayashi, Yu Ikeda and Susumu Itoh "A Lossless Re-encoding Scheme for MPEG-1 Video",Proceedings of 17th European Signal Processing Conference (EUSIPCO-2009), pp.1834-1838.

[11] S.A, Savari, W. Szpankowski, "On the analysis of variable-to-variable length codes ", Proceedings. 2002 IEEE International Symposium on Information Theory,page176, 2002.

[12] Kazuo OHZEKI, Tsuyoshi KATO, and Engyoku GI, “Basic consideration for lossless re-encoding of MPEG coded files using V-V codes”, IEICE Technical Report, IE2010-6, pp.31-36, April, 2010. (in Japanese).

[13] http://www.sic.shibaura-it.ac.jp/~ohzeki/oz4c/mmap/videos/index.html

ISBN 978-83-60810-27-9
ISSN 1896-7094

Analyzes of the processing performances of a Multimedia Database Server

Cosmin Stoica Spahiu
University of Craiova,
Craiova, Dolj, Romania
Email: stoica.cosmin@software.ucv.ro

Abstract—**The paper presents an original dedicated integrated software system for managing and querying alphanumerical information and images from medical domain. The software has a modularized architecture controlled by a multimedia relational database management server. The server is designed to manage database creation, updating and complex querying based on several criteria: simple text-based or content-based image query on color or texture feature, extracted from color and gray-scale image.**

Keyword—**smultimedia; database server; content based retrieval; insert operations.**

I. Introduction

ONE of the domains where a large quantity of alphanumerical and visual information is acquired daily is the medical one. This information is obtained during the patients' diagnosis and treatment process. Some of the imagistic medical data sources are:

- Electronic medical sheets: these files contain information about patients' name, birth date, medical antecedents, signs, main diagnosis, secondary diagnosis, values of the analysis and treatment.
- Medical images that are stored in digital format or images stocked on different media (X-ray film, paper, etc).
- Digital Imaging and Communications in Medicine (DICOM) files – these standard files are produced by the most part of medical devices (echographs, endoscopes, magnetic resonance imaging devices) which are used in patients diagnosis. A DICOM file contains alphanumerical information (patient name, doctor name, consulting date, diagnosis) and one or several images stored in different formats, compressed or uncompressed.

That is why the problem of storing the medical images collections in digital format along with the associated information (patient name, diagnosis, consulting date and treatment), managing the database and executing efficient queries, it is intensely studied in order to find new and more efficient solutions.

There are only few systems on the market that have already integrated algorithms for image processing and features extraction into the medical diagnosis process. Most of the applications use classic Database Management Systems, like: Microsoft SQL Server, MySQL or Interbase.

The problem is that almost none of these servers offer support for multimedia data. The users have to implement their own algorithms and methods for images processing.

It is presented in this paper an original solution that integrates both the methods needed to process the images and methods for executeing complex queries, based on the content.

The paper has the following structure: Section 2 presents a short overview of the images format, Section 3 describes the algorithms used for characteristics extraction, in Section 4 it is described the system and made an analysis of the images processing performances and Section 5 presents the conclusions.

II. An original Implementation of a Multimedia Database Server

In order to manage content based retrieval for images collections there have been implemented a series of applications. Most of them are using classic Database Management Systems, like: Microsoft SQL Server, MySQL or Interbase.

The main drawback of these systems is that, most of them offer no support for multimedia data (neither for processing, nor for searching).

In [5][6] it is presented an original solution for managing visual information from images. This solution implemented in Visual C++ is a multimedia database management system (MMDBMS) that includes algorithms for extracting texture and color characteristics and for executing content-based visual queries.

This server is designed to manage medium sized personal digital collections having at most few tens of thousands of records. Over this number of records it should be redesigned the indexes system in order to enhance the execution time.

The implementation includes both a server and a client application that can be executed from any internet browser.

An element of novelty for this implementation is that the client application has the possibility to build visual content-based queries directly from the interface.

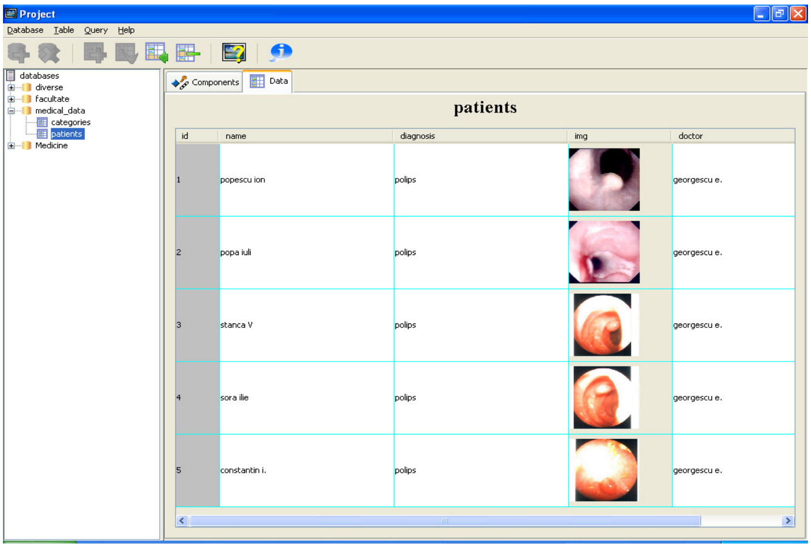
The elements of this window which permit content based retrieval are (figure 2):

– Similar With – opens the window for choosing the query image

– Select – permits to choose the field (or fields) that will be presented in the results of the query

– From – it represents the tables in database, that will be used for the query

The support of the The National University Research Council under Grant CNCSIS IDEI 535 is gratefully acknowledged.

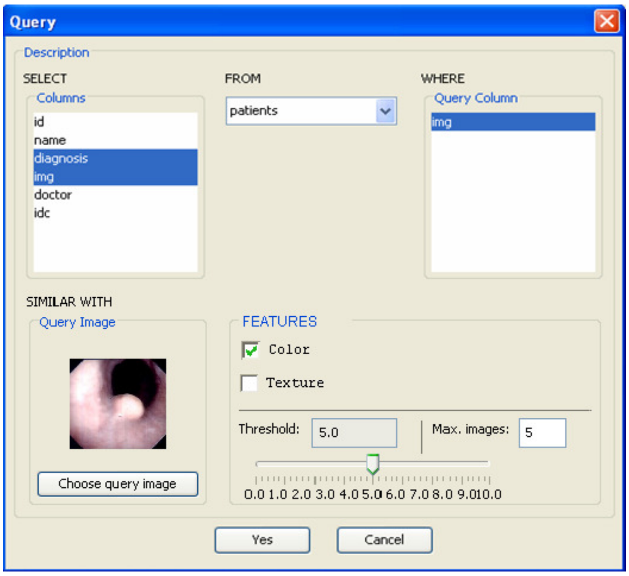


patients

id	name	diagnosis	img	doctor
1	popescu ion	polips		georgescu e.
2	popa iuli	polips		georgescu e.
3	stanca V	polips		georgescu e.
4	sora ilie	polips		georgescu e.
5	constantin i.	polips		georgescu e.

Fig. 1. Main page of the client application

- Where – the image type column used for content-based image query
- Features – it is chosen the characteristic used for content based visual query – color, texture or a combination of them
- Threshold – it is chosen a threshold of accepted similitude between query image and target image. An image with a similitude under that threshold will not be added into the resulted query images



SELECT Columns: id, name, diagnosis, img, doctor, idc

FROM: patients

WHERE Query Column: img

☑ Color

☐ Texture

Threshold: 5.0

Max. images: 5

Fig. 2. The window implementing content-based queries

- Maximum images – specify the maximum number of images returned by the query

Based on these options, it is generated a modified SQL command for content-based retrieval that is sent to the server along with the query image.

The client communicates with the server through messages exchange using sockets. The client sends SQL commands and receives the results. The results can be either strings (responses to queries or error messages), or multimedia files in which case it is needed to return images to client.

The execution time for content based queries is influenced by two factors: the time needed to process the query image and the time needed to search similar images into the database.

Each image is processed before being stored into the database and extracted color and texture characteristics. That is why it is needed in this step to process only the query image. Depending to the size of the image, this can take an important time.

Only after this step is finished the server can apply specific algorithms in order to find the most similar images.

An original aspect of the server is that it includes all the methods needed to extract the characteristics and compare the similitude of the images.

III. Characteristics Representation and Extraction

The server is intended to be used especially with medical images. The studies have shown that not all the methods used to extract characteristics give the best results in any circumstances [10][11]. That is why it is important to know the domain where the server will be mainly used.

In our experiments there were considered images from gastric tract. For these images the best results were obtained using histograms for color characteristics and Gabor filters for texture characteristics [10][11].

A. Color characteristics representation

Image processing is a technique used to increase the images quality in order to be easily understood by the human eye, or to help extract some important characteristics. In order to succeed it is necessary to identify pixels groups that are interconnected by common characteristics. Some useful information can be extracted from colors repartition in pixels, obtaining the color histogram. For that it is measured and represented the number of pixels for each color. The histogram does not say anything about the geometric relations between pixels. The image-histogram relation is not mutual, meaning that there can be several images with the same histogram [3].

If the measured characteristics can have k_1 values for the first coordinate, k_2 values for the second…, and k_n for the last, then the resulted histogram will have the size:

$$size=\prod_{i=1}^{n} k_n \tag{4}$$

A major disadvantage of the histograms is that they need a lot of space, especially if the number of colors is high. A good solution would be to store only characteristics that have a non-zero value, or the value is above a specified threshold.

The histograms are invariant to translations, rotations, and they have only small variations when the viewing angle is changed.

The following definition can be given: a histogram represents the colors distribution in an image, region or object. It is calculated using the following formula:

$$h_c[m]=\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1}\left\{\begin{array}{ll}1, \text{ if } :Q_c(T_c I[x,y])=m \\ 0, & otherwise\end{array}\right\} \tag{5}$$

Another disadvantage of the histograms is they do not specify the geometric relations between pixels. That means several images might have the same histogram, although they are totally different.

B. Texture characteristics representation

The texture is the second important characteristic of the images that can be used in content-based retrieval. It is hard to give an exact definition of texture. Usually the word “texture” is used to define the touch feeling of objects, without touching them.

The dictionary definition is: the internal organization of an object, the association modality of an object, characterized by its shape, size of each element and by the geometric relations between each component.

A texture is created by regularly repeating an element or a model on a surface. This model/element is called texture point (textel). The computer graphic techniques define two types of textures: deterministic (regular) and statistic (random).

The deterministic textures are created by repeating the same specified geometric form (e.g.: circle or rectangle). An example of such a deterministic texture is the bricks in the wall.

The statistical textures are obtained by modifying the patterns using specified statistic properties (e.g.: the texture of wood or rocks). They are typically specified by properties of the spatial frequencies.

In order to extract the texture characteristics there have been studied a series of methods. The most representative are Gabor filters and Coocurence matrix. Although there are a lot of other available techniques, there is no one to be considered the best. This depends especially by the type of images used (from nature, medical, etc.).

C. Extracting texture characteristics using Gabor filters

Starting from HSV color space representation, the color can be represented in complex. Any point from the HSV cone can be computed using the formula [4]:

$$z_M = S\,(\cos H + i \sin H) \tag{6}$$

The saturation can be interpreted as size, the hue as the phase of the complex value. There are a lot of advantages of such a representation. First, it is simple due to the fact that color is a scalar and not a vector and the combination between channels is made before filtering. In conclusion, the color can be represented in complex using:

$$b(x,y)=S(x,y)\times e^{iH(x,y)} \tag{7}$$

To compute the Gabor characteristics for an image represented in HS-complex it is used a method similar as computing the monochromatic Gabor characteristics. This is due to the fact that the color channels are combined before filtering [4]:

$$C_{f\phi}=\left(\sum\left(FFT^{-1}\left\{P(u,v)\times M_{f,\phi}(u,v)\right\}\right)\right)^2 \tag{8}$$

The Gabor characteristics are created using the value $C_{f,\varphi}$ computed for 3 scales and 4 directions:

$$f = (C_{0,0}, C_{0,1},\ldots\ldots, C_{2,3}) \tag{9}$$

The similitude between these characteristics is defined by the metric:

$$D^2(Q,T)=\sum\sum d_{f,\phi}(Q,T)\text{,whered}_{f,\phi}=(f^Q-f^T)^2 \tag{10}$$

IV. Experiments and Results

Because it is a client-server application, the execution time depends on two aspects: the network speed that is used to send data to the network and the data processing speed.

Table 1.
Time needed to transfer images

N o	Image resolution	Image size	Time needed
1	160 ×160	75 KB	0,30 s
2	240 × 240	172 KB	0,75 s
3	480 × 480	675 KB	2,90 s
4	640 × 640	1204 KB	5,20 s
5	1200 × 1200	4220 KB	18,30 s

Usually the information sent to server is based on strings, less than 100KB in size. Nowadays the network speed is high enough that the sending time can be considered close to zero.

In our case, an important role in data transfer is represented by the images data. Depending to the resolution and compression used, they might have a size up to several MB. In this case it is very important the speed used to send data to the network.

Taking into account that the size of a BMP image with resolution of 100×100 pixels, is 30 KB it can be deducted the time needed to send images to the server:

The time effectively needed by the server for an insert operation is higher than presented above because the server uses a handshaking communication technique for client communication and data sending: the client sends to the server the insert command, the server responds with an OK confirmation. It analysis the data next and if needs more data (such as receiving an image), sends to the client a data request. The client responds sending the image. To the last step, the server acknowledges receiving the image. If one of them does not receive an acknowledgement it will resend the information.

After all the information is receive, before making the insertion in the database, the server has to process the image received and extract the color and texture information.

In order to extract the characteristics, the image colors have to be translated from RGB to HSV color space, quantized to 166 colors and after that applied the algorithms presented above to extract histogram and texture. These operations are made to the pixel level and need a high processing power and important execution time.

In the tested version, the server process only BMP images. If the image would be compressed (e.g.: JPEG), it should be decompressed first and only after that processed as the BMP images.

In Table 2 it is presented the time needed to extract the color and texture characteristics from images having different resolutions.

The graphic resulted from the above table is presented in figure 2.

It can be noticed a linear growth of the execution time with the image size, due to the linearity of the algorithms used.

Table 2.
Time needed to process the images

No	Image resolution	Time for color characteristics	Time for texture characteristics	Total time
1	160 ×160	0,45 s	14 s	14,45 s
2	240 × 240	0,60 s	14,5 s	15,10 s
3	320 × 320	0,70 s	15,30 s	16 s
4	480 × 480	1 s	63,70 s	64,70 s
5	640 × 640	1,50 s	108,50 s	110,50 s

For an image having the 160 × 160 pixels resolution it is necessary approximate 15 seconds to extract the characteristics. For an image of 640 × 640, the time will increase to almost 2 minutes.

Taking into account that there are cases when several users want to execute insert operations simultaneously, it was considered to be useful to limit the images resolution to 500×500 pixels. This will give the possibility to finalize the insert operations in a reasonable time.

V. Conclusions

The paper presented a new solution for managing and querying multimedia images collections. The implemented multimedia relational database management system includes an original data type, called IMAGE used to store all the characteristics extracted from images. These characteristics are used in content based retrieval.

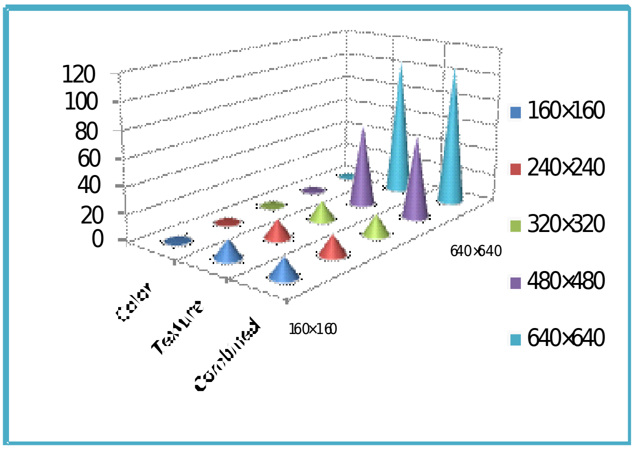


Fig. 3. Time vs. Image resolution

It is created for managing and querying medium sized personal digital collections that contain both alphanumerical information and digital images (for examples the ones used in private medical consulting rooms). The software tool allows creating and deleting databases, creating and deleting tables in databases, updating data in tables and querying. The user can use several types of data as integer, char, double or image.

The quality of the server is tested from the execution point of view. It is tested the time needed to process the insert queries, namely the time needed to extract color and texture characteristics.

This software can be extended in the following directions:

- Adding new types of traditional and multimedia data types (for example video type or DICOM type - because the main area where this multimedia DBMS is used it is the medical domain and the DICOM type of data is used for storing alphanumerical information and images existing in a standard DICOM file provided by a medical device)
- Studying and implementing indexing algorithms for data inserted in the tables in order to enhance the execution time.

References

[1] D. Cardani, *"Adventures in HSV Space"*

[2] D. Cojocaru, *"Images Acquisition, processing and recognition"*. Universitaria Craiova, 2005

[3] C. Palm, D. Keysers, T. Lehmann, K. Spitzer, "Gabor Filtering of Complex Hue/Saturation Images For Color Texture Classification". In *5th Joint Conference on Onformation Science (JCIS2000)*, pp. 45-49, 2000

[4] C. Stoica Spahiu, C. Mihaescu, L. Stanescu, D.D. Burdescu, M. Brezovan, "Database Kernel for Image Retrieval". In *The First International Conference on Advances in Multimedia (MMEDIA 2009)*, Colmar – France, pp. 169-173, 2009

[5] C. Stoica Spahiu, L. Stanescu, D.D. Burdescu, M. Brezovan, "File Storage for a Multimedia Database Server for Image Retrieval". In *The Fourth International Multi-Conference on Computing in the Global Information Technology (ICCGI 2009)*, Cannes/ La Bocca, France, pp.35-40 (2009)

[6] M. Kratochvil, *"The Move to Store Images In the Database",* 2005 http://www.oracle.com/technology/products/intermedia/pdf/why_images_in_database.pdf

[7] G. Lu, *"Multimedia Database Management Systems".* Artech House Publishers, 1999

[8] H. Muller, A. Rosset, A.Garcia, J.P.Vallee, A. Geissbuhler, *"Benefits of Content-based Visual Data Access in Radiology".* Radio Graphics. 25, pp. 849-858, 2005

[9] L. Stanescu, D.D. Burdescu, M. Brezovan, *"Multimedia Medical Databases Chapter Book".* In: Sidhu, Amandeep S.; Dillon, Tharam; Bellgard, Matthew (Editors.), Biomedical Data and Applications, Series: Studies in Computational Intelligence, Vol. 224, Springer Verlag, 2009

[10] L. Stanescu, D. D. Burdescu, C. Stoica Spahiu, M.Brezovan, "A study of two color systems used in content-based image query on medical imagery". In *International Conference on Informatics in Control, Automation and Robotics* (ICINCO 2007). pp. 337-340, 2007

[11] T. Gevers, A. Smeulders, Color-based object recognition. *Pattern Recognition*, Vol. 32, pp. 453-464, 1999.

[12] HSV Color Space, http://en.wikipedia.org/wiki/HSV_color_space.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Constructive Volumetric Modeling

Mihai Tudorache
Faculty of Automatics, Computers
and Electronics
Craiova, Romania 200440
Email: mtudorache@software.ucv.ro

Mihai Popescu
Faculty of Automatics, Computers
and Electronics
Craiova, Romania 200440
Email: mpopescu@software.ucv.ro

Razvan Tanasie
Faculty of Automatics, Computers
and Electronics
Craiova, Romania 200440
Email: tanasie_razvan@software.ucv.ro

***Abstract*—In this article we intend to present a method of obtaining high complexity sinthetic scenes by using simple volumes as the building blocks. The below described method can be used to obtain both homogenous and heterogenous volumes. This is done by combining volumes of different voxel densities.**

***Index Terms*—volumetric data, voxel, constructive solid geometry, volume modelling, constructive volume geometry.**

I. Introduction

THE VOLUMETRIC data imaging technology has greatly improved sice the 1990's. Before this many fields had to work with data images that used the depth or field effect known from 2D screens.

Because of the improved graphical representation, the volumetric data has found extensive use in medical applications such as 3D ultrasound, CAT (Computed Axial Tomography) or MRI (Magnetic Resonance Imaging). Other fields putting the technology to use include geological surveying, security scanning and, potentially, 3D gaming.

Given the importance of volumetric data, a lot of research has been done lately in the field, ranging from volumetric generation and rendering to volumetric segmentation, indexing and compression. This research was mainly done using data that resulted from real medical cases. Only recently did researchers start to use data obtained by scanning physical objects using lasers. Even so, volumetric data is stil scarce and not readilly available.

Because volumetric data is in general obtained through medical imaging devices it is usually hard to come by. Given the ease with wich sinthetic scenes can be manufactured, the sinthetic senes are somtimes prefered for volumetric analysis. It must be said that the sinthetic scenes offer less diversity than the real medical data but they can be custom made to the precise needs of the desired field of analysis. Volume segmentation and indexing can greatly take advantage of sinthetic scence tailormade for its needs.

II. Constructive volume modeling

The constructive volume generation technology is not a new thing and many articles have been written on this topic. For example the Constructive Volume Geometry (CVG) [1] article presents an algebraic framework for modelling complex spatial objects using combinational operations. In this article we present another approach for volume generation. We will use basic volumes like spheres, prisms, cylinders, cones, tori, etc. as building blocks for more complexe volumes. These volumes are combined using boolean operators like union and intersection. The resulting volumes can be combined with other volumes to form more complex objects.

The volumes that we are using are made up of voxels that have a position in the volumetric spece they are defined in and a density, with values in the interval [0,1]. The density can later on be interpreted as a color in a given spectrum. For the examples given in this article we chose a palette consisting of shades of green ranging from light green for density 0 to dark green for density 1.

A volume is stored in a 3D matrix of densities, where a voxel is represented by its position in the matrix and the density stored at that position.

contenta-images2eps.com

Fig. 1. Sphere and cone union.

In order to combine their volumetric data, all volumes must be defined in the same subspace.For example we consider a 256x256x256 cube made up of voxels. Each voxel in the cube has a density between 0 and 1. Let us take two such cubes, A and B, which contain the volumetric data for a homogenous sphere of radius 100, centered in (100, 100, 100) and a homogenous cone of radius 100 and height 50, centered in (100, 200, 100) respectively. Both volumes have the density equal to 1. The union of A and B will be another 256x256x256 cube of voxels as presented in Fig. 1.

When we make a union between two volumes, we actually add the densities of corresponding voxels in the cubes containing these volumes, caping the densities at a maximum of 1, thus obtaining a valid new cube, with all densities between 0 and 1. The resulting cube can be used in subsequent operations.

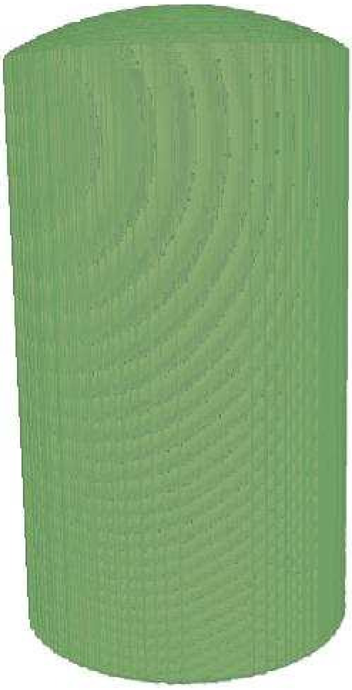


Fig. 2. Sphere and cylinder intersection.

For the intersection of two volumes we compute the product of all corresponding densities in the cubes containing the volumes. Given that the densities of voxels have values in the interval $[0, 1]$, after multiplication the resulting densities also have values in the $[0, 1]$ interval. An example of an intersection between a sphere and a cylinder is given in Fig. 2.

The difference of two volumes is obtained by subtracting the corresponding densities of voxels in the cubes containing the volumes. Because the resulting densities can fall below 0, we need to limit these values at 0. An exemple of a difference is given in Fig. 3.

The complement of a volume can be obtained by subtracting from 1 the voxel densities of the cube containing the volume. The resulting cube has all densities in the $[0, 1]$ interval.

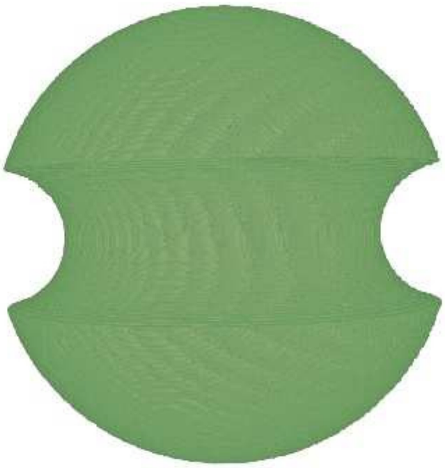


Fig. 3. Sphere and torus difference.

These operations have the potential of creating realy complex volumes. In Fig. 4. there is a volume created from a sphere and three cylinders.

The basic volumes used in these operations are created using their parametrized equations. For example, a sphere is defined by the equation (1).

$$\begin{aligned} x&=x_0 + r\sin\theta\cos\phi \\ y&=y_0 + r\sin\theta\sin\phi \quad (0 \leq \phi \leq 2\pi \; and \; 0 \leq \theta \leq \pi) \\ z&=z_0 + r\cos\theta \end{aligned} \qquad (1)$$

By taking discrete values from the intervals $[0, 2\pi]$,$[0, \pi]$, and $[0, r]$ we are able to build our sphere voxel by voxel. The

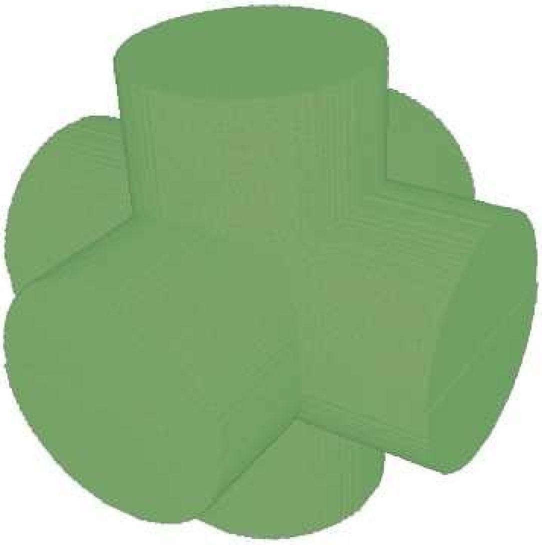


Fig. 4. Union of three cylinders.

algorithm for constructing a homogenous sphere is given in Algorithm 1.

Algorithm 1 Generating voxels for a homogenous sphere of density 1.

```
Require: $r > 0$
  $r \leftarrow 1$;
  while $r <= radius$ do
    $theta \leftarrow 0$;
    while $theta \leq 2\pi$ do
      $phi \leftarrow 0$;
      while $phi \leq \pi$ do
        $x \leftarrow x_0 + r * \sin(theta) * \cos(phi)$;
        $y \leftarrow y_0 + r * \sin(theta) * \sin(phi)$;
        $z \leftarrow z_0 + r * \cos(theta)$;
        $volume_{x,y,z} \leftarrow 1$;
        $phi \leftarrow phi + \arcsin(1/r)$;
      end while
      $theta \leftarrow theta + \arcsin(1/r)$;
    end while
    $r \leftarrow r + 1$;
  end while
```

In order to obtain a heterogenous volume we can combine homogenous volumes using the operations defined previously. For example a sphere with three layers of density d_1, d_2 and d_3 can be obtained by making a union between a homogenous sphere with density d_1, a homogenous sphere shell with density d_2 and another homogenous sphere shell with density d_3. A section through the resulting sphere can be seen in Fig. 5.

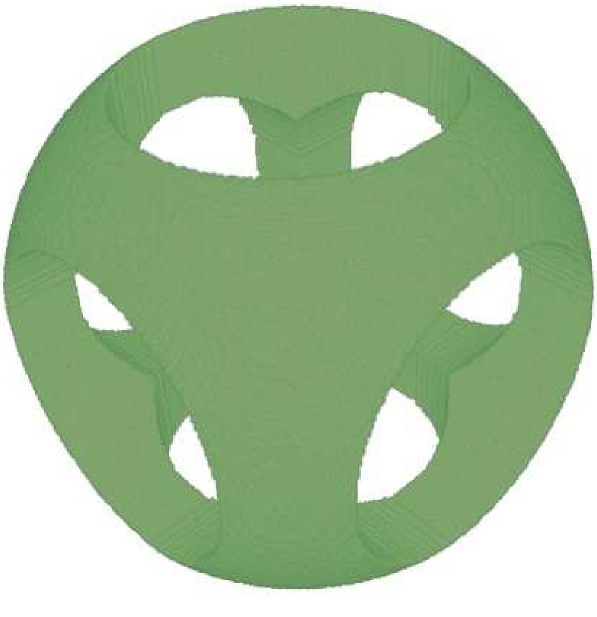


Fig. 5. Sphere and cylinders difference.

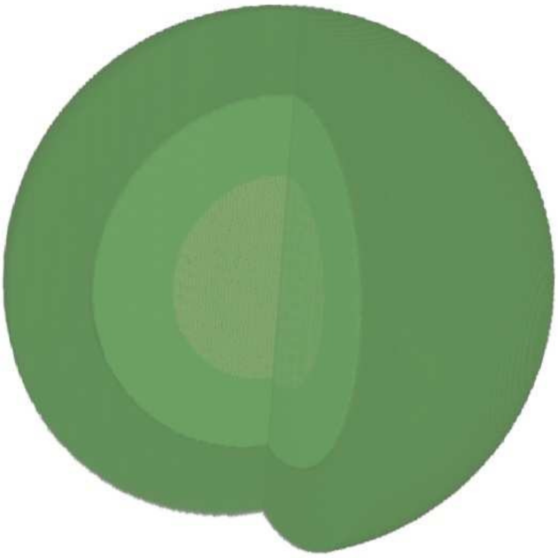


Fig. 6. Heterogenous sphere obtained by combining three basic homegenous volumes through union. The section through the sphere was made by means of a difference with a prism.

III. Conclusion

We have shown that more complexe volumetric objects can be obtained by combining basic volumes using boolean operators. This is very helpful in obtaining synthetic data for other volume related fields of research like volumetric segmentation and indexing, and volume compression.

The simple sinthetic data can be combined into more complex forms thus giving a large colection of objects from where to choose when performing volumetric analysis.

The method that we have used is a very simple but an ingenious one as presented above. We will use this method in our future work.

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] M. Chen and J. V. Tucker, *Constructive Volumetric Geometry.* Computer Graphics forum, United Kingdom, 2000.

[2] A. A. G. Rwquicha, *Representations for rigid solids: theory, methdos and systems*, Computing Surveys, 12(4), pp. 437-464 1980.

[3] S. E. Follin, *Scientific visualization*, in A. Kent and J.G. Williams (eds), Encyclopedia of Microcomputers, 15, pp. 147-178, Marcel Dekker, New York, 1998.

[4] J.C. Torres and B. Clares, *A formal approach to the specification of graphic object functions*, Computer Graphics Forum, 13(3) pp. C371-C380, 1994.

[5] K. Meinke and J. V. Tucker, *Universal algebra*, Handbook of Logic in Computer Science, Volume I, pp. 189-411, Oxford University Press, 1992.

[6] M. Levoy, *Efficient ray tracing of volume data*, ACM Transactions on Graphics, 9(3), pp. 245-261, 1990.

[7] D. Laur and P. Hanrahan, *Hierarchical splatting: a progressive refinement algorithm for volume rendering*, ACM/SIGGRAPH Computer Graphics, 25(4), pp. 285–288, 1991.

[8] A. Leu and M. Chen, *Modelling and rendering graphics scenes composed of multiple volumetric datasets*, Computer Graphics Forum, 18(2), pp. 159–171, 1999.

[9] S. Wang and A. Kaugman, *Volume sculpting*, in Proceedings of Symposium on Interactive 3D Graphics, pp. 151-156, 1995.

[10] T. A. Galyean and J. F. Hughes, *Sculpting: an interactive volumetric modeling technique*, ACM/SIGGRAPH Computer Graphics, 25(4), pp. 267–274, 1991.

[11] W. Lorensen and H. Cline, *Marching cubes: a high resolution 3D surface construction algorithm*, ACM/SIGGRAPH Computer Graphics, 21(4), pp. 163–169, 1987.

[12] S. D. Roth, *Ray casting for modelling solids*, Computer Graphics and Image Processing, 18, pp. 109-144, 1982.

[13] G. Nielson, *Volume modelling*, in M. Chen, A. Kaufman and R. Yagel(eds), Volume Graphics, pp. 29–48, Springer, London, 2000.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 759–766

ISBN 978-83-60810-27-9
ISSN 1896-7094

Real-Time Embedded Fault Detection Estimators in a Satellite's Reaction Wheels

Nicolae Tudoroiu
Concordia University
1455 De Maisonneuve Blvd. West,
Montreal, Quebec, Canada
Email: tnicolae@excite.com

Ehsan Sobhani-Tehrani
McGill University
805 rue Sherbrooke Ouest
Montreal, Quebec, Canada
Email:ehsan.sobhani@gmail.com

Kash Khorasani
Concordia University
1455 De Maisonneuve Blvd. West,
Montreal, Quebec, Canada
Email:kash@ece.concordia.ca

Tiberiu Letia
Technical University Cluj-Napoca
15 Constantin Daicoviciu Street,
Cluj-Napoca, Romania
Email:tsletia@gmail.com

Roxana-Elena Tudoroiu
Technical University Cluj-Napoca
15 Constantin Daicoviciu Street,
Cluj-Napoca, Romania
Email: tudelena@excite.com

Abstract— **The main idea of this paper is the real-time implementation of the Fault Detection Kalman Filter Estimators (FDKFE) in a satellite's Reaction Wheels during its scientific mission. We assume that the satellite's reaction wheels are subjected to several failures due to the abnormal changes in power distribution, motor torque, windings current as well as the temperature caused by a motor current increase or friction. The proposed real-time FDKFE strategies consist of two embedded multiple model bank of nonlinear Kalman Filter (Extended/Unscented) estimators. This research work is based on our previous results in this field and we intend to extend this approach by real-time implementations of the developed FDKFE strategies (FDDM-EKF and FDDM-UKF). Furthermore we will construct a benchmark to compare their results to have an overall image how perform these strategies.**

I. Introduction

THE most important faults in the aerospace satellite's systems are the result of unexpected failures, interferences as well as the age of their crucial components. Also the defective measurement and control loops equipment, in particular some of the sensors and actuators should be considered. Whenever these critical situations come out the satellite's systems could lose the control, require much more energy, and could operate harmfully. Therefore to operate in real-time at high energy efficiency and to guarantee the equipment safety and reliability it is important to develop suitable FDKFE strategies able to detect and diagnose any time every faulty satellite's system components and consequently corrective and reconfiguration actions should be initiated promptly. Up to date for the majority part of the satellites, the onboard measurements data acquisition is sent to the land stations relatively frequently. However the operators still remain implicated to monitor some of the telemetry measurements and to diagnose the inconsistency of the operating equipment. Effectively the existing methods to identify and to adjust the equipment failures are mostly labor-intensive task, and consequently sustained, rhythmic and error-prone. In the majority situations the operators inspect telemetry plots manually to determine the satellite's reaction wheels health motors current. Also they use statistical evaluation that still necessitates considerable human knowledge, consequently error-prone that could generate harshly equipment operation. In these circumstances the problem of satellite monitoring and fault diagnosis becomes a critical issue, very complex that need to be implemented in mainframe environment using more sophisticated control systems and artificial intelligent strategies. Therefore the objective of our research is to develop more proficient, accurate and reliable real-time FDKFE strategies based on the nonlinear estimation techniques.

II. REACTION WHEEL's DYNAMICS

The Reaction Wheel (RW) model is developed based on detailed schematics shown in Figure 1 [3]. In a state-space representation associated to the RW actuator healthy mode (j=1) or its faulty modes ($j=\overline{1,N}$) we could write

(i) State Equation:

$$\begin{bmatrix} x_{j,1}(k+1) \\ x_{j,2}(k+1) \end{bmatrix} = \begin{bmatrix} T_s G_d \omega_d (f_1(x_{j,1}(k), x_{j2}(k)) - f_3(x_{j,2}(k)) \\ \dfrac{T_s[k_f x_{j,1}(k)(1+f_2(x_{j,2}(k))) - \tau_c f_4(x_{j,2}(k))]}{J} + (1 - \dfrac{T_s \tau_v}{J}) x_{j,2}(k) \end{bmatrix} + \\ T_s \begin{bmatrix} G_d \omega_d u(k) \\ \dfrac{\tau_{noise}}{J} \end{bmatrix} + w_j(k) := F_j(x_{jk}, u_k) + w_{jk} \tag{1}$$

(ii) Output equation:

$$\begin{bmatrix} y_{j,1}(k) \\ y_{j,2}(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{j,1}(k) \\ x_{j,2}(k) \end{bmatrix} + \upsilon_j(k) := H_j(x_{jk}, u_k) + v_{jk} \tag{2}$$

where the j-state vector mode is defined as:

$$x_j(k) := x_{jk} := \begin{bmatrix} x_{j,1}(k) \\ x_{j,2}(k) \end{bmatrix} = \begin{bmatrix} I_{m,j}(k) \\ \omega_{w,j}(k) \end{bmatrix} \tag{3}$$

The input command $u(k) = V_{com}$ is generated by the attitude controller, and it represents the command voltage applied to the reaction wheel motor axis. The functions H_s, H_b, H_f that appear in the reaction wheel diagram shown in Figure 1 represent the discontinuous Heaviside functions and the sign (.) block function represents the sign function. Also T_s is the sampling period of the discrete-time model of the reaction wheel actuator, w_{jk} and v_{jk} that appear in Figure 2 are the process and measurement noise, assuming that they are independent white Gaussian random processes with zero mean and covariance matrices:

$$E[w_n w_n^T] = \begin{cases} Q_w, n = k \\ 0, n \neq k \end{cases} \quad E[v_n v_n^T] = \begin{cases} R_v, n = k \\ 0, n \equiv k \end{cases} \tag{4}$$

The other parameters indicated in the model are identical to the reaction wheel parameters that are used in [3], and [6]-[9].

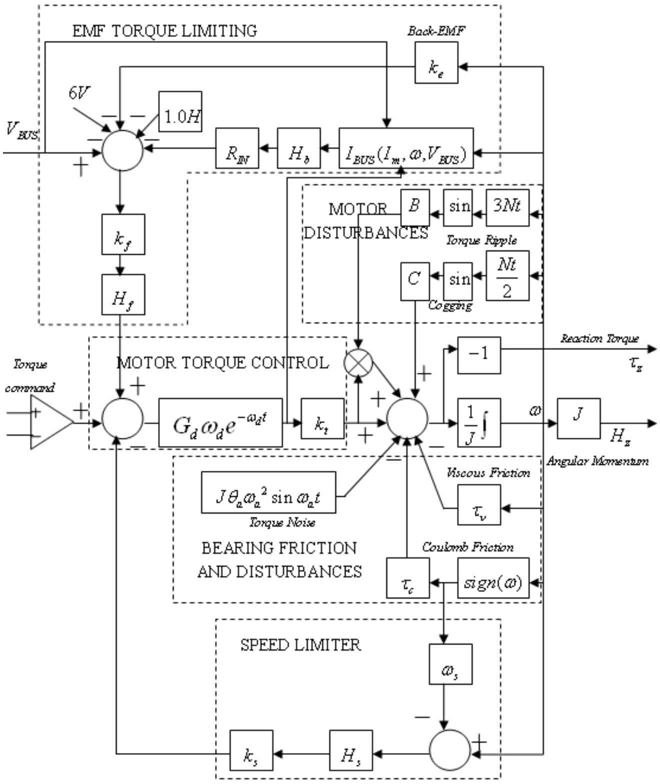


Fig.1 A detailed block diagram of a high fidelity reaction wheel model

III. Extended Kalman Filter Estimator (EKF)

Consider the dynamics of a linear stochastic system expressed in the state-space difference representation

$$x_{k+1} = Fx_k + Gu_k + w_k \tag{5}$$

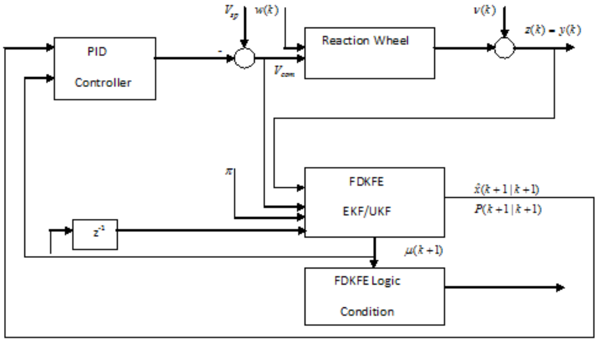


Fig. 2 The satellite's embedded FDKFE integrated structure

$$y_k = Hx_k + v_k \tag{6}$$

The process and measurement noise have normal probability distributions governed by

$$p(w) \sim N(0, Q_w)$$

$$p(v) \sim N(0, R_v) \tag{7}$$

The covariance matrices Q_w (process noise covariance) and R_v (measurement noise covariance) might change with each time step or measurement, but in our approach we assume that they are constant. Due to the process noise injected in the state space equation (5)-(6), the state vector $x_k \in R^n$ becomes random variable with its distribution approximated by a Gaussian distribution function $p(x) \sim G(\hat{x}, P_x)$.

Considering the nonlinear model of a reaction wheel as given by (1)-(2), the linearized state transition and observation matrices are defined according to the following Jacobeans

$$\begin{aligned} F_k &= \frac{\partial f(x_k, u_k)}{\partial x_k}\Big|_{\hat{x}_{k-1|k-1}, u_k} = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix} \\ H_k &= \frac{\partial h(x_k, u_k)}{\partial x_k}\Big|_{\hat{x}_{k|k-1}, u_k} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{8}$$

where G is an identity matrix. The EKF algorithm used for the state estimation of this dynamical system is now given according to the following steps [2]:

Step1. Initialization: for $k = 0$, set:

$\hat{x}_{0|0} = E[x_0], P_{0|0} = E[(x_0 - \hat{x}_{0|0})(x_0 - \hat{x}_{0|0})^T]$ representing the distribution of initial state estimate and its covariance matrix

Step2. Prediction step: Predict the state and the estimated covariance according to

$$\begin{aligned} \hat{x}_{k|k-1} &= f(\hat{x}_{k-1|k-1}, u_k) \\ P_{k|k-1} &= F_k P_{k-1|k-1} F_k^T + Q_w \end{aligned} \tag{9}$$

Step3. Update step:

3.1 Innovation or measurement residual

$$\tilde{y}_k = z_k - h(\hat{x}_{k|k-1}, u_k) \tag{10}$$

3.2 Innovation or residual covariance

$$S_k = H_k P_{k|k-1} H_k^{\ T} + R_v \tag{11}$$

3.3 Optimal Kalman filter gain

$$K_k = P_{k|k-1} H_k^{\ T} S_k^{\ -1} \tag{12}$$

3.4 Update state estimate

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k \tag{13}$$

3.5 Update estimate covariance

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \tag{14}$$

The covariance matrices Q_w (process noise covariance) and R_v (measurement noise covariance), together with the initial error covariance $P_{0|0}$ are the three tuning parameters in the EKF algorithm.

The matrices Q_w and R_v are determined empirically and account for uncertainty in the tracking data. Setting these matrices "properly" significantly contributes in making the EKF filter robust. The error covariance matrix P indicates uncertainty in the state estimate and provides criterion for the error bound.

IV. The unscented kalman filter estimator (UKF)

The Unscented Kalman Filter (UKF) is based on the unscented transformation (UT) which addresses the general problem of state estimation of a system that is governed by a nonlinear stochastic difference state-space representation [2], [4]-[5]:

$$\begin{aligned} \mathbf{x}_{k+1} &= F(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k \\ \mathbf{y}_k &= H(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k \end{aligned} \tag{15}$$

The critical operation that is performed in the Kalman filter is propagation of a Gaussian random state variable $\mathbf{x}_k \in \mathbf{R}^n$ through the system dynamics. In the Extended Kalman Filter (EKF) estimator the Gaussian random state variable is propagated analytically through the first-order linearization of the nonlinear system. This can introduce large errors in the true *a posterior* mean and covariance of the transformed Gaussian random state variable, leading to sub-optimal performance and sometimes divergence of the EKF estimator.

The UKF estimator is developed as an alternative to the EKF estimator and addresses this problem by using a deterministic sampling approach.

Using the principle that a minimal set of carefully chosen weighted sample points, called sigma points, can be used to parameterize mean and covariance, the UKF estimator yields superior performance when compared to the EKF estimator, especially for nonlinear systems.

These sigma points should completely capture the true mean and covariance of the Gaussian random state variable, and are propagated through the true nonlinear system dynamics. In it was shown that a random n-dimensional state variable x with mean $\overline{x}$ and covariance P could be approximated by $2n+1$ weighted sigma points X_i, $i = 1,....,2n+1$, that are given by:

$$\begin{aligned} X_0 &= \overline{x}\ , \ W_0 = \frac{k}{n+k} \\ X_i &= \overline{x} + (\sqrt{(n+k)P})_i\ , \ W_i = \frac{1}{2(n+k)} \\ X_{i+n} &= \overline{x} - (\sqrt{(n+k)P})_i\ , \ W_{i+n} = \frac{1}{2(n+k)} \end{aligned} \tag{16}$$

where k is a scalar that provides an extra degree of freedom for "fine tuning" the higher order moments of the approximation, and can be used to reduce the overall prediction error, $(\sqrt{(n+k)P})_i$ is the i-th row or column of the matrix square root of $(n+k)P$, and W_i is the weight associated with the $i-th$ point X_i.

The cloud of the transformed sigma points distribution X_{k+1} given by

$$X_{k+1} = F(X_k, u_k) \tag{17}$$

captures the *a posterior* mean that is given by the weighted average of the transformed sigma points:

$$\overline{x} = \sum_{i=0}^{2n} W_i X_{k+1,i} \tag{18}$$

and the covariance (the weighted outer product of the transformed sigma points) given by

$$P = \sum_{i=0}^{2n} W_i (X_{k+1,i} - \overline{x})(X_{k+1,i} - \overline{x})^T \tag{19}$$

In some applications the sampling rate could be an important source of degrading the UKF estimator performance.

The main advantage of the UKF estimator is that it is suitable for most general class of process models, due to the fact that the mean and covariance are calculated using standard vector matrix operations, and the implementation is extremely fast since UKF does not require calculation of Jacobean matrices that could in certain cases lead to numerical complications and difficulties.

Also the sigma points capture the mean and the covariance irrespective of the choice of matrix square root that is used.

Numerically efficient and stable methods such as the Cholesky decomposition [5] can be used for this purpose.

Similar to EKF estimator the UKF estimator design steps for the both phases (prediction and correction) are well presented in figure 3.

V. Fault Detection Strategy Design

The multiple model approach for fault detection and diagnosis assumes that the actual system at any time can be modeled sufficiently accurately by the following jump Markov hybrid nonlinear system [1]:

$$\begin{aligned} & x(k+1) = F(k, m(k+1), x(k), u(k)) + T(k, m(k+1)) w(k, m(k+1)) \\ & x(0) : N(\hat{x}_0, P_0) \\ & z(k) = G(k, m(k), x(k), u(k)) + v(k, m(k)) \end{aligned} \tag{20}$$

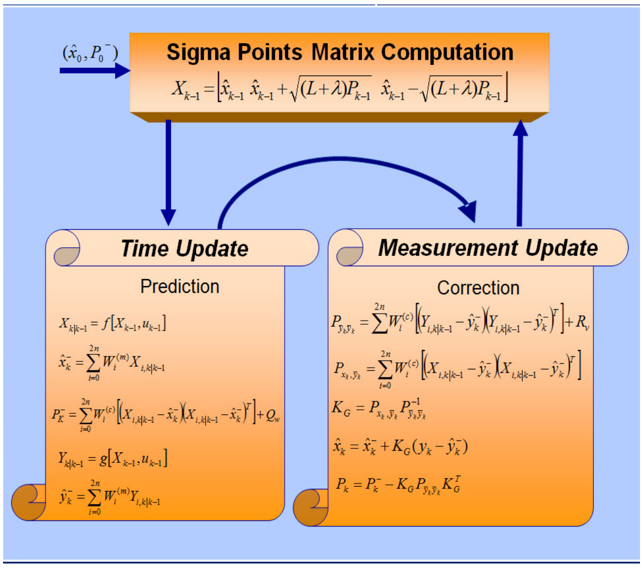

$(\hat{x}_0, P_0^-)$
Sigma Points Matrix Computation
$X_{k-1} = \left[\hat{x}_{k-1}\ \hat{x}_{k-1} + \sqrt{(L+\lambda)P_{k-1}}\ \ \hat{x}_{k-1} - \sqrt{(L+\lambda)P_{k-1}}\right]$
Time Update
Prediction
$X_{k|k-1} = f[X_{k-1}, u_{k-1}]$
$\hat{x}_k^- = \sum_{i=0}^{2n} W_i^{(m)} X_{i,k|k-1}$
$P_K^- = \sum_{i=0}^{2n} W_i^{(c)} \left[(X_{i,k|k-1} - \hat{x}_k^-)(X_{i,k|k-1} - \hat{x}_k^-)^T\right] + Q_w$
$Y_{k|k-1} = g[X_{k-1}, u_{k-1}]$
$\hat{y}_k^- = \sum_{i=0}^{2n} W_i^{(m)} Y_{i,k|k-1}$
Measurement Update
Correction
$P_{\bar{y}_k \bar{y}_k} = \sum^{2n} W_i^{(c)} \left[(Y_{i,k|k-1} - \hat{y}_k^-)(Y_{i,k|k-1} - \hat{y}_k^-)^T\right] + R_v$
$P_{x_k, \bar{y}_k} = \sum_{i=0}^{2n} W_i^{(c)} \left[(X_{i,k|k-1} - \hat{y}_k^-)(X_{i,k|k-1} - \hat{y}_k^-)^T\right]$
$K_G = P_{x_k, \bar{y}_k} P_{\bar{y}_k \bar{y}_k}^{-1}$
$\hat{x}_k = \hat{x}_k^- + K_G(y_k - \hat{y}_k^-)$
$P_k = P_k^- - K_G P_{\bar{y}_k \bar{y}_k} K_G^T$

Fig. 3 The block scheme of the UKF estimator

The mode of the system (normal or faulty) at time k is selected by a discrete process m_j and modeled as a s-state, first-order Markov chain with transition probabilities $\pi_{ij}(k)$ given by:

$$\pi_{ij}(k) = P\{m_j(k+1) \mid m_i(k)\},\ \forall m_i, m_j \quad S \qquad (21)$$

and

$$0 \le \pi_{ij}(k) \le 1,\ i = \overline{1,N}, j = \overline{1,N}, \qquad \sum_j \pi_{ij}(k) = 1,\ \ i = \overline{1,N} \qquad (22)$$

The initial state distribution of the Markov chain is $\pi(0) = [\pi_1\ \pi_2\ \pi_3 \ldots\ldots \pi_N]$, where

$$0 \le \pi_j \le 1,\ \forall j = \overline{1, N},\ \sum_{j=1}^{N} \pi_j = 1 \qquad (23)$$

and where $x(k)$ is the state vector, $z(k)$ is the mode-dependent measurement vector, and $u(k)$ is the control input vector. The process and measurement noise vectors $w(k)$ and $v(k)$, respectively, are mutually independent, additive, white Gaussian of zero mean and covariance matrices $Q_w(k)$ and $R_v(k)$, and are independent of the initial state $x(0)$. In expression (44), P{.} denotes the probability operator. The event that m_j is in effect at time k is denoted as $m_j(k) = \{m(k) = m_j\}$, and $S = \{m_1, m_2, \ldots, m_N\}$ represents the set of all possible system modes. The system may jump from one such system to another at a random time. It can be observed from that the state vector observations are in general noisy and dependent. Therefore, the mode information is embedded in the measurement sequence. The system mode sequence is an indirectly observed Markov chain, from which the transition probability matrix $\pi = \{\pi_{ij}\}$ represents a design parameter. The simulation results and our comparative studies are presented in the next section.

The root-causes of faults injected in the reaction wheels are due to the following sources:

(i) unexpected viscous friction changes generating anomalies in the temperature *T*,

(ii) unexpected changes in the bus voltage V_{bus},

(iii) loss of effectiveness in the motor torque as represented by unexpected changes in the coefficient k_t.

The FDKFE strategy for the hybrid system can be stated as that of determining the current model state. In other words, it involves determining whether the normal or a faulty mode is currently in effect from a sequence of noisy observations. How to design set of modes to represent the possible system modes is a key issue in multiple model approach, which is problem-dependent. This design should be achieved by attempting to have models (approximately) that represent or cover all possible system modes at any given time. This represents the model set design that is critical for multiple model based FDKFE. To design a good set of models requires *a priori* knowledge of possible faults in the system. In application of multiple model estimation techniques for fault detection and diagnosis, the following tasks should be implemented [1]:

a) model set design,
b) filter selection,
c) estimate fusion, and
d) filter re-initialization.

Filter selection deals with the problem of selecting a model-based recursive filter such as Kalman Filter for each model of the nonlinear system. The estimate fusion task combines model-conditional estimates to obtain an overall estimate. Towards this end, three approaches could be investigated, namely soft, hard and random decisions. The procedure for reinitializing each single-model based filter from time to time is of significant importance for multi-model estimation. The simplest approach for reinitializing each filter is to use its previous state estimate and filter covariance at the current cycle. In this case filters are operating in parallel and no interactions exist among them.

However, this may lead to unsatisfactory performance when the system structure or its mode changes. For this reason, it would be more appropriate to reinitialize each filter using the previous overall state estimate and covariance matrix which does lead to an interacting multiple model estimation technique. For each faulty mode corresponding to a set of possible ACS reaction wheel faults and a normal operating mode, one can apply an unscented Kalman filter based on measurements collected from reaction wheels angular velocity vector ϖ. The input considered can be taken as the torque command voltage vector *u* that is generated by, e.g. a PID controller. The dynamics of an unscented Kalman filter associated with each mode is described by the following nonlinear state space representation:

$$\begin{aligned} x_j(k+1) &= F_j(k, x_j(k), u_j(k)) + T_j(k) w_j(k) \\ z_j(k) &= G_j(k, x_j(k), u_j(k)) + v_j(k) \end{aligned} \qquad (24)$$

where $x_j = \varpi(k)$ and the subscript j denotes the quantities pertaining to mode m_j. The nonlinear functions F_j, G_j and the weighting matrix T_j may have different structures for different values of j. The process noise and measurement noise vectors w_j and v_j are white Gaussian of zero mean with covariance matrices Q_{wj} and R_{vj}, respectively. In principle the probability of a given model matches the system mode provides the required information for the fault detection and diagnosis. Taking into account historical behavior of modes at time $\boldsymbol{k}$ ensures that the interacting multiple model algorithm yields a good estimate.

Consequently, exponential increase in complexity of a detection algorithm is avoided by mixing previous estimates at beginning of each cycle.

The model probabilities provide an indication of the mode in effect at any given time, and therefore can provide an indication of the reaction wheel actuator fault. By using model probabilities information, both fault detection and diagnosis can be achieved. This decision making process is formally stated according to:

$$\begin{aligned} &\mu_j(k+1) = \max_j \, \mu_j(k+1) = \mu \\ &\textit{If } \; \mu > \mu_T \; \textit{then } \; \text{mod}\,e \;\; j \;\; \textit{is faulty} \\ &\textit{Otherwise no fault occurs} \end{aligned} \tag{25}$$

where $\mu_{Threshold}$ represents the fault detection threshold value. The interacting estimation algorithm runs each parallel filter banks only once in each cycle. Each of these filters at time $t_{k+1} = k+1$ has its own input, the state estimate at time t_k, $\hat{x}^0(k|k)$, and its own covariance matrix, $P^0(k|k)$, which form a valid quasi-sufficient statistics of all the past information, under the assumption that model of each filter matches the system mode.

The above decision rule yields not only fault detection capability but also information about the type (sensor or actuator), the location (which sensor or actuator), the size (total failure or partial failure with fault severity), and the fault occurrence time.

VI. Nhe Embedded Fault Detection Kalman Filter Estimators Interacting Strategy

The embedded structure of an Interactive Multiple Model algorithm and EKF/UKF estimators is included in the references [1]-[2], [4]-[9]. Due to space limitations only the UKF based procedures is described below, the algorithm steps remaining almost the same for the both cases.

Step 1: Interaction and mixing of the estimates [1]

1.1 Compute the predicted mode probability from k to $k+1$

$$\hat{\mu}_j(k+1|\,k) = \sum\nolimits_1^N \pi(i,j)\mu_i(k) \tag{26}$$

1.2 Compute the mixing probability at k:

$$\mu_{ij}(k) = \frac{\pi(i,j) \;\; \mu_i(k)}{\hat{\mu}_j(k+1|\,k)} \tag{27}$$

1.3 Compute the mixing estimate at k:

$$\hat{x}_{j0}(k \mid k) = \sum\nolimits_1^N \hat{x}_i(k \mid k) \times \mu_{ij}(k) \tag{28}$$

1.4 Compute the mixing covariance at k, that is, P_0

$$P_{j0}(k|k) = \sum\nolimits^N [P_i(k|k) + (\hat{x}_{j0}(k|k) - \hat{x}_i(k))(\hat{x}_{j0}(k|k) - \hat{x}_i(k|k))^T] \times \mu_{ij}(k) \tag{29}$$

Step 2: Model conditional filtering [1]

(i) Prediction step [2], [4]-[5]:

2.1 Compute the global state sigma points matrix:

$$X_{k|k} = [\hat{x}(k|k) \;\; \hat{x}(k|k) + \sqrt{(L+\lambda)P(k|k)} \;\; \hat{x}(k|k) - \sqrt{(L+\lambda)P(k|k)}] \tag{30}$$

where the state covariance matrix $P(k|k)$ is updated for each time increment.

2.2 Compute the transformed global state sigma points matrix from k to $k+1$:

$$X_{k+1|k} = F(k, X_{k|k}, u_k) \tag{31}$$

2.3 Compute the predicted global state from k to $k+1$:

$$\hat{x}(k+1 \mid k) = \sum_{i=0}^{2n} W_i^{(m)} X_{i,k+1|k} \tag{32}$$

2.4 Compute the global predicted covariance from k to $k+1$:

$$\begin{aligned} P(k+1|\,k) = {} & (W_0^{(c)} + 1 - \alpha^2)(X_{0,k+1|k} - \hat{x}(k+1|\,k))(X_{0,k+1|k} - \hat{x}(k+1|\,k))^T + \\ & + \sum_{i=1}^{2n} W_i^{(c)}[(X_{i,k+1|k} - \hat{x}(k+1|\,k))(X_{i,k+1|k} - \hat{x}(k+1|\,k))^T] + Q_w \end{aligned} \tag{33}$$

2.5 Compute the transformed global output sigma points matrix from k to $k+1$:

$$Y_{k+1|k} = G(k, X_{k+1|k}, u_k) \tag{34}$$

2.6 Compute the predicted global output from k to $k+1$

$$\hat{y}(k+1 \,|\, k) = \sum_{i=0}^{2n} W_i^{(m)} Y_{i,k|k-1} \tag{35}$$

2.7 Compute the measurement residual for each mode j at $k+1$:

$$v_j(k+1) = z(k+1) - \hat{y}_j(k+1 \,|\, k) \tag{36}$$

where $\hat{x}_j(k+1 \,|\, k)$ represents the j-th component of the global predicted state $\hat{x}(k+1 \,|\, k)$ (the mode j), and $z(k+1)$ represents the new available measurement for the active mode in effect, let say i, namely

$$z(k+1) = z_i(k+1) \tag{37}$$

2.8 Compute the global predicted output covariance matrix $P_{\tilde{y}\tilde{y}}(k+1 \,|\, k)$

$$\begin{aligned} P_{\tilde{y}\tilde{y}}(k+1 \,|\, k) = {} & (W_0^{(c)} + 1 - \alpha^2)(Y_{0,k|k-1} - \hat{y}(k+1 \,|\, k))(Y_{0,k|k-1} - \hat{y}(k+1 \,|\, k))^T + \\ & + \sum_{i=1}^{2n} W_i^{(c)}[(Y_{i,k|k-1} - \hat{y}(k+1 \,|\, k))(Y_{i,k|k-1} - \hat{y}(k+1 \,|\, k))^T] + R \end{aligned} \tag{38}$$

2.9 Compute the global cross-covariance matrix

$$\begin{aligned} P\tilde{x}\tilde{y}(k+1|\,k) = {} & (W_0^{(c)} + 1 - \alpha^2)(X_{0,k|k-1} - \hat{x}(k+1|\,k))(Y_{0,k|k-1} - \hat{y}(k+1|\,k))^T + \\ & + \sum_{i=1}^{2n} W_i^{(c)}[(X_{i,k|k-1} - \hat{x}(k+1|\,k))(Y_{i,k|k-1} - \hat{y}(k+1|\,k))^T] \end{aligned} \tag{39}$$

2.10 Compute the global Kalman filter gain at *k*+1:

$$K_G(k+1) = P_{\tilde{x}\tilde{y}}(k+1 \mid k)(P_{\tilde{y}\tilde{y}}(k+1 \mid k))^{-1} \quad (40)$$

(ii) Correction step:

2.11 Update the state estimated at k+1

$$\hat{x}_j(k+1 \mid k+1) = \hat{x}_j(k+1 \mid k) + K_{G,j}(k+1)\nu_j(k+1) \quad (41)$$

2.12 Update the state covariance matrix

$$P_j(k+1|k+1)=P_j(k+1|k)-K_{G,j}(k+1)(P_{\tilde{y}\tilde{y}}(k+1|k)^{-1}(K_{G,j}(k+1))^T \quad (42)$$

<u>**Step 3:** Update the mode probability at *k*+1</u>

3.1 Compute the likelihood function at *k*+1

$$L_j(k+1)=\frac{1}{\sqrt{|(2\pi)P_{j,\tilde{y}\tilde{y}}(k+1|k)|}}\exp[-\frac{1}{2}\nu_j^T(k+1)(P_{j,\tilde{y}\tilde{y}}(k+1|k))^{-1}\nu_j(k+1)] \quad (43)$$

3.2 Update the mode probability at *k*+1:

$$\mu_j(k+1) = \frac{\mu_j(k+1 \mid k)L_j(k+1)}{\sum_{1}^{N} \mu_j(k+1 \mid k)L_j(k+1)} \quad (44)$$

<u>**Step 4:** Fault detection at *k*+1</u>

4.1 Compute the mode probability vector at *k*+1:

$$\vec{\mu}(k+1)=\left[\mu_1(k+1)\ \ \mu_2(k+1)\ \ \mu_3(k+1)\dots\mu_N(k+1)\right] \quad (45)$$

4.2 Obtain the maximum value of the mode probability vector:

$$\mu_{FDD\ \max} = \max_j \{\mu_j(k+1))\ , \quad (46)$$

4.3 Determine the index of the maximum value of mode probability vector

$$j = find(\vec{\mu} == \max(\vec{\mu})) \quad (47)$$

and subsequently assign:

$$index = j \quad (48)$$

<u>**Step 5**: Fault decision – FDD logic</u>

if $\mu_{FDD\max} > \mu_{Threshold}$ the fault occurs, (49)

Otherwise no fault occurs.

VII. Real-Time Control and FGKE Strategy Implementation

The control system literature rarely includes extensively the real-time software and hardware implementation aspects, and it doesn't pay attention beyond algorithms and sampling time selection. Normally the implementation aspect and real-time system design are connected together but in the most cases this connection is always ignored. Moreover the real-time system design is treated from control perspective ignoring the implementation aspects of the control algorithms. However in the last years the real-time implementation and design aspects get more transparency to control engineering field due to advent of software tools like MATLAB/SIMULINK with its RTW (Real-Time Workshop) and the RTWT (Real-Time Windows Target) Toolboxes. Definitely these new real-time platforms do the implementation of real-time experiments easier and save much time but on the other hand they have some drawbacks regarding a good insight view of the real-life problems that could emerge during the real-time implementation of the control system. Building a real-time ACS requires normally two stages: PID controller and estimators design as well as their digital implementations. At controller and estimators design stage some performance indices are defined and the PID controller and EKF and UKF estimators are designed to optimize these indices.

At implementation stage, multiple control tasks should be scheduled to run the FPGA (*Field-Programmable Gate Array*) onboard microprocessor or controller module. We have to take care about task scheduling taking into account the limited available computing resources. Firstly to select the sampling time T_s, it should take into account the limited computation time provided by the hardware such that to avoid the conflict with its computation time delay (control latency). Since the control latency is typically affected by control jitter, delay and loss can occur alternatively in the system at different instants of time. The loss of the ACS control signal $u(k)$ leads to the controller computer failure, unable to update its output during any one sampling interval, and accordingly the one step delayed input $u(k-1)$ it will be applied again. Since this situation may well occur accidentally at any instant of time, the failure to deliver a control signal can be treated as a casual ACS input disturbance $u_p(k)$. Anyway this interaction between control performance and task scheduling will be investigated in the future work.

VIII. Simulation Results

The real-time platform used to perform these real-time simulations was a MATLAB R2007b with SIMULINK running on the two processors WINDOWS OS machine. The simulation results are obtained for a sampling time T_s equal to 1s, but a carefully selection of it and the interaction between control performance and task scheduling will be done in the future work. In spite of a large number of scenarios, to have an overall image about the estimator's performance, we present in figures (4)-(5) only the partial simulation results for the both real-time embedded fault detection strategies. In these figures we will present in terms of the mode probability index the transient and the steady state of different fault modes occurrences in a multi-fault sequence. These figures reveal in the same time the robustness of the both estimators to the noise level and to the probability matrix. Also the estimator's performance comparison is available in the benchmark presented in Table 1.

IX. Conclusions

In this paper, we have studied the possibility of using two interactive multiple models based on Extended and Unscented Kalman Filter estimators to detect and diagnose the faults in reaction wheels of the attitude control system (ACS) in a satellite. The main contributions in our research are summarized briefly as follows:

(a) Detection and identification of reaction wheel faults of the ACS due to a number of possible

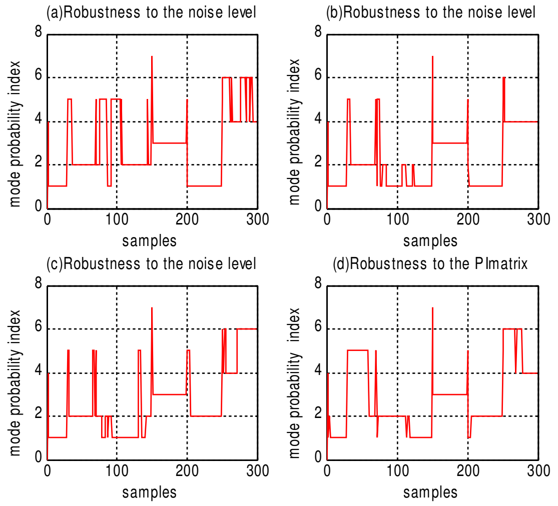


Fig. 4 The performance in terms of mode probabilities index for robustness analysis using the FD-EKF strategy

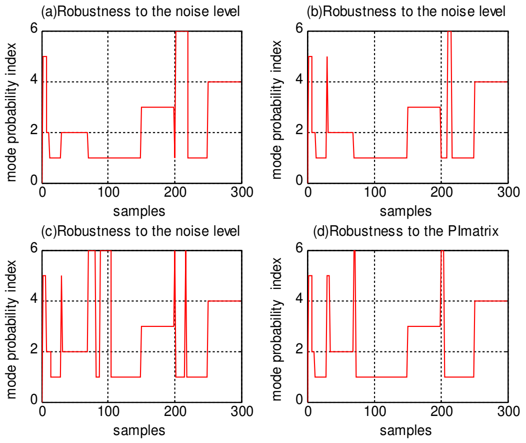


Fig. 5 The performance in terms of mode probabilities index for robustness analysis using the FD-UKF strategy

TABLE 1
THE PERFORMANCE COMPARISON OF THE FDKFE STRATEGIES

Item	Performance	KF	EKF	UKF
1	Accuracy Degree for modeled faults	Very good	Good	Very Good
2	Accuracy Degree for un-modeled faults	Small	Small	Good
3	Robustness to R ,Q matrices	Not Robust	Not Robust	Robust
4	Robustness to the matrix probabilities, π	Not Robust	Not Robust	Robust

*Embedded estimators: KF-Standard Kalman Filter, EKF-Extended Kalman Filter, UKF-Unscented Kalman Filter.

sources that generate soft and hard anomalies during a scientific mission of a satellite,

(b) Implementation of a bank of real-time parallel Kalman filters for faulty modes covering a quite large set of the most likely and commonly possible scenarios of reaction wheel failures,

(c) Detection and diagnosis of both partial and significant reaction wheel failures for different scenarios using the real-time EKF and UKF estimation algorithms,

(d) Comparison of performance capabilities and advantages of real-time UKF estimation algorithm with respect to the real-time EKF estimation algorithm, and

(e) Robustness analysis of both real-time estimation algorithms to the selection of model transition probabilities, modeling errors, and noise statistics under different scenarios.

The approach proposed in this paper is probabilistic in nature and yields results that are more accurate and having good fault classification capabilities than the spectral analysis that is well studied in the literature. Based on fault identification analysis that is carried it can be observed that the real-time UKF estimation algorithm is robust to modeling uncertainties, and to statistics of noise measurements and process noise. The both real-time algorithms work similar to a real-time neural network estimator and classifier employed to perform satisfactory FDKFE. Compared to a real-time neural network estimator and classifier, the real-time UKF estimation algorithm doesn't need an on-line training that takes an extensive amount of computational resources. Compared to the above approaches and real-time EKF estimation algorithm, the real-time UKF estimation algorithm performs better and has a much less computational burden and complexity, and it furthermore operates much faster. Perhaps the biggest drawback of predictive model-based approaches is the need for a suitable quantity of data for training and testing the system during the development phase [10], [13]. Moreover, stability of the real-time UKF estimation algorithm still remains an open question, which needs further investigation. Also, the behavior of the real-time UKF estimation algorithm for diagnosis in a fast or rapidly changing process dynamics needs to be explored further.

References

[1]. Y. Zhang and Rong-Li Xiao, “Detection and Diagnosis of Sensor and Actuator Failures using IMM Estimator”, *IEEE Transaction on Aerospace and Electronic Systems*, Vol.34, No.4, pp. 1293-1311, 1998.

[2]. S. Haykin, *Kalman Filtering and Neural Networks*, John Willey & Sons, 2001.

[3]. B. Bialke, “High Fidelity Mathematical Modeling of Reaction Wheel Performance”, *Advances in the Astronautical Sciences*, 1998, pp. 483-496.

[4]. S. J. Julier and J. K. Uhlman, “A New Extension of the Kalman Filter to Nonlinear Systems”. *Proceedings of AeroSense, Proceedings of the11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, 1997, pp.182-193.

[5]. E. A. Wan, R. Merwe, “The Unscented Kalman Filter for Nonlinear Estimation”, *Proceedings IEEE Symposium,* Alberta, Canada, 2000.

[6]. N. Tudoroiu and K. Khorasani, “Fault Detection and Diagnosis for Reaction Wheels of Satellite’s Attitude Control System Using a Bank of Kalman Filters”, *Proceedings of IEEE International Symposium on Signal, Circuits and Systems*, Iasi, Romania, 2005, pp. 199-202.

[7]. N. Tudoroiu and K. Khorasani, “Fault Detection and Diagnosis for Satellite’s Attitude Control System using an Interactive Multiple Model (IMM) Approach”, *Proceedings of the Conference on Control Applications*, Toronto, Canada, 2005.

[8]. N.Tudoroiu, K. Khorasani, “Satellite Fault Diagnosis using a Bank of Interacting Kalman Filters”, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 43, No.4, 2007, pp. 1334-1350.

[9]. N. Tudoroiu, E. Sobhani-Tehrani, and K. Khorasani, “Interactive Bank of Unscented Kalman Filters for Fault Detection and Isolation in Reaction Wheel Actuators of Satellite Attitude Control System”, *IECON’2006,The 32nd Annual Conference of the IEEE Industrial Electronics Society*, 2006.

[10]. J. D. Boskovic and K. M. Mehra, 2001, “Hybrid Fault Tolerant Control of Aerospace Vehicle”, *Proceedings of the IEEE International Conference on Control Applications*, Mexico, 2001, pp.441-446.

[11]. Z. Li, L. Ma, and K. Khorasani, “Fault Detection in Reaction Wheel of a Satellite using Observer-based Dynamic Neural Networks”, *Proceedings of International Symposium on Neural Networks*, 2005.

[12]. E. Sobhani, K. Khorasani, and S. Tafazoli, “*Dynamic Neural Network-based Estimator for Fault Diagnosis in Reaction Wheel Actuator of Satellite Attitude Control System*”, *Proceedings of the International Joint Conference on Neural Networks*, 2005.

[13]. R. J. Patton, P. M. Frank, and R. N. Clark, “Fault Diagnosis in Dynamic Systems, Theory and Applications”, Englewood Cliffs, NJ: Prentice Hall, 1989.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Application of optimal settings of the LMS adaptive filter for speech signal processing

Jan Vaňuš
VŠB-Technical University of
Ostrava, FEECS, Dept. of
Electrical Engineering
17. listopadu 15, 70833 Ostrava-
Poruba, Czech Republic
Email: jan.vanus@vsb.cz

Vítězslav Stýskala
VŠB-Technical University of
Ostrava, FEECS, Dept. of
Electrical Engineering
17. listopadu 15, 70833 Ostrava-
Poruba, Czech Republic
Email: vitezslav.styskala@vsb.cz

***Abstract*–This paper describes a proposition of the method for optimal adjustment parameters of the adaptive filter with LMS algorithm in the practical application of suppression of additive noise in a speech signal for voice communication with the control system. By the proposed method, the optimal values of parameters of adaptive filter are calculated with guarantees the stability and convergence of the LMS algorithm. The DTW criterion is used for the quality assessment of speech signal processing obtained from output of adaptive filter with LMS algorithm. In the experimental section is described the way of verification of the proposed method on the structure of the adaptive filter with LMS algorithm and on the structure of the adaptive filter with LMS algorithm in application of suppressing noise from speech signal by simulations in MATLAB software and implementation on DSK TMS320C6713.**

***Keywords*–LMS adaptive filter (Least Mean Square), DTW criterion (Dynamic Time Warping), noise canceller.**

I. Introduction

FOR optimum settings of a step size parameter μ and the length M of the adaptive filter with the LMS algorithm is necessary ensuring the stability and convergence of the LMS algorithm. As a result of appropriate setting of the adaptive filter parameters is correct speech signal processing and subsequent correct the isolate words recognition through the use of the DTW criterion.

II. The Adaptive Filter With LMS Algorithm

A. LMS algorithm

Least mean – square (LMS) algorithm was developed by Widrow and Hoff in 1960. This algorithm is a member of stochastic gradient algorithms [2]. The LMS algorithm is a linear adaptive filtering algorithm, which, in general, consists of two basic processes:

a) filtering process, which involves

- Computing the output $y(n)$ of adaptive filter in response to vector input signal $\mathbf{x}(n)$ (1),
- Generating an estimation error $e(n)$ (Fig.5) by comparing this output $y(n)$ with desired response $d(n)$ (Fig.2) (2),

b) An adaptive process (3), which involves the automatic adjustment of the parameters $\mathbf{w}(n+1)$ of the filter in accordance with the estimation error $e(n)$ [3], [1]

$$y(n) = \mathbf{w}^{\mathrm{T}}(n)\,\mathbf{x}(n) \tag{1}$$

$\mathbf{w}(n)$ tap – weight vector,

$$e(n) = d(n) - y(n) \tag{2}$$

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n) \tag{3}$$

$\mathbf{w}(n+1)$ tap – weight vector update,
μ step size parameter.

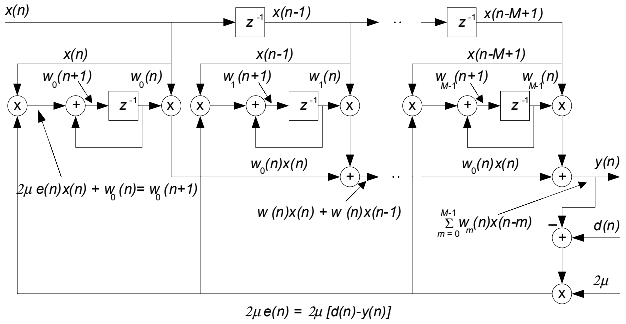


Fig. 1 FIR LMS adaptive filter realization [2].

B. Settings of a step size parameter μ

a) Calculating of a step size parameter μ to ensure the stability of adaptive filter with the LMS algorithm.

For determination, when the LMS algorithm remains stable is necessary find the upper bound of $\mu_{\max}$, that guarantees stability of LMS algorithm (4) [1]

$$\mu_{\max} < \frac{1}{3\operatorname{tr}[\mathbf{R}]} \tag{4}$$

tr[**R**] trace of **R**, which mean sum of the diagonal elements of **R**,

R Toeplitz autocorrelation matrix calculated from vector of input signal $\mathbf{x}(n)$, size **R** is M x M.

Toeplitz autocorrelation matrix **R** is calculated by equation (5)[2]

$$\mathbf{R} = \mathrm{E}[\mathbf{x}(n)\mathbf{x}^{T}(n)]. \tag{5}$$

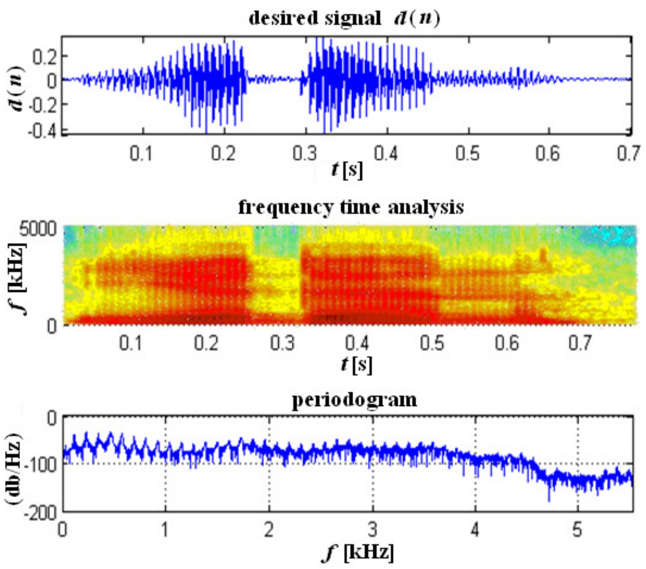


Fig. 2 Waveform, spectrogram (frequency time analysis) and periodogram of power spectral density estimate of desired speech signal $d(n)$ of isolated czech word "jeden" to the input of adaptive filter with LMS algorithm.

The significance of the upper bound of μ, which is provided by (4), is that it can easily be calculated from the filter input samples. Range of μ that is provided by (4) is sufficient for the stability the LMS algorithm, but is not necessary [1].

b) Calculating of a step size parameter μ to ensure the convergence of adaptive filter with the LMS algorithm

Convergence behaviour of LMS algorithm is directly linked to the eigenvalue spread of the autocorrelation matrix **R** and the power spectrum of $\mathbf{x}(n)$. Convergence of the LMS algorithm is directly related to the flatness in the spectral content of the underlying input process. E[$\mathbf{v}(n)$] converges to zero when μ remains within the range of formula (6). E[$\mathbf{v}(n)$] is expectation of weight – error vector $\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{w}_0$.

$$\mu_{\mathrm{conv}} \leq \frac{1}{\lambda_{\max}} \qquad (6)$$

$\lambda_{\max}$ maximum eigenvalue of autocorrelation matrix **R** of the input vector $\mathbf{x}(n)$.

The above range does not necessarily guarantee stability of LMS algorithm. The convergence of LMS algorithm requires convergence of the mean of $\mathbf{w}(n)$ towards $\mathbf{w}_0$ and also convergence of the variance of the elements of $\mathbf{w}(n)$ to some limited values [1]. Vector $\mathbf{w}_0$ is calculated by Wiener – Hopf equation and the superscript "$_0$" indicates the optimum Wiener solution for the Wiener filter [2].

c) Calculating of optimal value of a step size parameter μ_{opt} of adaptive filter with the LMS algorithm

Determination of a step size parameter μ_{opt} value is important to conduct an algorithm LMS. When selecting parameter μ_{opt} terms of a compromise between the two aspects. On the one hand, large values μ can leads quickly to the optimal settings the LMS algorithm for speech signal processing. On the other hand, may increase the value μ of a mistake in the speech signal processing in further steps. Small value μ, on the contrary, ensure the stability and the convergence of LMS algorithm [1].

As a result small value μ is the slowdown in the convergence of LMS algorithm and, consequently, increasing the inaccuracies in the filtration non-stationary signals [9]. For the optimal value of the parameter μ_{opt} is the following equation (7) [1]

$$\mu_{\mathrm{opt}} = \frac{M}{(1+M).\mathrm{tr}[\mathbf{R}]}, \qquad (7)$$

tr[**R**]trace of **R**, which mean sum of the diagonal elements of **R**, M parameter misadjustment.

Parameter misadjustment M is defined as ratio of the steady – state value of the excess mean-square error (MSE) ξ_{excess} to the minimum mean square (MSE) error $\xi_{\min}$.

Table I.

CALCULATED VALUES OF A STEP SIZE PARAMETERS $\mu_{\mathrm{opt}}, \mu_{\max}, \mu_{\mathrm{conv}}$ OF LMS ADAPTIVE FILTER FOR INPUT SIGNAL $\mathbf{x}(n)$ WITH DIFFERENT SSNR VALUES.

SSNR$_a$ = 6,731(dB)	**SSNR$_{w1}$= 18,187(dB)**	**SSNR$_{w2}$= 3,119(dB)**	**SSNR$_{w3}$= –1,783(dB)**
$\mu_{\max}=3,25.10^{-2}$ (M=10%)	$\mu_{\max}=3,54.10^{-2}$ (M=10%)	$\mu_{\max}=2,99.10^{-2}$ (M=10%)	$\mu_{\max}=2,25.10^{-2}$ (M=10%)
$\mu_{\mathrm{conv}}=1,21$ (M=10%)	$\mu_{\mathrm{conv}}=1,221$ (M=10%)	$\mu_{\mathrm{conv}}=1,239$ (M=10%)	$\mu_{\mathrm{conv}}=1,168$ (M=10%)
$\mu_{\mathrm{opt}}=5,9.10^{-3}$ (M=10%)	$\mu_{\mathrm{opt}}=6,4.10^{-3}$ (M=10%)	$\mu_{\mathrm{opt}}=5,4.10^{-3}$ (M=10%)	$\mu_{\mathrm{opt}}=4,09.10^{-3}$ (M=10%)
$\mu_{\mathrm{opt}}=1,08.10^{-2}$ (M=20%)	$\mu_{\mathrm{opt}}=1,18.10^{-2}$ (M=20%)	$\mu_{\mathrm{opt}}=1.10^{-2}$ (M=20%)	$\mu_{\mathrm{opt}}=7,5.10^{-3}$ (M=20%)
$\mu_{\mathrm{opt}}=14,99.10^{-3}$ (M=30%)	$\mu_{\mathrm{opt}}=1,63.10^{-2}$ (M=30%)	$\mu_{\mathrm{opt}}=1,38.10^{-2}$ (M=30%)	$\mu_{\mathrm{opt}}=1,04.10^{-2}$ (M=30%)

$$M = \frac{\xi_{excess}}{\xi_{\min}} = \mu\,\mathrm{tr}[\mathbf{R}]. \qquad (8)$$

The misadjustment M is a dimensionless parameter that provides a measure of how close the LMS algorithm is to optimality in the mean - square – sense.

The smaller M is compared with unity, the more accurate is the adaptive filtering action being performed by the LMS algorithm. Values of misadjustment M are usually the 10%, 20% and 30% (Tab.I), (Tab.III), (Tab.V), (Tab.VI), (Tab.VII) [1].

A value of M = 10% means, that the adaptive system has an MSE only 10 percent greater than $\xi_{\min}$ [8].

III. Determination of the LMS adaptive filter length *M* by way of Widrow method [8]

Time constant τ_{mse} is calculating

$$\tau_{\text{mse}}=\frac{1}{4\mu\lambda}=\frac{M}{4\mu\,\mathrm{tr}[\mathbf{R}]}, \qquad (9)$$

λ eigenvalue of autocorrelation matrix **R** of the input vector **x**(*n*),
tr[**R**]trace of **R**, which mean sum of the diagonal elements of **R**.

When the eigenvalues λ are sufficiently similar for the learning curve to be approximately, fitted by a single exponential, its time constant τ_{mse} may be applied to (9) to give an approximate value of *M* [8].

Values of order *M* LMS adaptive filter can be calculated from input signal **x**(*n*) to adaptive filter (Tab.III)

$$M=\frac{\mathrm{tr}[\mathbf{R}]}{\lambda}. \qquad (10)$$

IV. Using DTW criterion for determination of the adaptive filter length *M*

The correct determination of the adaptive filter length *M* is very important. When the length *M* of the adaptive filter is low, the speech signal processing as a result of a small number of parameters of the adaptive filter is inaccurate. High value of the adaptive filter length *M* lead to inaccurate speech signal processing by influence of the estimator variance increase. In draft method in this work was used DTW criterion for determining value of length *M* of the LMS adaptive filter.

By way of DTW criterion is compare two sequences of vectors: reference vector **P** = [p(1), . . . p(P)] of the length P and test vector **O** = [o(1), . . . o(T)] of the length T [6].

Value of the LMS adaptive filter order *M* is determined by setting values of the order *M* in interval {0 to 150} and calculating of the minimum distance *d* (similarity) between the reference vector **P** (desired signal *d*(*n*) (Fig.2)) and the test sequence vector **O** (error signal *e*(*n*) (Fig.5)). Words are almost never represented by the sequence of the same length P$\neq$ T. The distance *d* between the sequences **O** and **P** is given as minimum distance over the set of all possible paths (all possible lengths, all possible courses) [4]. When the distance *d* was ***d*<0,2**, the word **was recognized**. This value ***d*<0,2** was determined empirically from the measured results of implemented experiments (Tab.II), (Tab.III), (Tab.V), (Tab.VI), (Tab.VII), (Tab.VIII).

Minimum distance computation

$$D(\mathbf{O},\mathbf{P})=\min_{\{C\}} D_c(\mathbf{O},\mathbf{P}) \qquad (11)$$

is simple, when normalization factor N_c is no function of path and is possible write N_c=N for $\forall_c$

$$D(O,P)=\frac{1}{N}\min_{\{C\}}\sum_{k=1}^{Kc} d\left[o\left(t_c(k)\right),p\left(r_c(k)\right)\right]W_c(k) \qquad (12)$$

V. Using of the additive noise in experiments with speech signal

For implementation of experiments are used additive noises with calculated segmental SNR (Signal to Noise Ratio) – SSNR (Tab.IV) for speech signal processing [5]

$$\mathrm{SSNR}=\frac{1}{K}\sum_{i=0}^{L-1}\mathrm{SNR}_i VAD_i, \qquad (13),$$

L is the number of segments of speech signal,
K the number of segments in speech activity,
VAD_i is information about speech activity (values 1 and 0) in *i*-th segment, SNR_i is local (short term) SNR.

Table II.
The values of distance *d* are calculated by comparing of the czech isolated words "jeden" (one) with the words "dva" (two) for up to "pět" (five) and compared the word "dva" (two) with the words "jeden"(one), "tři" (three) for up to "pět"(five).

jeden–jeden	jeden–dva	jeden–tři	jeden–čtyři	jeden–pět
***d* = 0**	*d* = 0,713	*d* = 1,218	*d* = 1,415	*d* = 0,552
dva–jeden	dva–dva	dva–tři	dva–čtyři	dva–pět
d = 0,713	***d* = 0**	*d* = 0,406	*d* = 0,568	*d* = 0,373

Table III.
The values of order *M* of adaptive filter and distance *d* between desired speech signal *d*(*n*) to adaptive filter and error signal *e*(*n*) from adaptive filter calculated by way of widrow methods (simulated in MATLAB).

	SSNR$_a$=6,731(dB)	**SSNRw1=18,187(dB)**	**SSNR$_{w2}$=3,119(dB)**	**SSNR$_{w3}$=–1,783(dB)**
Widrow M=10 %	μ_1=5,9.10^{-3}; ***M*=19** ***d*=9,919.10^{-2}**	μ_1=6,4.10^{-3}; ***M*=17** ***d*=1,697.10^{-1}**	μ_1=5,4.10^{-3}; ***M*=21** *d*=2,919.10^{-1}	μ_1=4,09.10^{-3}; ***M*=26** *d*=2,908.10^{-1}
Widrow M=20 %	μ_2=1,08.10^{-2}; ***M*=19** ***d*=1,181.10^{-1}**	μ_2=1,18.10^{-2}; ***M*=17** *d*=2,54.10^{-1}	μ_2=1.10^{-2}; ***M*=21** *d*=4,152.10^{-1}	μ_2=7,5.10^{-3}; ***M*=26** *d*=4,011.10^{-1}
Widrow M=30 %	μ_3=14,996.10^{-3}; ***M*=19** ***d*=1,357.10^{-1}**	μ_3=1,63.10^{-2}; ***M*=17** *d*=3,194.10^{-1}	μ_3=1,38.10^{-2}; ***M*=21** *d*=4,907.10^{-1}	μ_3=1,04.10^{-2}; ***M*=26** *d*=4,645.10^{-1}

In the paper were used following additive noises for compare of the results of the proposed methods:

- Additive noise $n_a(n)$ (Fig.3) with the ratio of the desired signal $d(n)$ (Fig.2) of isolated czech words "jeden" to noise $n_a(n)$ **SSNR$_a$=6,731(dB)** (Tab.IV).
- White noise $n_{w1}(n)$ with the ratio of the desired signal $d(n)$ (Fig.2) of isolated czech words "jeden" to noise $n_{w1}(n)$ **SSNR$_{w1}$=18,187(dB)** (Tab.IV).
- White noise $n_{w2}(n)$ the ratio of the speech signal of isolated words "jeden" (Fig.2) to noise $n_{w2}(n)$ **SSNR$_{w2}$=3,119(dB)** (Tab.IV).

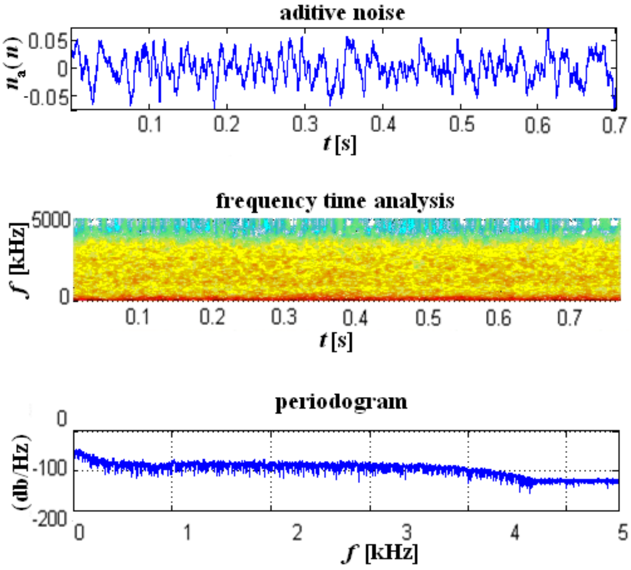


Fig. 3 Waveform, spectrogram (frequency time analysis) and periodogram of power spectral density estimate of desired speech signal $d(n)$ of additive noise $n_a(n)$ with SSNR$_a$=6,731(dB).

TABLE IV.
SEGMENTAL SIGNAL TO NOISE RATIO VALUES CALCULATED FOR THE SPEECH SIGNAL WORD "JEDEN" (FIG.1) TO ADDITIVE NOISE AND TO ADDITIVE WHITE NOISE.

Noise signification	Calculated values of ratio signal to noise
additive noise $n_a(n)$	SSNR$_a$=6,731(dB)
additive white noise 1 $n_{w1}(n)$	SSNR$_{w1}$=18,187(dB)
additive white noise 2 $n_{w2}(n)$	SSNR$_{w2}$=3,119(dB)
additive white noise 3 $n_{w3}(n)$	SSNR$_{w3}$= –1,783(dB)

- White noise $n_{w3}(n)$ the ratio of the speech signal of isolated words "jeden" (Fig.2) to noise $n_{w3}(n)$ **SSNR$_{w3}$=-1,783(dB)** (Tab.IV).

VI. DRAFT DTW METHOD

Draft method for optimal adjustment of a step size parameter μ_{opt} and the length M of the LMS adaptive filter was applied in next steps [7]:

1. Calculation of a step size parameter μ_{opt} optimal value (7) from input signal $x(n)$ (with SSNR) to the LMS adaptive filter (M=10%, M=20%, M=30%) (Tab.I).

2. For reference vector **P** is used desired signal $d(n)$ (Fig. 2) to the LMS adaptive filter.

3. As a test vector **O** was chosen error signal $e(n)$ (Fig.5).

4. Next was calculated the distance d (11), (12) between the signals $d(n)$ and $e(n)$ for sets values of LMS adaptive filter lengths M in interval {1 to 150}.

5. As the optimal value of the LMS adaptive filter order M was chosen value of the adaptive filter length M for minimum distance d (Fig.4) between two compared signals $d(n)$ and $e(n)$.

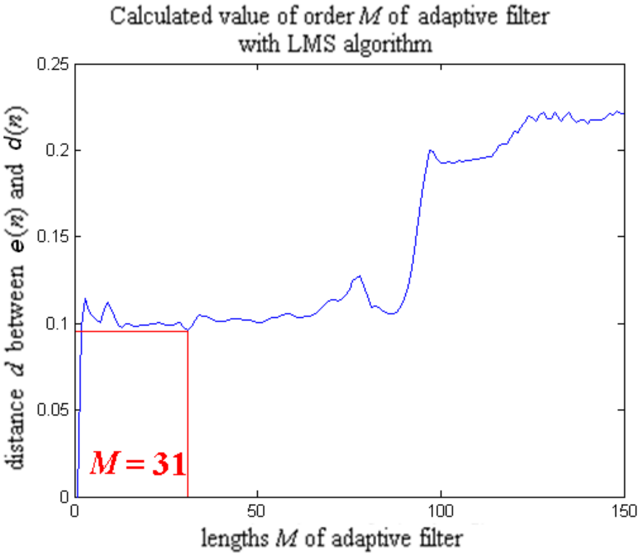


Fig. 4 Calculated values M=31 and d=9,61.10^{-2} of the LMS adaptive filter (μ=5,9.10^{-3}, SSNR=6,731(dB), M =10%).

TABLE V.
THE VALUES OF ORDER M OF ADAPTIVE FILTER AND DISTANCE d BETWEEN DESIRED SPEECH SIGNAL $d(n)$ TO LMS ADAPTIVE FILTER AND ERROR SIGNAL $e(n)$ FROM LMS ADAPTIVE FILTER CALCULATED BY WAY OF DRAFT METHOD WITH DTW CRITERION (SIMULATED IN MATLAB).

	SSNR$_a$=6,731(dB)	**SSNR$_{w1}$=18,187(dB)**	**SSNR$_{w2}$=3,119(dB)**	**SSNR$_{w3}$=–1,783(dB)**
M=10 %	μ_1=5,9.10^{-3}; ***M*=31** ***d*=9,61.10^{-2}**	μ_1=6,4.10^{-3}; ***M*=16** ***d*=1,691.10^{-1}**	μ_1=5,4.10^{-3}; ***M*=17** d=2,902.10^{-1}	μ_1=4,09.10^{-3}; ***M*=88** d=2,784.10^{-1}
M=20 %	μ_2=1,08.10^{-2}; ***M*=12** ***d*=1,124.10^{-1}**	μ_2=1,18.10^{-2}; ***M*=16** d=2,517.10^{-1}	μ_2=1.10^{-2}; ***M*=107** d=3,996.10^{-1}	μ_2=7,5.10^{-3}; ***M*=88** d=3,877.10^{-1}
M=30 %	μ_3=14,996.10^{-3}; ***M*=12** ***d*=1,249.10^{-1}**	μ_3=1,63.10^{-2}; ***M*=37** d=3,142.10^{-1}	μ_3=1,38.10^{-2}; ***M*=107** d=4,614.10^{-1}	μ_3=1,04.10^{-2}; ***M*=88** d=4,463.10^{-1}

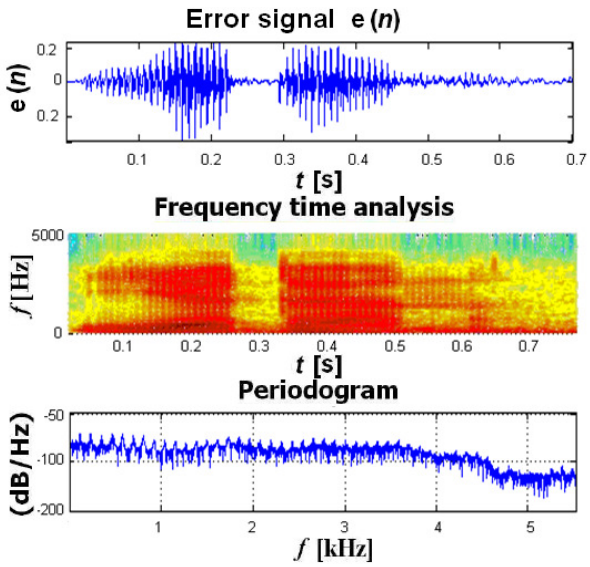


Fig. 5 Waveform, spectrogram (frequency time analysis) and periodogram of power spectral density estimate of error signal $e(n)$ (LMS adaptive filter – first iteration, M=31, μ=5,9.10^{-3}, simulated in MATLAB) [7].

In Table V can be seen, that the speech signal $e(n)$ at the output of the LMS adaptive filter **was recognized** (***d*<0,2**) from first iteration for SSNR$_a$=6,731(dB) (μ_1=5,9.10^{-3}; M=31, **M**=10%), (μ_2=1,08.10^{-2}, M=12, **M**=20%), (μ_3=14,996.10^{-3}, M=12, **M**=30%) and for SSNR$_{\text{w1}}$=18,187(dB) (μ_1=6,4.10^{-3}; M=16, **M**=10%). When the additive noise values SSNR$_w$ in speech signal were higher, the speech signal was not recognized.

VII. Using of the Draft Method with DTW criterion for LMS adaptive Noise Canceling from speech signal

B. Matlab simulation

The draft method with DTW criterion was used for the LMS adaptive noise canceling from speech signal, simulated in MATLAB in two channel structure of the adaptive filter with LMS algorithm in an application for the suppression of additive noise (Fig.6). A primary input contains desired signal $d(n)$, and an additive noise $n(n)$. A noise reference input is assumed to be available containing $n''(n)$, which is correlated with the original corrupting noise $n(n)$. As shown figure 6 the LMS adaptive filter receives the reference noise, filters it, and subtracts the result from the primary input. From the point of view of the adaptive filter, the primary input ($d(n)$+$n(n)$) acts as its desired response and the system output acts as its error. The noise canceller output $e(n)$ (Fig.7) is obtained by subtracting the filtered reference noise $n(n)$ from the primary input. Adaptive noise canceling generally performs better, than the classical approach since the noise is subtracted out rather than filtered out [8].

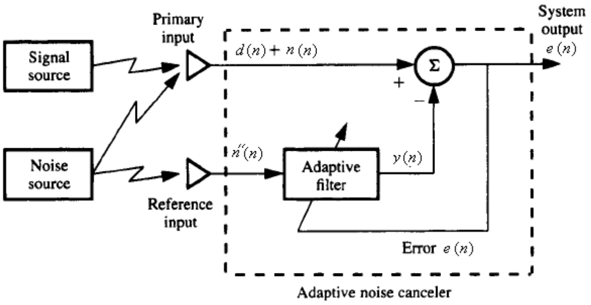


Fig. 6 Separation of signal $d(n)$ and noise $n(n)$ LMS adaptive noise-canceling approach [8].

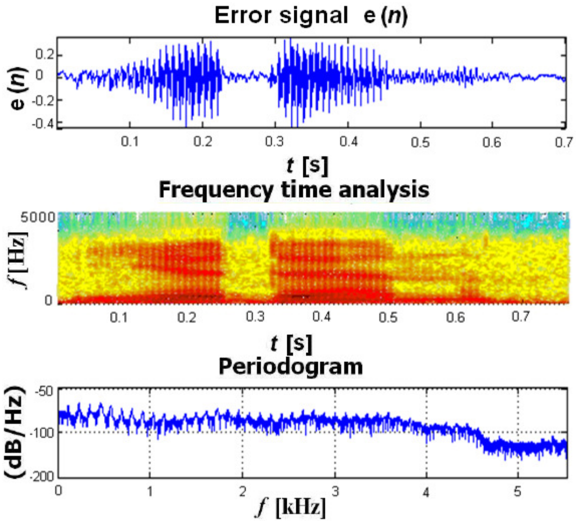


Fig. 7 Waveform, spectrogram (frequency time analysis) and periodogram of power spectral density estimate of error signal $e(n)$ (LMS adaptive noise canceller – first iteration, M=17, μ=5,9.10^{-3}, simulated in MATLAB) [7].

The draft DTW method was used for optimal settings values of the adaptive filter length M and a step size factor μ of

Table VI.
The optimal values of order M of filter and distance d between desired speech signal $d(n)$ and error signal $e(n)$ from LMS adaptive Noise canceller calculated by way of draft method with DTW criterion (simulated in MATLAB).

	SSNR$_a$=6,731(dB)	**SSNR$_{\text{w1}}$=18,187(dB)**	**SSNR$_{\text{w2}}$=3,119(dB)**	**SSNR$_{\text{w3}}$=–1,783(dB)**
M=10 %	μ_1=5,9.10^{-3}; ***M*=17** ***d*=9,9.10^{-2}**	μ_1=6,4.10^{-3}; ***M*=43** ***d*=5,421.10^{-1}**	μ_1=5,4.10^{-3}; ***M*=149** ***d*=9,142.10^{-1}**	μ_1=4,09.10^{-3}; ***M*=74** ***d*=1,473**
M =20 %	μ_2=1,08.10^{-2}; ***M*=10** ***d*=9,8.10^{-2}**	μ_2=1,18.10^{-2}; ***M*=99** ***d*=5,409.10^{-1}**	μ_2=1.10^{-2}; ***M*=103** ***d*=1,127**	μ_2=7,5.10^{-3}; ***M*=74** ***d*=1,42**
M =30 %	μ_3=14,996.10^{-3}; ***M*=7** ***d*=9,87.10^{-2}**	μ_3=1,63.10^{-2}; ***M*=99** ***d*=5,405.10^{-1}**	μ_3=1,38.10^{-2}; ***M*=103** ***d*=1,111**	μ_3=1,04.10^{-2}; ***M*=74** ***d*=1,374**

the adaptive filter with LMS algorithm in the application of the suppression of additive noise from the speech signal. Calculated optimal values – the order M of the LMS adaptive noise canceller and distance d between desired speech signal $d(n)$ (Fig.2) and error signal $e(n)$ (Fig.7) from the LMS adaptive noise canceller are calculated in Table VI. The speech error signal $e(n)$ from the output of LMS adaptive noise canceller **was recognized** (***d*<0,2**) from first iteration only for the SSNR$_a$=6,731(dB) (μ_1=5,9.10^{-3}; M=17, M =10%), (μ_2=1,08.10^{-2}, M=10, M =20%), (μ_3=14,996.10^{-3}, M=7, M =30%).

When additive noise values SSNR$_w$ in speech signal are higher, speech signal was not recognized.

C. Implementation LMS adaptive noise canceller on DSK TMS 320C6713

The draft method with DTW criterion for determining of the order M of the adaptive filter with LMS algorithm was used in an application to suppressing noise $n(n)$ from the speech signal $x(n)$ in implementation of two channel structure of LMS adaptive noise canceller on DSP (Digital Signal Processor) Starter Kit (DSK) TMS320C67113 (Fig.10) [11].

Input signal $x(n)$ (Fig.8) is composed from desired signal $d(n)$ (Fig.2) + additive noise $n(n)$. The segmental signal to noise ratio of input signal $x(n)$ (Fig.8) is SSNR=6,676(dB).

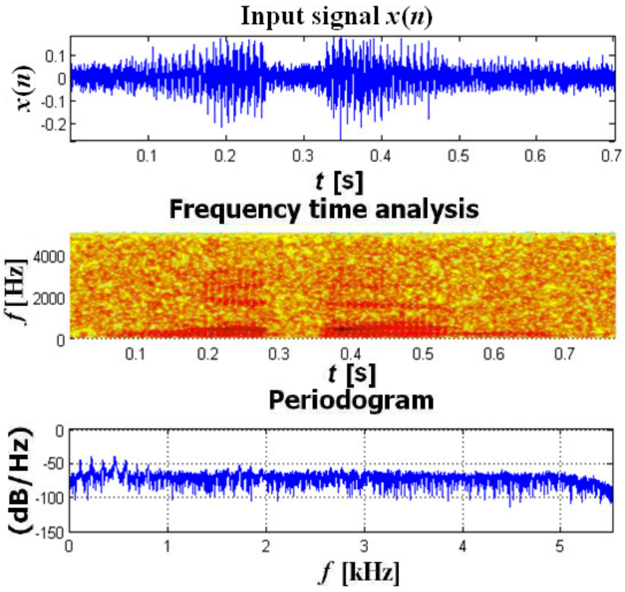


Fig. 8 Waveform, spectrogram (frequency time analysis) and periodogram of power spectral density estimate of input signal $x(n)$ (desired signal $d(n)$ + additive noise $n(n)$ - SSNR=6,676(dB)) to LMS adaptive filter in an application to suppress noise from the speech signal – first iteration, implemented on DSK TMS320C6713) [7].

Applications of the draft method with DTW criterion was carried out in several steps:

1. step - calculation of a step size parameters μ (M =10%, M =20%, M =30%) (Tab.VII) with guarantees the stability and convergence of the LMS algorithm by using of input signal $x(n)$ (Fig.8) SSNR=6,6756(dB) to LMS adaptive noise canceller (simulated in MATLAB).

2. step - the calculation values of the LMS adaptive filter order M (Tab.VII) for sets a step size parameters μ (M =10%, M =20%, M =30%) by using of input signal $x(n)$ (Fig.8) SSNR=6,6756(dB) to the LMS adaptive noise canceller (simulated in MATLAB).

In the Table VII are calculated values d between output signal $e(n)$ (Fig.9) from adaptive filter with LMS algorithm and desired signal $d(n)$ (Fig.2) for the values parameters set of the LMS adaptive filter order M.

The calculated values of distance d (Tab.VII) in MATLAB shows, that an isolated word "jeden" (Fig.9) from adaptive filter output **was recognized** **<u>*d*=0,184</u>** (***d*<0,2**), when optimal set parameters of adaptive filter with LMS algorithm are M=21, μ_1=0,103 for M =10%.

Table VII.
The calculation values of distance d, length M and a step size parameter μ of the LMS adaptive noise canceller for (M =10%, M =20%, M =30%) SSNR=6,676(dB), (simulated in MATLAB).

M	M =10%	M =20%	M =30%
μ	μ_1=0,103	μ_2=0,188	μ_3=0,26
M	***M*=21**	M=40	M=99
D	**<u>*d*=0,184</u>**	d=0,265	d=0,307

3. step - empirically was found, that the parameter μ for the LMS adaptive noise canceller, implemented on the DSK TMS320C6713 allow set only in the range μ=1.10^{-8} to μ=1.10^{-12} . The length M of the LMS adaptive noise canceller can be set only in the range M=16 to M=52.

Optimal settings values of a parameter μ and the order M was M=21 and μ=1.10^{-8} for the LMS adaptive noise canceller implemented on the DSK TMS320C6713 (Tab.VIII).

The speech error signal $e(n)$ from the output of LMS adaptive noise canceller (implemented on DSK TMS320C6713) **was recognized** (***d*<0,2**) from first iteration for the SSNR$_a$=6,676(dB) (μ=1.10^{-8}; M=21).

Table VIII.
The Calculation values of distance d for settings of length ***M*=21** and parameters μ ($x(n)$) (Fig.8) with SSNR=6,676(dB)) (LMS adaptive noise canceller was implemented on DSK TMS320C6713).

Settings of parameterμ	μ=1.10^{-12}	μ=1.10^{-10}	μ=1.10^{-8}
Calculated value of d	d=4,266.10^{-1}	d=3,475.10^{-1}	**<u>*d*=8,97.10^{-2}</u>**

VIII. Using LMS adaptive noise canceller in voice communication with control BUS system Nikobus

The draft method with DTW criterion was used for optimal settings parameters of LMS adaptive noise canceller, implemented on DSK TMS320C6713, applied in voice communications with control BUS system NIKOBUS (Fig.10). System NIKOBUS was implemented in simulation of visualization operational control of the technical features of the building through visualization software Promotic. For speech recognition in voice communication with control BUS system has been used software My Voice (Fig.10)

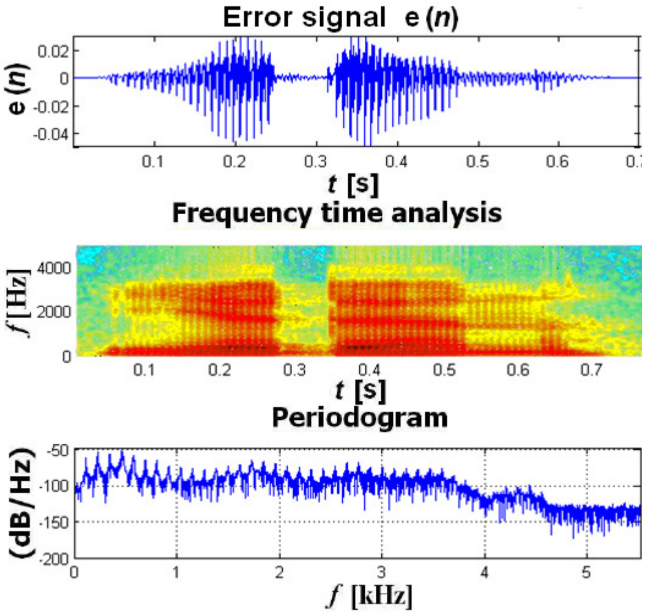


Fig. 9 Waveform, spectrogram (frequency time analysis) and periodogram of power spectral density estimate of error signal *e*(*n*) (LMS adaptive Noise Canceling from the speech signal – first iteration, *M*=21, μ=1.10^{-9}, implemented on the DSK TMS320C6713) [7].

linked with software Promotic. By using of software My Voice is done voice control of operational technical functions in the buildings.

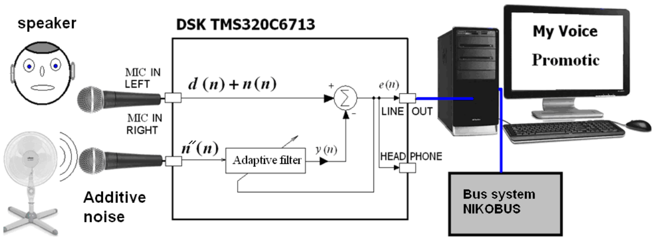


Fig. 10 Implementation of the LMS adaptive noise canceller for voice communications with control system NIKOBUS (Xcomfort) [7].

The aim of the experiment was to determine the success of the detection of selected voice commands. A microphone for capturing speech was located at a distance of about 5 cm from the mouth, according to the manufacturer's instructions. The second microphone was directed to the source of additive noise.

As source of additive noise were used the blower noise and loud radio on (radio station in D major, with classical music). The fan was placed in a distance of 25cm from the microphone. Radio speakers were placed approximately 70 cm from the microphone.

One order was one-word command "boiler" (wash-boiler) is for switching on and off the boiler.

Conditions for the experiment were the following (Fig. 11):

1. 100 x spoken command “boiler” without the LMS adaptive noise canceller:

- Measure 1 without additive noise – 99% successfully speech recognition.
- Measure 2 with additive noise – 81% successfully speech recognition.

2. 100 x spoken command “boiler” with the LMS adaptive noise canceller implemented on DSK TMS320C6713:

- Measure 3 without additive noise – 99% successfully speech recognition.
- Measurement of 4 with additive interference – 99% successfully speech recognition.

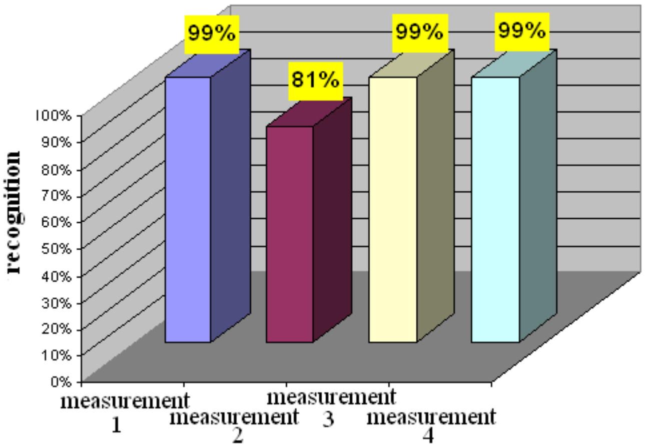


Fig. 11 Evaluation of recognition of isolated czech word - the command "boiler" (wash - boiler) by way of recognition software MyVoice.

Conclusion

In this paper was described the way of verification of the proposed method on the structure of adaptive filter with LMS algorithm in application of suppressing noise from speech signal by way of simulation in MATLAB software.

Through the use of DTW criterion was obtained a tool for determining the quality of speech signal processing using in optimal settings step size parameter μ and order *M* of the LMS adaptive filter or the LMS adaptive noise canceller

The proposed method was verified by way of the practical realization of the structure of the LMS adaptive noise canceller in the application for suppressing additive noise from speech signal by implementation on the DSK TMS320C6713 (Fig.12). This implementation was used for voice communication with control BUS system NIKOBUS for simulation controlling of operating technical functions in buildings.

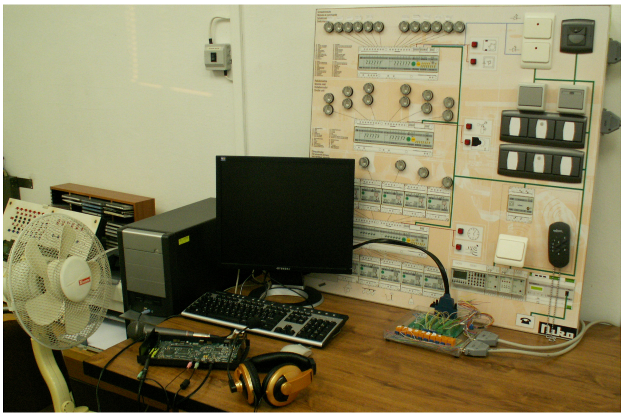


Fig. 12 Experimental workplace with DSK TMS 320C6713 and control panel with BUS system NIKOBUS.

Acknowledgments

This paper has been supported by the VŠB TU grant No. SV/4200011. Authors thanks for the support.

References

[1] B. Farhang-Borounjeny, *“Adaptive Filters, Theory and applications,”* John Wiley & Sons, Chichester, 2005, ISBN 0–471–98337–3, pp. 139-168.

[2] D. A. Poularikas, M. Z. Ramadan, *“Adaptive filtering primer with MATLAB”,* Taylor & Francis Group, 2006, ISBN 0–8493–7043–4, pp. 101-122.

[3] S. Haykin, *“Adaptive filter theory,”* PRENTICE HALL, New Jersey 2002, ISBN 0-13-090126-1, pp. 231-319.

[4] J. Uhlíř, P. Sovka, P. Pollák, V. Hanžl, R. Čmejla, *“Technologie hlasových komunikací,”* nakladatelství ČVUT Praha 2007, ISBN 978–80–01–03888–8, pp. 161-165.

[5] P. Sovka, P. Pollák, *“Vybrané metody číslicového zpracování signal,”* vydavatelství ČVUT, Praha, 2003, ISBN 80–01–02821–6, pp. 89-97.

[6] J. Černocký, *“Zpracování řečových signalů”* – studijní opora, http://www.fit.vutbr.cz/.cernocky, VUT Brno, 2006

[7] J. Vaňuš, *“Hlasová komunikace s řídícím systémem”,* (“Voice communication with control system”), Dissertation thesis, VŠB TU Ostrava, 2010

[8] B. Widrow, E. Walach, *“Adaptive Inverse Control: A Signal Processing Approach,”* Published by John Wiley & Sons, Inc., Hoboken, New Jersey 2008. ISBN 978-0-470-22609-4, pp. 59-87.

[9] J. Jan, *“Číslicová filtrace, analýza a restaurace signal,”* nakladatelství VUTIUM, Brno, 2002, ISBN 80-214-1558-4, pp. 287-308.

[10] J. Vaňuš, *“Implementation of the adaptive filter for voice communications with control systems,”* TSO 2009 Proceedings, Prešov, Slovakia, 2009 ISBN 978-80-553-0312-3, pp. 144-147.

[11] Chassaing R., Reay D.: *Digital Signal Processing and Applications with the TMS320C6713 and TMS320C6416 DSK,* John Wiley & Sons, Inc. New Jersey 2008, ISBN 978–0–470–13866–3, pp. 319-353.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Obfuscation Methods with Controlled Calculation Amounts and Table Function

YuanYu Wei
Shibaura Institute of Technology
Toyosu, Koutou-ku Tokyo , 135-8548 Japan
Email: m710101@shibaura-it.ac.jp

Kazuo Ohzeki
Shibaura Institute of Technology
Toyosu, Koutou-ku Tokyo , 135-8548 Japan
Email: ohzeki@sic.shibaura-it.ac.jp

***Abstract*—This paper describes a new obfuscation method with two techniques by which both computational complexity can be controlled and semantic obfuscation can be achieved. The computational complexity can be strictly controlled by using the technique of encryption. The computational complexity can be arbitrarily specified by the impossibility of factorization of prime numbers by length from one second to about one year. Semantic obfuscation is achieved by transforming a function into a table function. A nonlinear, arbitrary function can be incorporated into the functions, while only linear functions are used in the conventional methods. Because the explicit function form is hidden, it is thought that analysis takes time. The computational complexity technique and semantic technique can be used at the same time, and the effect of integrated obfuscation with both techniques is great.Introduction.**

I. Introduction

OBFUSCATION is used for protecting software copyright and hiding ID and passwords when you disclose the software program. Barak showed the impossibility of obfuscation with theoretical proof [1]. Using a virtual black--box, the proof was carried out for an unnatural function. The idea was restricted to such an unnatural function. It is still meaningful to consider obfuscation of software programs beyond the impossibility discussion. Moreover, obfuscation requirements have been upgraded to keep the calculation process secret in public areas. This is called the white-box scheme.Chow et al presented the white-box obfuscation scheme in [3]. document template contains different styles for appropriate text elements. Most of styles are divided into two classes: paragraph styles and symbol styles.

Pa In this paper, we will present both controlling accurate calculation amounts and creating semantic difficulty in conversion from a plain software program to an obfuscated version. For evaluating the calculation amounts of a obfuscated program, we will introduce a cipher method to obtain an accurate evaluation. As for semantic obfuscation, we will introduce a table for a function used in the program. The table function does not present the method of calculation explicitly

Monden [4] presented obfuscation methods according to loops, branches and control of orders. Hachez [5] and Myles [6] presented a wide variety of obfuscation methods including opaque predicates, computation, quality evaluation and applications to watermarking. We studied the loops and branches and then transformed these into complex functions [3]. But these methods were not suitable for reverse-engineering analyses. In this paper, we provide two different methods. One is to control calculation amounts of a software program. The other is to produce semantic obfuscation. We can use both methods in one software program because the two methods correspond to different parts of programs. The former is used for constant values in the program, while the latter is used for numerical functions such as Fourier transform etc. A basic discussion of the former method for calculation amounts has been carried out in technical reports in Japanese [7][8]. Brief results of the former method were presented in [9]. The basic idea of the latter method of semantic obfuscation was presented in [3] but it was a basic method with the name of the ROM function. ROM means 'read only memory'. This means that the function output values are all pre-calculated for a limited number of input discrete variables. Therefore the data of the function operation can be written in read-only-memory (ROM). As a result, the explicit function description disappears and only numerical values can be seen in the program. If the function is a simple linear one, reconstruction of the function formula is easy. But if the function is of second order or more, reconstruction of the function is done by numerical calculation. We can also add fake data into the ROM at the function values of unused input variables. In this paper, the basic method in [3] is implemented by giving a specific function and making part of a table.

In the following section of this paper, we present the obfuscation method by controlling calculation amounts in section 2. Then semantic obfuscation by using table function is presented in section 3.

II. Controlling calculation amounts

A. *Encrypting Essential Number (EEN)*

It is important to measure the number of calculations needed to analyze and break an obfuscated program. If we can prove the required number of calculations, which is the minimum number, it would be very useful as a component of obfuscation. The method to realize this is that using an intermediate result for a decoding key of RSA encryption, an essential number needed for processing the following calculation is encrypted by an encoding key that corresponds to the decoding key. This method is effective because it cuts the serial calculation flow. No parallel analyses are effective. Consequently, only after obtaining the intermediate value, can the program proceed to the next step by decoding the en-

This work was supported by Universal Robot Corporation.

crypted essential number. The method is called Encrypting Essential Number (EEN).

Fig. 1 shows a block diagram of the ENN software program flow. At first, an appropriate intermediate number "y" is selected. Then decoding key "Y" is decided in relation to "y". If the value "y" is a large prime number, Then Y=y is sufficient. If "y" is not a large prime number, we search for a large prime number Y and get the difference between Y and y as Bias, and decide Y=y+Bias. Then, encoding key "e", which corresponds to "Y", is obtained. To obtain the encoding key, we should decide p, q N in the same way as the RSA encrypting method in the background. Then the essential number "a" is encrypted using encoding key "e", and we get E(a). Next, the essential number - usually it is a constant in the program - is removed from the program, and instead encrypted value E(a) is written where the essential number "a" existed.

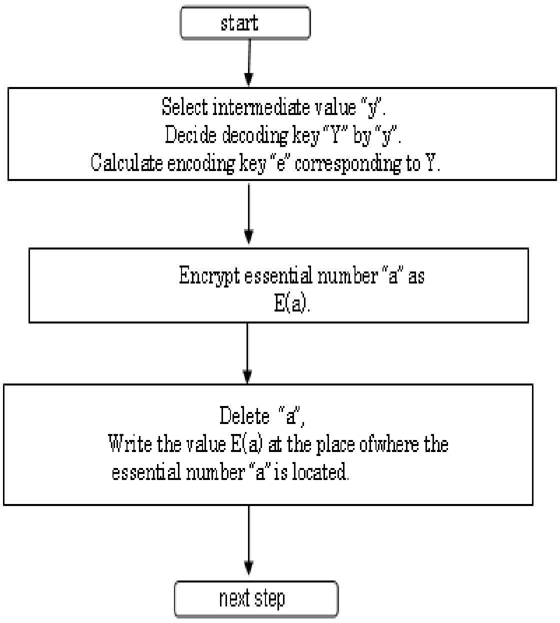
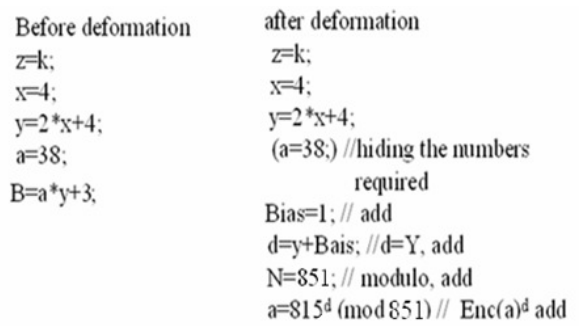

start
Select intermediate value "y".
Decide decoding key "Y" by "y".
Calculate encoding key "e" corresponding to Y.
Encrypt essential number "a" as
E(a).
Delete "a",
Write the value E(a) at the place ofwhere the
essential number "a" is located.
next step

Fig.1 EEN obfuscation block diagram

Fig. 2 shows a transformation from the original program to the proposed method of encrypting essential number (EEN). The numbers of k, z and x are given. The value "y=12" is an intermediate number, which is calculated from the values in the upper part. The value "a=38" is an essential number. To transform the original program first, a prime number 13 related to the value y=12 is selected. The difference between y and 13 is described as Bias, and "Bias=1" is written in the transformed program. Based on the prime number 13, P=38 q=23 are selected in the same manner as the RSA method. Also, the least common multiplier of p-1 and q-1 is calculated as n=396. From N=p*q=851, encrypting key Y=e=61 is obtained. The essential number a=38 is encrypted as a message. In fact, the 61-th power of "a=38" yields an encrypted value as E(a)=3861 (mod 851) =815. In an operation after transforming, the intermediate value y=12 is first obtained. Then the bias value "Bias=1" is added to the intermediate value, and decoding key "d=13" is obtained. The encrypted value "N=815" is raised to 13th power, yielding "a=38".



```
Before deformation        after deformation
z=k;                       z=k;
x=4;                       x=4;
y=2*x+4;                   y=2*x+4;
a=38;                      (a=38;) //hiding the numbers
                                   required
B=a*y+3;                   Bias=1; // add
                           d=y+Bais; //d=Y, add
                           N=851; // modulo, add
                           a=815^d (mod 851) //  Enc(a)^d add
```

Fig.2 Transform from the original program to EEN form

In this way, the hidden essential number can only be obtained by proceeding to the intermediate value position. The minimum required number of calculations is equivalent to calculating E(a)e raised to the d-th power for hidden essential number E(a)e. This kind of exponential calculation can be reduced to a known level by Montgomery multiplication etc.

B. Evaluation of Calculation Amounts

Evaluation of the obfuscation degree in a quantitative way is very difficult. Hachez said that there were several evaluation methods of software complexity, but they were qualitative methods [5]. In the proposed method in this paper, the number of calculations is specified. RSA is used by the fact that there is no way at this moment to decompose the arbitrary number into a product of prime numbers. So the evaluation of this EEN can be described by the number of multiplications. In this EEN method, encoding key "e" need not be disclosed, on the contrary, RSA does not disclose an encoding key. This means that this method is more difficult than RSA. At this moment the excess part is not considered and put forward for further study.

Table I
Examples of decoding key values

intermediate value	digit number of power r	after being raised to the r-th power	decoding key	Bias	index
3.19584	5	319584	319733	-149	i
	5.1	351542	355753	-4211	ii
	5.5	479376	479371	5	iii
	6	3195840	3195869	-29	iv

The evaluation of EEN is carried out by the following rule;

> "Amounts of calculation required to decode the encrypted value" (i)

Table 1 shows examples for several specific numbers of large decoding key values. In these examples is a real fractional value of the intermediate number. The intermediate number can be such a real and fractional number. The intermediate number value "3.19584 is shifted to several large numbers by digit number “r” by an exponential operation. For the case of r=5 at the top row, 3.19584 is raised to the 5-th power to 319584. The exponent values can be fractional as 5.1 or 5.5 as seen in the second and third rows. Next, a decoding key value is selected from an associated value of 319584. In this example, 319733 is selected as an associated value. But it is easy to understand that this value can be freely selected from the intermediate value because they are related to each other by the Bias value. The Bias value is the difference between 319584 and the decoding key value 319733.

Figs. 3(a),(b) show examples of transforming EEN methods of (i) and (ii) in Table 1. Based on these examples, we confirmed actual encoding keys derived from decoding keys, intermediate values in excecution.

For small values of decoding keys, the obtained results are shown in Table 2. A relation between an encoding key and a decoding key is,

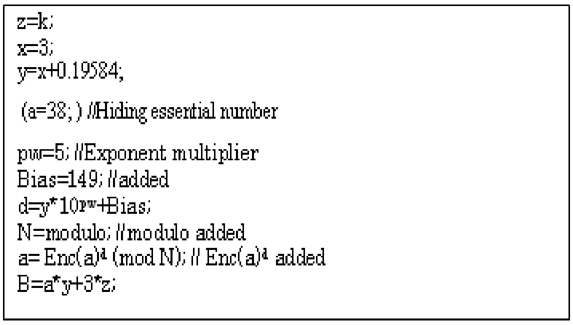
$$e \cdot d = 1 \quad (\mathrm{mod}\ N)\ . \qquad (2)$$

Obtaining an encoding key from a decoding key is done using the same process as obtaining a decoding key from an encoding key in the RSA method. In obtaining the encoding key, two prime numbers p and q are arbitrary selected. There are many choices in selecting p and q. Also after deciding p and q and the least common multiplier N of (p-1,q-1), there are many choices for selecting an encoding key.

The calculation amounts for this part of the program are decided according to the size of decoding key “d=Y” and the size of module value N. To design the system, it is sufficient to adjust these two sizes.

A detailed transform flowchart of EEN obfuscation from the original software program is shown in Fig. 4. That expression above was described for one unit of obfuscation. Within the frequency of occurrence of the intermediate numbers, requisite times of units can be arranged.

Figure 5 shows a block diagram of the digital watermark system with the type of detector that is disclosed to the public. As for the digital watermark of the image, there are various attacks, and it is difficult to insist on authenticity.

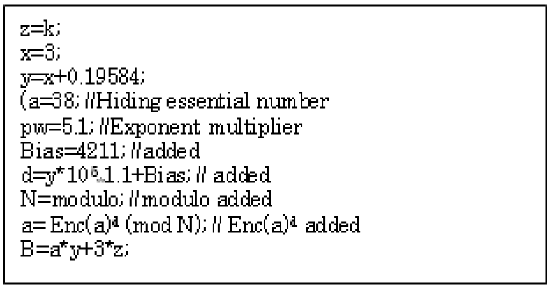


```
z=k;
x=3;
y=x+0.19584;

 (a=38; ) //Hiding essential number

pw=5; //Exponent multiplier
Bias=149; //added
d=y*10^pw+Bias;
N=modulo; //modulo added
a= Enc(a)^d (mod N); // Enc(a)^d added
B=a*y+3*z;
```

Fig. 3 (a) Detailed program sentences transformed by EEN method with index (i) in Table 1.



```
z=k;
x=3;
y=x+0.19584;
(a=38; //Hiding essential number
pw=5.1; //Exponent multiplier
Bias=4211; //added
d=y*10^5.1.1+Bias; // added
N=modulo; //modulo added
a= Enc(a)^d (mod N); // Enc(a)^d added
B=a*y+3*z;
```

Fig. 3 (b) Detailed program sentences transformed by EEN method with index (ii) in Table 1

Ref [3] proposed by the idea that credibility is sure to be improved in this method to the extent that it opens the detector to the public compared with copyright owners secretly verifying it by themselves. The execution module is obfuscated, though the detector of the digital watermark is a software program. The operation verification is made more difficult by increasing the load in the computational complexity corresponding to an attack that executes the operation analysis coping with slow down in the execution speed. Moreover, semantic obfuscation is being examined as the method replacing the complicated calculation method and the table by ROM function [3]. This uses the input value as is at used positions and uses fake functions at not-used positions. This is different from the DES calculations of Chow [2] with a table realizing as an affine-transformation.

Table Ⅱ
Calculated encoding keys and decoding keys

d	e	p	q	N	lcm
13	61	23	37	851	396
17	233	23	37	851	396
19	667	23	37	851	396
29	437	23	37	851	396
31	511	23	37	851	396
41	425	23	37	851	396
43	571	23	37	851	396
47	455	23	37	851	396

Finally, even if a minute error margin is included, it is a thing that the device such as becoming a correct value can be included because the output of the function of the input value not used is adjusted to the value of the imitation, and it will quantize the result where a discrete calculation is done.

Estimation from the calculation and the example of the execution time of the calculation frequency in obfuscation that provides for the computational complexity, and the design data of the length of the decoding key matched at the required operation time are made. Fig. 6 shows the design time when searching for the encoding key versus the length of the designed decoding key.

Searches for the prime number by the calculation, enable to assume the further estimation for the larger required length of calculation. The value of encoding key e, prime numbers of p and q are selected as each of resembling length comparable to the decoding key length. They are not neces-

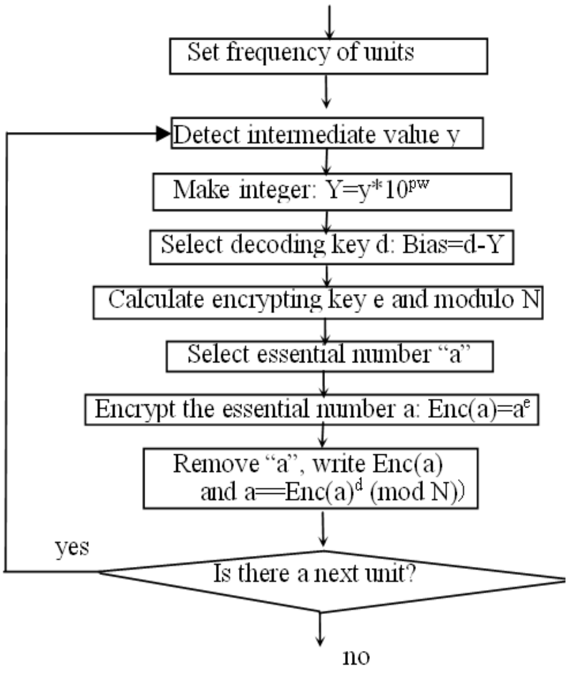

Set frequency of units
Detect intermediate value y
Make integer: Y=y*10^pw
Select decoding key d: Bias=d-Y
Calculate encrypting key e and modulo N
Select essential number "a"
Encrypt the essential number a: Enc(a)=a^e
Remove "a", write Enc(a)
and a=Enc(a)^d (mod N))
yes
Is there a next unit?
no

Fig. 4 Detailed transform flowchart of EEN obfuscation from the original software program.

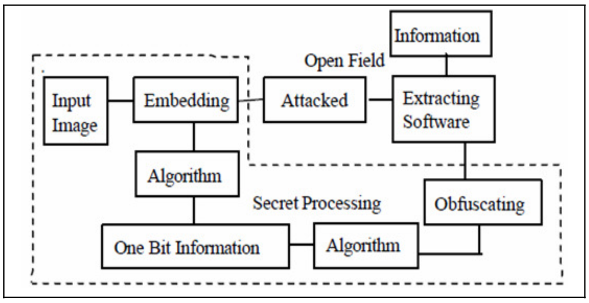


Fig. 5 Watermarking system disclosing a detector.

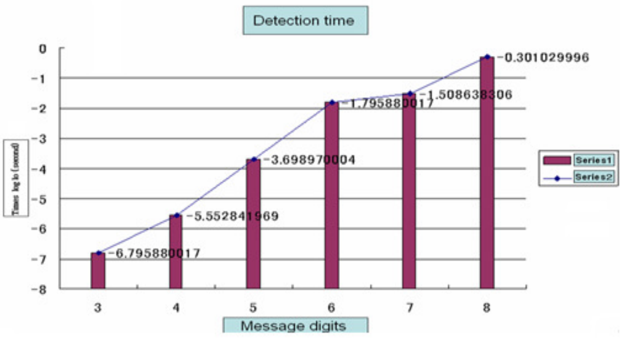


Fig. 6 Time for decoding vs. digit number of decoding key

sarily united because they are made as a design procedure for which it searches from the small number.

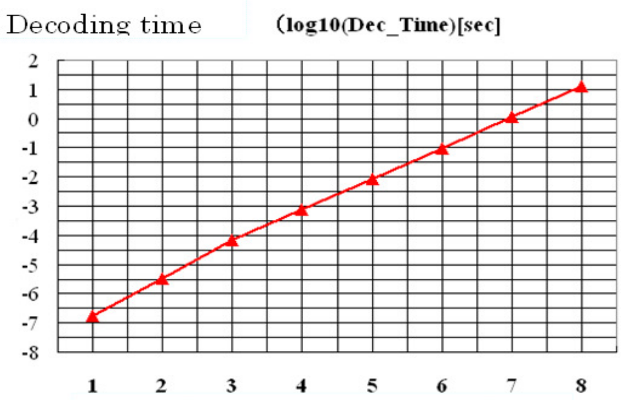


Fig.7 Decoding time vs. length of key number

Fig. 7 shows the decoding times versus the lengths of the decoding keys. The message length had the same length as the decoding key though the message length also influenced the decoding time.

The decoding time for a number composed of eight digits is about one second. The computer specification used was a Pentium 4, 3.2GHz with C language. Modulo operation was used for each multiplication and speed-up processing such as the Montgomery multiplication etc. was not used. The vertical axis was a logarithm at time, and the result for a longer digit number can be worked out by extending the graph if it is almost considered as a straight line.

When Fig. 7 was calculated for 1-3 of the horizontal axis, the digit numbers of decoding key d, encoding key e, and p, q, lcm, N, and the message were calculated as shown in Table 3. For one of four or more of the horizontal axes, almost maximum prime number was used for the digit number of p, q, and decoding key d.

It is on a regular, actual measuring evaluation because the computational complexity depends on the number of digits and the number of multiplies, divisions of modulo.

It is possible to contribute to the improvement of credibility by making the detector of the digital watermark available to the public instead of processing the secret by the closed-door method. It focused only on the increase of computational complexity as the technique for carrying out obfuscation when the detector was disclosed to the public, and the amount was estimated. The program cannot be analyzed in parallel by dividing it, which provides computational complexity of the decoding key.

In this construction, an accurate estimate can be given for computational complexity with exponent power and the modulo calculation. For a computer that uses it to experiment, about one year will cost 13-14 digits since it will cost a day and one month every 12 digits every 10-11 digits from about 10 seconds by eight digits and seconds it about ten times a digit when fast computation is not used.

III. Semantic Method by Table Function

Table 4 shows a comparison of the methods using the table function. Method (i) is a method of [3], and obfuscates the DES operation processing. The DES operation consists of permutation and xor. Because the method of permutation

Table Ⅲ

Calculating data for obtaining decoding time of an EEN unit in a detector. (The values in a bald frame really exist. The others are provisional with lengths of digit numbers

Number Of digit	1	2	3	4	5	6	7	8
p	7	97	997	9973	99991	999983	9999991	99999989
q	5	89	991	9967	99989	999979	9999979	99999971
N	35	8633	988027	99400891	9998000099	999962000357	99999640000243	9999996000109989
n	24	8448	986040	99380952	9997800120	999960000396	99999620000280	9998998800120000
d	5	79	907	9931	99961	999931	9999937	25956377
e	29	1711	97843	84560371	8325115681	552335018575	86684306711833	3826082737342313
a	4	78	900	600	6080	600800	6008000	60080000
E(a)	9	6157	306326	30121083	5305700101	953843280291	86311048530626	5357971538660731

processing is known beforehand, it is possible to decipher it by examining the concealed relation of the I/O and examining the law of permutation. Method (ii) obfuscation is for general, arbitrary functions [3]. An example of the shape of the function is shown in [3]. Moreover, the relation of I/O is not understood for those not used only from observing input and output values because of the error incorporated. Method (iii) is proposed in this paper, basically the same as case of (ii). The example of the function is concretely given, and the shape of the table function for obfuscation is considered in this paper.

To transform a linear function to a table function is not effective obfuscation because the transform unknown coefficients can be obtained from as many numbers of relations between the input and output as those of the unknown coefficients. To transform quadratic polynomial functions or general non-linear functions to table functions can be effective obfuscation because the calculation amounts become huge even though input and output relations are obtained.

In this paper, the function of the Fourier transform etc. is considered for embedding digital watermarks. Because the Fourier transform is a linear transformation, a modified version of a nonlinear transformation that provides pseudo frequency components is newly developed. Examples of the nonlinear transformations are (1) RGB-HSV color conversion, and (2) a non-linear pseudo frequency transform with exponent multiplication.

First, the original RGB-HSV color conversion:

Max=MAX(R,G,B), Min=MIN(R,G,B)

H=undefined if Max=0

$$H = \begin{cases} C_{60} \times \dfrac{G-B}{\text{Max}-\text{Min}} + C_0, & \text{if } \text{Max}=R \\ C_{60} \times \dfrac{B-R}{\text{Max}-\text{Min}} + C_{120}, & \text{if } \text{Max}=G \\ C_{60} \times \dfrac{R-G}{\text{Max}-\text{Min}} + C_{240}, & \text{if } \text{Max}=B \end{cases}$$

H=H+360 if H<0

$$S = \frac{\text{Max}-\text{Min}}{\text{Max}}$$

$$V = \text{Max}$$

H=H+360 if H<0

$$S = \frac{Max - Min}{Max}$$

$$V = Max$$

C_* are constants.

Next, a pseudo-frequency transform with exponent multiplication for the case of the least number of I/Os, which has two inputs (x,y) and two outputs (p,q) is introduced;

$$p = a \cdot x^e + b \cdot y^f$$

$$q = c \cdot x^g + d \cdot y^h$$

Here, a, b, c, d, e, f, g, and h are hidden constants. Examples of these are a=1.01, b=0.98, c=1.02, d=-0.99, e=1.01, f=1.02, g=0.997, and h=0.996, as are likewise as the Hadamard transform.

This transform converts signals into a kind of pseudo frequency region. As the exponent part is set to a number near one, the behavior of this transform resembles the Fourier transform. While, the analysis of this transform is still difficult because it is non-linear, the amount of the operation of the analysis of this transform increases remarkably compared with a linear case. For the use of a digital watermark, it need not be an accurate Fourier transform, and quantizing can use pseudo frequency components.

IV. Conclusion

This paper presents a new obfuscation method with two techniques by which both the computational complexity can be controlled and the semantic obfuscation can be achieved. Decoding keys and encrypting keys are actually calculated by using intermediate values and essential values in a software program. Among many choices of selecting encrypting keys from the decoding key, the one whose length is the same as the decoding key is selected. Actual computing times are obtained for the small sizes of encrypting key numbers. For the large sizes of them are obtained from the tendency to a extended graph for smaller sizes. The computational complexity can be arbitrarily specified by the impossibility of factorization on prime numbers by lengths from 1 second to about 1 year. Semantic obfuscation is achieved by transforming a function into a table function. A nonlinear, arbitrary function can be incorporated into the functions, while

Table Ⅳ
Comparison of table functions

method	(i) [2]	(ii) [3]	(iii)
type	permutation XOR	General	Linear Non-linear
error data	non	incorporable	incorporable
object	DES operation with Key	General functions	Matrix Fourier Transform Higher order functions

only linear functions are used in conventional methods. Two functions of the form with exponential power terms are shown. These proposed table functions effectively provide quasi-frequency components like the Fourier transform though these are nonlinear functions. These obfuscation processing methods can be applied also to many applications in signal processing such as embedding digital watermarks etc.

Acknowledgment

The authors thank Eizaburo Iwata for his dicussion to this paper.

References

[1] Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S. and Yang, K.: On the (Im)possibility of Obfuscating Programs, pp. 1-18, CRYPTO (2001).

[2] Chow, S., Eisen, P., Johnson, H., Van Oorschot, P.C.,: A White-Box DES Implementation for DRM Applications, Proceedings of ACM CCS-9 Workshop DRM pp.1-15, Nov., (2002)

[3] Kazuo Ohzeki and Cong Li, “Fingerprinting System Depending On An Anonymous Third Party Authentication Using An Assumption of Computationally Measurable Obfuscation”, IPSJ Tech Rept. CSEC-32, pp.61-66, Mar. 2006. in Japanese

[4] Akito Monden, Yoshihiro Takada and Koji Torii, “Methods for Scrambling Programs Containing Loops”, IEICE Trans D-I Vol. J80-D-I, No.7 pp.1-11 July 1997. in Japanese

[5] Gael Hachez, "A Comparative Study of Software Protection Tools Suited for E-Commerce with Contributions to Software Watermarking and Smart Cards", Thesis submitted to Belgian Catholic University, UCL, March 2003.

[6] Ginger Myles and Christian Collberg, "SoftwareWatermarking via Opaque Predicates: Implementation, Analysis, and Attacks" Proc. 7th International Conference on Electronic Commerce Research, (ICECR-7) Dallas, Texas, June 2004

[7] Engyoku Gi(YuaYu Wei), and Kazuo Ohzeki,” Evaluation of Methods and Computational Amount of Watermarking with Disclosing Obfuscated Detector”, ITE 2009 Winter Symposium 9-6, in Japanese
Engyoku Gi(YuaYu Wei), and Kazuo Ohzeki,”Computational Effort of Watermarking System with Disclosing Obfuscated Detector”, Proc. IMPS 2009 I-6-14, pp.147-148, in Japanese.

[8] Kazuo Ohzeki and Engyoku Gi (YuanYu Wei),” An Obfuscation Method for a Detector of Watermarking “, IPSJ Tech Rept. 2010-DPS142CSEC48-(30) Vol.142, pp.1-7, in Japanese.

[9] Yuanyu Wei and Kazuo Ohzeki,“A New Obfuscation Method Using Random Functions ” proceedings of Telecommumications,Networks and Systems, IADIS International conferences. P263-265,July 2010.

International Workshop on Real Time Software

International Workshop on Real Time Software will be held within the framework of the International Multiconference on Computer Science and Information Technology, and will be co-located with the XXVI Fall Meeting of Polish Information Processing Society.

Proliferation of computers interfacing with real world and controlling their environment requires careful investigation of approaches related to the specification, design, implementation, testing, and use of modern computer systems. Timing constraints, dependability, fault-tolerance, interfacing with the environment, reliability and safety constitute integral components of the software development process. Appropriate education of engineers developing such systems, working in interdisciplinary teams and in a global environment is of paramount importance.

In addition to traditional papers, we plan to organize a round table discussion forum on safety critical aspects of the education/training. We are soliciting brief one-page position papers on education and training of engineers developing dependable software intensive systems – presenting the views of academia and industry (in the submission clearly identity the "position paper" for the engineering education round table discussion). The accepted position papers will be a base for 10 minutes presentation followed by a discussion.

The workshop is planned around three main focus areas:

- Real-Time Control
- Safety, Reliability, and Dependability
- Real-Time Education

Traditional Papers topics include but are not limited to:

- Real-time system development
- Scheduling
- Safety
- Reliability
- Dependability
- Fault-tolerance
- Feedback control real-time scheduling
- Hardware-software co-design
- Standards and certification
- Control software
- Robotics and UAV
- Software development tools
- Model-based development
- Automatic code generation
- Real-time systems education
- Related engineering curricula
- Laboratory infrastructure
- Internet-based support

Computationally effective algorithms for 6DoF INS used for miniature UAVs

Jan Floder
VSB – Technical University of Ostrava, Department of Measurement and Control, 17. listopadu 15, 70833
Ostrava, the Czech republic
Email: jan.floder@vsb.cz

***Abstract*—The article aims at 6 degrees of freedom inertial navigation systems for miniature UAVs. It shows a new filter design which replaces standard solutions represented by EKF/UKF filters. The new filter is designed to significantly reduce filter complexity and processing power requirements (but keeping estimation accuracy) to be useful in small embedded systems with minimum processing power.**

I. Introduction

DEVELOPMENT in semiconductor technologies in past several decades brings technologies which used to belong to military/hi-tech domain to everyday use in variety of fields.

This includes also a development of inertial measurement related sensors like MEMS technology sensors (accelerometers, gyrometers), tiny but powerful GPS modules, small magnetometers, barometric sensors and of course fast and small processors. The sensors are used for inertial navigation systems which provide information about vehicle orientation and movement in space. These navigation systems are used for example in autonomous robots, ground, underwater and aerial vehicles.

Although all sensors mentioned above undergo rapid development accuracy is still far beyond their high-end counterparts and traditional technologies. So the sensors themselves used in inertial measurement units (abbreviation IMU) are not able to provide acceptable navigation results over a longer period of time. To solve the issue GPS measurements have to be integrated to the INS measurements to keep navigation results usable (in contrary to INS output errors, GPS measurement errors are not function of time). This solution is known as a GPS aided inertial navigation. The sensor integration and INS information calculation is usually handled by an EKF (Extended Kalman Filter) or an UKF (Unscented Kalman Filter) with adaptive gain. But these filters have drawbacks in certain situations. Major drawback is their relative complexity and requirements of powerful processors.

The article tries to describe and explain ways how replace traditional INS algorithms (represented by EKF, UKF) by algorithms which are more suitable for miniature unmanned aerial vehicles (UAVs) which have limited processing power. The article continues in development of the vector filter algorithm presented in [1].

A. INS

The INS does estimate orientation, acceleration, velocity and position of a vehicle moving anywhere on earth (or close to its surface). This is done by measurement and integration of several motion parameters. There are several types of inertial navigation but the article is aimed at a GPS aided tightly coupled inertial navigation for systems with 6 degrees of freedom of motion. Six degrees of freedom of motion means that vehicles, where the navigation is applied on, can “virtually freely” move and rotate in any direction. The only omitted parameter are huge long term centripetal accelerations because UAVs the article is aimed at do not undergo them.

The motion equations for a UAV are (presented in form prepared for inertial sensors)

$$\begin{aligned}
&\dot{\boldsymbol{q}}_{RPY\to ENU}=\boldsymbol{q}_{RPY\to ENU}*\frac{1}{2}\boldsymbol{\omega}_{RPY}\\
&\dot{\boldsymbol{v}}_{ENU}=-\boldsymbol{g}_{ENU}+\boldsymbol{C}_{RPY\to ENU}(\boldsymbol{q}_{RPY\to ENU})\cdot\boldsymbol{a}_{RPY}\\
&\dot{\boldsymbol{p}}_{ECEF}=\boldsymbol{C}_{ENU\to ECEF}(\boldsymbol{p}_{ECEF})\cdot\boldsymbol{v}_{ENU}\\
&\boldsymbol{p}_{LLA}=\begin{bmatrix}\varphi_{WGS84}\\ \lambda_{WGS84}\\ h_{WGS84}\end{bmatrix}=\boldsymbol{f}_{ECEF\to WGS84}(\boldsymbol{p}_{ECEF})
\end{aligned}\tag{1}$$

Vehicle motion parameters velocity vector $\boldsymbol{v}_{ENU}$ and position vector $\boldsymbol{p}_{ECEF}$ (and $\boldsymbol{p}_{LLA}$) are derived from vehicle acceleration vector $\boldsymbol{a}_{RPY}$ which is measured by a MEMS accelerometer (measurement contains gravity acceleration part which has to be subtracted - $\boldsymbol{g}_{ENU}$). Vehicle orientation quaternion $\boldsymbol{q}_{RPY\to ENU}$ is derived from angular rate vector $\boldsymbol{\omega}_{RPY}$ measured by a MEMS gyrometer.

The quaternion $\boldsymbol{q}_{RPY\to ENU}$ stores rotation between vehicle local frame RPY and earth local frame ENU [2]. The position is calculated in earth global frame ECEF and alternatively in LLA (longitude, latitude, altitude) coordinates (usually the WGS84 Earth model). For coordinate systems and transformations ($\boldsymbol{C}_{RPY\to ENU}$, $\boldsymbol{C}_{ENU\to ECEF}$) consult next section.

As shown above accelerometer and gyrometer are enough to calculate all parameters of the inertial navigation. But this is valid for ideal sensors only. In the real world there are many sources of errors which cripple navigation results. In the equation you can see that acceleration and angular rate are integrated to provide vehicle orientation, velocity and

978-83-60810-27-9/09/$25.00 © 2010 IEEE

position. So measurement errors (like offset) are integrated too and do rise in time. That is why magnetic field and GPS measurements have to come to stage.

B. Coordinate Systems and Transformations

The equation 1 represents INS parameters in several coordinate systems. They can be divided in 3 groups.

1. Vehicle local Cartesian coordinate system RPY (Rol-l-Pitch-Yaw) is fixed to vehicle with center in its center of gravity.

2. Earth local Cartesian coordinate system ENU/NED (East-North-Up / North-East-Down). This is a local coordinate system related to earth.

3. Earth global coordinate system ECEF, LLA (Earth-Centered-Earth-Fixed and Longitude-Latitude-Altitude). The ECEF is a Cartesian coordinate system and the LLA is a spherical (or elliptical) coordinate system.

For complete definitions consult for example [2] and [7].

Coordinate transformations transform a vector represented in one coordinate system to another and vice versa (for example $RPY \rightarrow ENU$), so they can describe vehicle orientation in space (when considering rotations only).

There are several ways to calculate/represent coordinate transformations. The best suited (for INS systems) are [2]:

- Rotation matrix $\boldsymbol{C}_{\boldsymbol{from \rightarrow to}} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$

- Quaternion $\boldsymbol{q}_{\boldsymbol{from \rightarrow to}} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T$

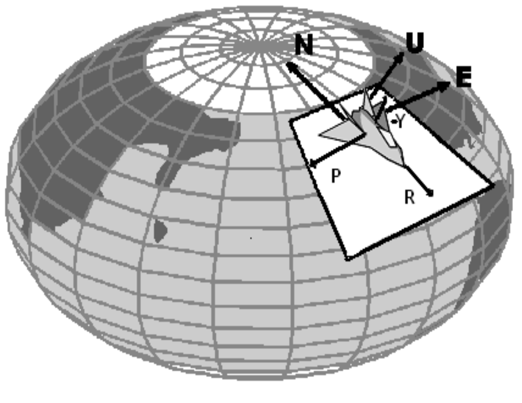


Fig 1. Illustration of coordinate systems relations (RPY, ENU, LLA).

The figure 1 illustrates RPY, ENU and LLA coordinate systems.

II. Sensors

Modern MEMS sensors are small in size (usually below 1cm2) require minimum external components (supply filter capacitor and some output low-pass single pole filters), single voltage supply and consume minimum power (below 1mA). But they also have drawbacks. As said before they are not accurate enough for classical INS.

The main sources of INS error related to MEMS sensors are additive white noise $\boldsymbol{\eta}$, temperature, random and high frequency vibration bias drift and offset $\boldsymbol{b}$, scaling error $\boldsymbol{s}$ and cross axis sensitivity $\boldsymbol{c}$.

A. MEMS Accelerometers

Bearing in mind the equation 1, accelerometer output is transformed from vehicle coordinates (RPY) to local earth coordinates (ENU) and integrated to obtain vehicle velocity (and position). Ignoring possible attitude estimation error $\delta \boldsymbol{q}_{\boldsymbol{RPY \rightarrow ENU}}$ (see section *INS Algorithms*) there are three types of accelerometer errors which corrupt velocity estimation in time. It is a drift & offset, mis-scaling and cross-axis sensitivity. According to practical measurements (3 axis ADXL335) these errors may lead up to 0.3 $\mathrm{m \cdot s^{-2}}$ (each axis) ignoring high frequency vibrations drift - in helicopter-like UAVs, where powerful engines are needed to create enough lift, there is a danger of strong vibration bias drift due to motor vibrations (close to resonant frequency of sensor's mechanical parts). The only way to avoid it is a mechanical damping.

B. MEMS Gyrometers

Gyrometer outputs are integrated to obtain attitude information from angular rate. The most important error, which is usually estimated and compensated by the INS filter, is a bias drift $\boldsymbol{b}_{\boldsymbol{g}}$. For modern gyrometers overall drift does not usually exceed 0.5 deg/s (MLX90609) but initial bias may differ on every run so it is worthy to estimate it. There may also be a danger of a high frequency vibration drift but as long as resonant frequencies of gyrometer mechanical parts are higher compared to accelerometers the danger is smaller.

C. Magnetometers

Modern magneto-resistive or magneto-inductive magnetometers are size and power consumption friendly too. But in their case another type of errors is most important. It is a hard iron distortion and a soft iron distortion. The distortions are caused by close sources of magnetic field (like DC motor magnets) or magnetic materials (chassis, shielding, etc.). Without compensation these errors lead in overall error that may exceed even 20 degrees. But ways how to compensate these errors are not subject of the article.

D. Miniature GPS Modules

GPS modules just receive and process navigation data from geostationary satellites (and in some cases additional navigation data from ground stations too). They are also small (especially models with integrated antennas) and may weight even less than 20 grams. They output actual position and velocity information with accuracy of several meters (RMS) for position and tenths of meters per second for 2D/3D velocity (later GPS chipsets, like µBlox-5, provide 3D velocity estimations).

III. INS Algorithms

At first a standard tightly coupled INS based on the multiplicative EKF (MEKF) design is presented. It was chosen

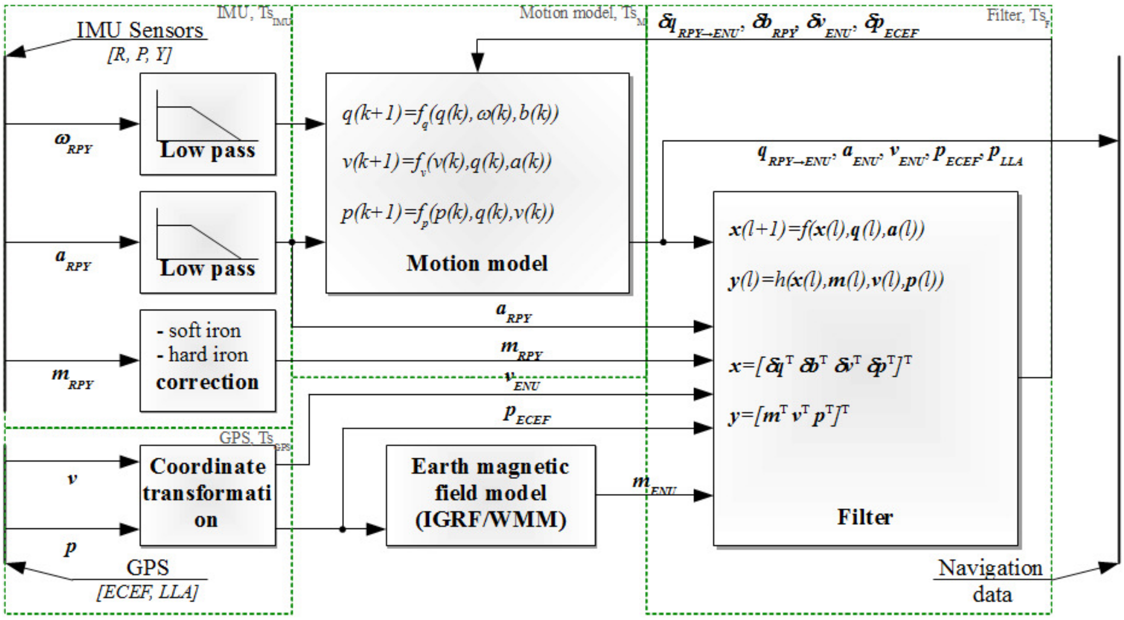

IMU, Ts_IMU
IMU Sensors
[R, P, Y]
ω_RPY
a_RPY
m_RPY
Low pass
Low pass
- soft iron
- hard iron
correction
Motion model, Ts_M
q(k+1)=f_q(q(k), ω(k), b(k))
v(k+1)=f_v(v(k),q(k),a(k))
p(k+1)=f_p(p(k),q(k),v(k))
Motion model
δq_RPY→ENU, δb_RPY, δv_ENU, δp_ECEF
Filter, Ts_F
q_RPY→ENU, a_ENU, v_ENU, p_ECEF, p_LLA
x(l+1)=f(x(l),q(l),a(l))
y(l)=h(x(l),m(l),v(l),p(l))
x=[δq^T δb^T δv^T δp^T]^T
y=[m^T v^T p^T]^T
Filter
a_RPY
m_RPY
v_ENU
p_ECEF
m_ENU
GPS, Ts_GPS
v
p
Coordinate transformation
Earth magnetic field model (IGRF/WMM)
GPS
[ECEF, LLA]
Navigation data

Fig 2. Tightly coupled GPS aided INS (based on error model) block diagram.

due to its elegance (relatively "linear" model) and "simplicity" (sparse matrices reduce number of arithmetic operations needed for calculations) which enables it to be used in non-powerful processors unlike some other solutions (like for example GP-UKF). The MEKF also represents a standard navigation solution which is used as a reference to the navigation algorithm proposed in this article. The figure 2. shows a tightly coupled INS diagram.

A. Error Model

The MEKF INS filter, which can be found for example at [4] is based on an error model where the motion model itself is calculated alone and only model errors are subject of estimation (by the MEKF). So using accelerometer and gyrometer measurement and equation 1 the INS estimates vehicle orientation in space $\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU}$, velocity in local earth coordinates $\tilde{\boldsymbol{v}}_{ENU}$ and vehicle position on earth $\tilde{\boldsymbol{p}}_{ECEF}$ (and in more common longitude, latitude and altitude coordinate system $\tilde{\boldsymbol{p}}_{LLA}$). Because there are errors which corrupt INS outputs, estimation errors are calculated according to [4]

$$\begin{aligned}
&\delta\dot{\boldsymbol{q}}_{RPY\rightarrow ENU}=\frac{1}{2}\delta\boldsymbol{b}_{RPY}\times\delta\boldsymbol{q}_{RPY\rightarrow ENU}-\frac{1}{2}\delta\boldsymbol{b}_{RPY}-\frac{1}{2}\boldsymbol{\eta}_g\\
&\delta\dot{\boldsymbol{b}}_{RPY}=\boldsymbol{\eta}_b\\
&\delta\dot{\boldsymbol{v}}_{ENU}=-2\boldsymbol{C}(\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU})\boldsymbol{a}_{RPY}\times\delta\boldsymbol{q}_{RPY\rightarrow ENU}-\boldsymbol{C}(\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU})\boldsymbol{\eta}_a\\
&\delta\dot{\boldsymbol{p}}_{ECEF}=\boldsymbol{C}_{ENU\rightarrow ECEF}(\boldsymbol{p}_{ECEF})\delta\boldsymbol{v}_{ENU}
\end{aligned}\tag{2}$$

The filter is constructed of 4 state vectors (12 state variables) which serve to estimate: attitude error vector $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$ which contains only vector part of the quaternion because for small errors the scalar part is ≈ 1 – see [4], gyrometer bias drift $\delta\boldsymbol{b}_{RPY}$, velocity estimation error $\delta\boldsymbol{v}_{ENU}$ and position estimation error $\delta\boldsymbol{p}_{ECEF}$. $\boldsymbol{\eta}_g$ for a gyrometer, $\boldsymbol{\eta}_b$ for a gyrometer bias drift, $\boldsymbol{\eta}_a$ for an accelerometer represent sensor white noise acting on the error model. The filter is updated (every step) by a magnetometer measurement $\boldsymbol{m}_{RPY}$ [4] which is compared to expected magnetic filed vector $\boldsymbol{m}_{ENU}$ for given location,

$$|\boldsymbol{m}_{RPY}|=\boldsymbol{A}(\delta\boldsymbol{q}_{RPY\rightarrow ENU})\cdot\left(\boldsymbol{C}(\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU})^T\cdot|\boldsymbol{m}_{ENU}|\right)\tag{3}$$

$$\boldsymbol{A}(\delta\boldsymbol{q})=\begin{bmatrix}1 & 2\delta\boldsymbol{q}[3] & -2\delta\boldsymbol{q}[2]\\ -2\delta\boldsymbol{q}[3] & 1 & 2\delta\boldsymbol{q}[1]\\ 2\delta\boldsymbol{q}[2] & -2\delta\boldsymbol{q}[1] & 1\end{bmatrix}\tag{4}$$

When a GPS measurement is available the filter is also updated by a measured velocity vector in ENU frame $\boldsymbol{v}_{ENU}$ and a position vector $\boldsymbol{p}_{ECEF}$ in ECEF frame.

$$\begin{bmatrix}\boldsymbol{v}_{ENU}\\ \boldsymbol{p}_{ECEF}\end{bmatrix}=\begin{bmatrix}\tilde{\boldsymbol{v}}_{ENU}+\delta\boldsymbol{v}_{ENU}\\ \tilde{\boldsymbol{p}}_{ECEF}+\delta\boldsymbol{p}_{ECEF}\end{bmatrix}\tag{5}$$

These estimated INS solution errors have to be used to correct INS data (the INS data are calculated using equation 1). The correction is

$$\begin{aligned}
&\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU}=\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU}*\delta\boldsymbol{q}_{RPY\rightarrow ENU}\rightarrow\delta\boldsymbol{q}_{RPY\rightarrow ENU}=0\\
&\tilde{\boldsymbol{v}}_{ENU}=\tilde{\boldsymbol{v}}_{ENU}+\delta\boldsymbol{v}_{ENU}\rightarrow\delta\boldsymbol{v}_{ENU}=0\\
&\tilde{\boldsymbol{p}}_{ECEF}=\tilde{\boldsymbol{p}}_{ECEF}+\delta\boldsymbol{p}_{ECEF}\rightarrow\delta\boldsymbol{p}_{ECEF}=0
\end{aligned}\tag{6}$$

The errors can be transformed from the filter to INS parameters estimations and the errors themselves are set to zero [4], [5] after (every) filter step. The $\tilde{\boldsymbol{q}}*\delta\boldsymbol{q}$ may be rewritten as (derived from [4])

$$\tilde{\boldsymbol{q}}*\delta\boldsymbol{q}=\begin{bmatrix}-\tilde{\boldsymbol{q}}[2] & -\tilde{\boldsymbol{q}}[3] & -\tilde{\boldsymbol{q}}[4]\\ \tilde{\boldsymbol{q}}[1] & -\tilde{\boldsymbol{q}}[4] & \tilde{\boldsymbol{q}}[3]\\ \tilde{\boldsymbol{q}}[4] & \tilde{\boldsymbol{q}}[1] & -\tilde{\boldsymbol{q}}[2]\\ -\tilde{\boldsymbol{q}}[3] & \tilde{\boldsymbol{q}}[2] & \tilde{\boldsymbol{q}}[1]\end{bmatrix}\delta\boldsymbol{q}+\tilde{\boldsymbol{q}}\tag{7}$$

According to the general rule in quaternion calculations the quaternion $\tilde{\boldsymbol{q}}$ should be normalized always after several calculations to avoid accumulation of floating-point round-off errors (valid especially for 32-bit floating-point).

B. (M)EKF/UKF

The error model merits of relatively simple and quite "linear" equations, and sparse matrices when using EKF. But besides this there is one more advantage. Because motion model and error estimation are separated they can be calculated with different sampling periods (bearing in mind S-N-K theorem) – motion model can be calculated for example 100 times per second and the error estimation just 20 times per second. This does reduce arithmetic operations requirements even more. The MEKF is an EKF variant where the attitude estimation error quaternion is multiplied (not added) to the model [5] when every other filter aspect remains the same.

The EKF and UKF (adaptive versions) equations are well known and can be found among literature. The UKF has one disadvantage and this is that it cannot take advantage of spare matrices. Some example INS solution can be found for example at [6] or [9].

C. Vector Filter

First of all complete explanation of the vector filter is too long to fit in the article so it is described only briefly. Some basic principles are described in [1]. The EKF/UKF filter is a working solution proven by time and experience (used over 40 years) but for purposes of miniature UAVs there are some drawbacks. Embedded systems for these UAVs tend to be very cheap, small and light and all processing (basic signal processing, navigation and control) is usually done by a single System on Chip processor which should also have minimum power consumption. This means that processing power for the navigation is significantly reduced. Besides this hurdle EKF/UKF configureability is somewhat limited (it is easy to alter $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrices but impossible to adjust convergence mechanisms freely).

This is the main reason why the vector filter was developed. First stage was development of an IMU filter to provide orientation estimation for a loosely coupled INS filter (described in [1]). This article describes enhancement to a tightly coupled and mixed solution INS. The filter has these main features:

a) It is based on vectors rather than single variables. It means that $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$, $\delta\boldsymbol{b}_{RPY}$, $\delta\boldsymbol{v}_{ENU}$, $\delta\boldsymbol{p}_{ECEF}$ are taken directly as 4 variables (not 12 as in the EKF).

b) Equation searching for solution of $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$, $\delta\boldsymbol{b}_{RPY}$, $\delta\boldsymbol{v}_{ENU}$, $\delta\boldsymbol{p}_{ECEF}$ to approach measured values is solved directly. It does not matter if it is solved analytically, iteratively, using gradient or any other way. The goal is to find a proximate solution to reach measured values and distance between estimation and measurement.

c) The criteria to reach measured values can be/are defined for any vector independently and can by customized any way and also changed during run.

d) Almost all criteria of the filter are accessible and can be changed on run.

1) Equation solution

The filter is based on the same principles as presented in previous work in [1]. Firstly the bias drift $\delta\boldsymbol{b}_{RPY}$ is excluded from equation 2 because it can be estimated from $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$ directly:

$$\delta\boldsymbol{b}_{RPY}(k+1)=\delta\boldsymbol{b}_{RPY}(k)-\alpha_b(k)\delta\boldsymbol{q}_{RPY\rightarrow ENU}(k) \tag{8}$$

(α_b = correction size, see section *Correction Size Calculations*) and angular rate is corrected outside the error equation

$$\boldsymbol{\omega}_{RPY,compensated}=\boldsymbol{\omega}_{RPY}-\delta\boldsymbol{b}_{RPY} \tag{9}$$

So the modified equation 2 now consists of $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$, $\delta\boldsymbol{v}_{ENU}$ and $\delta\boldsymbol{p}_{ECEF}$ only. There are more ways how to find proximate solution of this modified equation. The solution presented here solves $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$ and $\delta\boldsymbol{v}_{ENU}$, $\delta\boldsymbol{p}_{ECEF}$ separately (because it involves minimum of arithmetic operations).

a) $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$ by quaternion math

Unlike in the [1], now the main vector is the measured magnetic field vector $\boldsymbol{m}_{RPY}$ and measured velocity $\boldsymbol{v}_{ENU}$ is the second vector. $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$ is calculated (using superposition in each axis (Roll, Pitch and Yaw)) as

```
for $k=1$ to 3
  $c_{rot,\pm}=\left(\boldsymbol{C}_{rot}(\boldsymbol{\omega}_{search}(k))|\tilde{\boldsymbol{m}}_{RPY}|\right)\cdot|\boldsymbol{m}_{RPY}|$
  if $c_{rot,+}\geq c_{rot,-}$ then $s=1$
   else $s=-1$
  $\boldsymbol{\omega}_{corr}[k]=s\left[acos(|\tilde{\boldsymbol{m}}_{RPY}|\cdot|\boldsymbol{m}_{RPY}|)-acos\left(max(c_{rot,+},c_{rot,-})\right)\right]$
end
$\delta\boldsymbol{q}_{RPY\rightarrow ENU}=-\frac{1}{2}\alpha_{\delta q,m}|\boldsymbol{\omega}_{corr}|$
```

$$\boldsymbol{C}_{rot}\left(\begin{bmatrix}\alpha\\ \beta\\ \gamma\end{bmatrix}\right)=\begin{bmatrix}\cos(\beta)\cos(\gamma) & -\cos(\beta)\sin(\gamma) & \sin(\beta)\\ \sin(\alpha)\sin(\beta)\cos(\gamma)+\cos(\alpha)\sin(\gamma) & \cos(\alpha)\cos(\gamma)-\sin(\alpha)\sin(\beta)\sin(\gamma) & -\sin(\alpha)\cos(\beta)\\ \sin(\alpha)\sin(\gamma)-\cos(\alpha)\sin(\beta)\cos(\gamma) & \cos(\alpha)\sin(\beta)\sin(\gamma)+\sin(\alpha)\cos(\gamma) & \cos(\alpha)\cos(\beta)\end{bmatrix} \tag{10}$$

Where $\boldsymbol{C}_{rot}$ is a rotation matrix [10]

$\boldsymbol{\omega}_{search}$ is a search rotation vector – for separate rotations in R, P and Y axis respectively (for k =1, 2, 3)

$\boldsymbol{\omega}_{corr}$ is a correction vector

$\alpha_{\delta q,m}$ is a magnetic vector correction size (see section *Correction Size Calculations*)

In case a new GPS measurement is available, $\delta\boldsymbol{q}_{RPY\rightarrow ENU}$ is updated using second pair of vectors (the velocity vector $\boldsymbol{v}_{ENU}$ and the model velocity estimation $\tilde{\boldsymbol{v}}_{ENU}$). These two vectors are not compared directly because there is a singularity when the vehicle is not moving (→ zero velocity vector). So the vectors are taken with added gravity and subtracted previous estimation

$$\begin{aligned}\tilde{\boldsymbol{v}}_{ENU,mod}&=\tilde{\boldsymbol{v}}_{ENU}(t)+\Delta t_{GPS}\cdot\boldsymbol{g}_{ENU}-\tilde{\boldsymbol{v}}_{ENU}(t_{previousGPS})\\ \boldsymbol{v}_{ENU,mod}&=\boldsymbol{v}_{ENU}(t)+\Delta t_{GPS}\cdot\boldsymbol{g}_{ENU}-\tilde{\boldsymbol{v}}_{ENU}(t_{previousGPS})\end{aligned} \tag{11}$$

The estimated (modified) velocity $\tilde{\boldsymbol{v}}_{ENU,mod}$ vector is rotated around measured magnetic field vector $\boldsymbol{m}_{\boldsymbol{RPY}}$ by angle $\alpha_{\delta q,v}$ towards the measured (modified) velocity vector $\boldsymbol{v}_{ENU,mod}$. The sequence of calculations is:

$$\begin{aligned}
&\delta \boldsymbol{q}_{ENU\rightarrow RPY}=\boldsymbol{C}\left(\tilde{\boldsymbol{q}}_{RPY\rightarrow ENU}\right)\delta \boldsymbol{q}_{RPY\rightarrow ENU}\\
&\boldsymbol{q}_{corr,+}=\left[\cos\left(\frac{\alpha_{\delta q,v}}{2}\right)\quad \sin\left(\frac{\alpha_{\delta q,v}}{2}\right)\left(\boldsymbol{C}\left(\boldsymbol{q}_{RPY\rightarrow ENU} * \delta \boldsymbol{q}_{RPY\rightarrow ENU}\right)\left|\boldsymbol{m}_{RPY}\right|\right)^T\right]^T\\
&\boldsymbol{q}_{corr,-}=\left[\cos\left(\frac{\alpha_{\delta q,v}}{2}\right)\quad -\sin\left(\frac{\alpha_{\delta q,v}}{2}\right)\left(\boldsymbol{C}\left(\boldsymbol{q}_{RPY\rightarrow ENU} * \delta \boldsymbol{q}_{RPY\rightarrow ENU}\right)\left|\boldsymbol{m}_{RPY}\right|\right)^T\right]^T\\
&\boldsymbol{v}_{rot,+}=\boldsymbol{C}\left(\boldsymbol{q}_{corr,+} * \delta \boldsymbol{q}_{ENU\rightarrow RPY}\right)\left|\tilde{\boldsymbol{v}}_{ENU,mod}\right|\\
&\boldsymbol{v}_{rot,-}=\boldsymbol{C}\left(\boldsymbol{q}_{corr,-} * \delta \boldsymbol{q}_{ENU\rightarrow RPY}\right)\left|\tilde{\boldsymbol{v}}_{ENU,mod}\right|\\
&\text{if } \left|\boldsymbol{v}_{ENU,mod}\right|\cdot\boldsymbol{v}_{rot,+}\geq\left|\boldsymbol{v}_{ENU,mod}\right|\cdot\boldsymbol{v}_{rot,-}\\
&\quad\text{then } \delta \boldsymbol{q}_{ENU}=\boldsymbol{q}_{corr,+} * \delta \boldsymbol{q}_{ENU\rightarrow RPY}\\
&\quad\text{else } \delta \boldsymbol{q}_{ENU}=\boldsymbol{q}_{corr,-} * \delta \boldsymbol{q}_{ENU\rightarrow RPY}\\
&\delta \boldsymbol{q}_{RPY\rightarrow ENU}=\left[\delta \boldsymbol{q}_{ENU}[1]\quad \boldsymbol{C}\left(\boldsymbol{q}_{RPY\rightarrow ENU}\right)^T \delta \boldsymbol{q}_{ENU}[2:4]\right]^T[2:4]
\end{aligned}\tag{12}$$

As mentioned before $\delta \boldsymbol{q}_{RPY\rightarrow ENU}$ is a vector part of quaternion and is used for small correction angels (→ scalar part of quaternion ≈ 1). If the angles are huge (usually than ten degrees per step) transform $\delta \boldsymbol{q}_{RPY\rightarrow ENU}$ to quaternion and calculate with pure quaternions.

This method is used because it is very fast (less than 500 arithmetic operations per step) and is easy to implement.

b) $\delta \boldsymbol{q}_{RPY\rightarrow ENU}$ by gradient

Usage of a minimum-squared-error (MSE) criterion and gradient (or Hessian) calculations is another way to find $\boldsymbol{q}$ or $\delta\boldsymbol{q}$ for two sets of vectors. The approach how estimate quaternion from two sets of vectors using Gauss-Newton or Newton method was presented by Marins *et al* in [3] (article available on web). But this method is not as efficient as the previous because it involves several matrix calculations (including inversion).

c) $\delta \boldsymbol{v}_{ENU}$ and $\delta \boldsymbol{p}_{ECEF}$ calculations

Velocity and position correction direction and size calculation is much easier. It involves subtraction and distance calculation of two vectors only (no transformations are necessary).

$$\begin{aligned}
\delta \boldsymbol{v}_{ENU}&=\alpha_v\left(\boldsymbol{v}_{ENU}-\tilde{\boldsymbol{v}}_{ENU}\right)\\
\delta \boldsymbol{p}_{ECEF}&=\alpha_p\left|\boldsymbol{p}_{ECEF}-\tilde{\boldsymbol{p}}_{ECEF}\right|
\end{aligned}\tag{13}$$

The formula shows that $\delta \boldsymbol{v}_{ENU}$ is calculated regarding distance between measurement and estimation while $\delta \boldsymbol{p}_{ECEF}$ not. This is because different correction calculations were used for each case.

2) Solution "distance" calculations

"Distance" calculations (estimation to measurement) serve to find a distance between estimated and measured values. They are much simpler than direction (or gradient) calculations and involve vector angle and vector distance calculations. General equations are as follows

$$\begin{aligned}
d_{vec1}&=acos\left(\frac{\boldsymbol{v}_1\cdot\boldsymbol{v}_2}{\|\boldsymbol{v}_1\|\|\boldsymbol{v}_2\|}\right)\\
d_{vec2}&=\|\boldsymbol{v}_1-\boldsymbol{v}_2\|
\end{aligned}\tag{14}$$

d_{vec1} represents vector angle calculations for the magnetic vector and modified velocity vector. d_{vec2} is used for velocity and position error distance calculations.

3) Correction Size Calculations

The first section of the filter gives only information which direction correct to (to the measurements) and about distance between estimation and measurement. So the aim of the second section is to find a correction size to keep estimation on INS sensors (accelerometer and gyrometer) in short term but also lock them to GPS and magnetometer measurements in long term. The function can be defined any way according to user specification. This part is under development but even in an early stage it provides good results (see *Results* section). In this case it was defined as

1. Bias drift $\delta \boldsymbol{b}_{RPY}$ accumulation. The $\alpha_b=0.1$ except initialization time ($\alpha_b=0$). This setting says that the bias accumulation is directly dependent on the attitude correction. But this simple accumulation algorithm works as proven in the [1].

2. $\delta \boldsymbol{q}_{RPY\rightarrow ENU}$. In this case situation is a bit complicated. $\delta \boldsymbol{q}_{RPY\rightarrow ENU}$ consists of two parts and it is the magnetic vector $\boldsymbol{m}$ and velocity vector $\boldsymbol{v}$. The are treated separately to calculate $\alpha_{\delta q,m}$ and $\alpha_{\delta q,v}$ to form final $\delta\boldsymbol{q}$. These gain coefficients are calculated according to the following general formulas

$$\alpha_{vec}(k+1)=\alpha_{vec}(k)-T_S\, d_{vec}(k)\,\beta_{fall}\tag{15}$$

or

$$\alpha_{vec}(k+1)=\alpha_{vec}(k)+T_S\, d_{vec}(k)\,\beta_{rise}\tag{16}$$

The equations show that actual gain depends on previous gain α_{vec} and actual measurement to estimation error d_{vec}. Decision whether to accumulate (equation 12) or de-accumulate (equation 13) gain depends on sensor noise, actual gain size, error between measurement and estimation and time. β_{rise} and β_{fall} are constants which determine speed of accumulation and de-accumulation (set respecting sensor drifts/offsets). The gain α_{vec} is limited by an upper border to prevent filter from divergence/oscillations (which may occur in EKF/UKF designs when coefficient are set improperly).

3. $\delta \boldsymbol{v}_{ENU}$. For most situations $\alpha_{v_{ENU}}=0.5$ is sufficient enough. This converges to measurement for all cases.

4. $\delta\boldsymbol{p}$. The calculations are based on the same principles (equations 15-16) as $\delta \boldsymbol{q}_{RPY\rightarrow ENU}$ size calculations.

As can be seen from equations above it is very easy and simple to calculate correction sizes but the algorithm still is very efficient (as proven by simulations shown in chapter IV).

4) Higher Level of Control

Because the vector filter calculates correction direction and correction size there can be some superior control which can bypass lower levels of the filter and set corrections , this is sufficient for example during initialization when the filter settles its estimation within one or several consecutive steps (figure 5., *Higher Level of Control* sub-section). Or when the GPS looses fix for several seconds - in this case all the correction may by zeroed for that time or the INS can be switched to a loosely coupled version to keep at least attitude estimation valid.

D. Mixed Solution

Another possible filter design is to leave some parameters for the smaller EKF and the rest calculate by the vector filter. In this case $\delta \boldsymbol{q}_{RPY \rightarrow ENU}$ and $\delta \boldsymbol{v}_{ENU}$ are estimated by the 6 state MEKF and $\delta \boldsymbol{b}_{RPY}$ and $\delta \boldsymbol{p}_{ECEF}$ are estimated by the vector filter (using rules from above). This reduces filter complexity dramatically but still the MEKF is used for crucial parts of the INS.

IV. Results

In this chapter results and filter comparisons are presented. The comparisons are based on simulated data which are set to represent sensor errors during UAV flights as close as possible or with worse parameters for stress tests. Simulated data were chosen because in this case true values are known and many variations can be tested. The sensor errors were set to exceed measured errors of real sensors during flight conditions to test filter limits.

Three INS solutions are compared. The first one is a "full" MEKF – it means 12 state EKF, the second one is a combined solution with 6 state EKF estimating attitude and velocity error with the rest being estimated by a vector filter and as the last one is the full vector filter (without higher level of control).

A. Simulated Data

The flight itself was simulated by these parameters: UAV accelerations were simulated by a quarters of sinusoids (simulating acceleration of an vehicle with some momentum) with random amplitudes which did not exceed ±4 m·s^{-2} and random periods which did not exceed 10 seconds. Resulting speed and position were calculated by integrations. Vehicle rotation was simulated similar way with maximum angular rates of ±250 deg·s^{-1} and random periods with less then 10 seconds. In fact these conditions cannot be reached during a flight because the UAV has limits in freedoms of motion (cannot rotate upside down etc.) and length ans size of acceleration but the values are suitable to test filter limits.

B. Sensor Errors

Every sensor included additive white noise, bias drift, offset, mis-scaling and quantization error (simulating ADCs). Besides these errors, magnetometer included uncompensated hard and soft iron distortion and GPS included measurement delay. Here is a short list of (proximate) parameters.

Table I.
Sensor parameters (errors)

White noise for each axis (std. deviation)	**offset/bias/other for each axis**	**Mis-scaling for each axis**	**quantization**
accelerometer [m·s^{-2}]			
< 0.3	< 0.25 , constant	< 1% , constant	0.01
Gyrometer [deg·s^{-1}]			
2.7	< 1.5 , changing	< 2% , constant	0.15
Magnetometer (46μT vector size) [μT]			
0.2	< 0.5 , constant	< 10% , constant	0.2
GPS ENU velocity [m·s^{-1}]			
0.3	0.25 sec measurement delay	-	0.01
GPS ECEF position [m]			
2.5	< 3 changing	-	0.01

As can be seen from previous data simulated sensors contain significant errors. For example overall error between real and measured magnetometer data may reach up to 4 degrees. Such a big error was set to simulate partially uncompensated iron distortions. With GPS velocity, 2D velocity error does exceed 0.7 m·s^{-1}.

C. Simulation Results for Extreme Conditions

The sub-chapter presents results of simulations constructed with parameters mentioned in table I. In table there are presented the most important estimation INS outputs: gyro bias drift estimation, attitude estimation and velocity estimation. Position estimation is not considered (as being estimated from velocity). Estimation errors during filter initialization are excluded. The simulation is considered to be a "worst condition" scenario. It means that in a real flight INS should perform considerably better (the maximum error should be less than maximum error presented in table II). So the table has to be treated that way.

Table II. shows that the vector filter performs a little bit better in this case but it is caused by more proper selection of parameters compared to the EKF. The 12 state EKF does not perform better compared to the mixed 6 state EKF solution. So the table shows that the vector filter can replace the EKF without any problem. Large attitude estimation error is caused by uncompensated hard and soft iron distortions which are in this case equal to 4 deg (compare with table III. where the distortion error is roughly 1.5 deg).

For a short comparison, table III shows attitude estimation error for the 6 state EKF (mixed filter) and the vector filter for more common conditions. Maximum magnetometer error is set to 1.5 deg and the rest of errors are lowered to a half (except quantization errors and GPS velocity and position errors which were kept the same).

TABLE II.
INS ESTIMATION ERRORS FOR EXTREME CONDITIONS

Filter	average	standard deviation	maximum
Attitude estimation error [deg]			
12 state EKF	3.59	1.64	10.66
Mixed Filter	3.49	1.56	10.23
Vector Filter	3.21	1.33	9.81
Bias drift vector estimation error [deg/s]			
12 state EKF	0.49	0.2	1.4
Mixed Filter	0.41	0.16	1
Vector Filter	0.41	0.17	0.95
velocity vector estimation error [m/s]			
12 state EKF	0.38	0.16	1.16
Mixed Filter	0.37	0.16	1.15
Vector Filter	0.36	0.16	1.16

TABLE III.
INS ESTIMATION ERRORS FOR MORE COMMON CONDITIONS

Filter	average	standard deviation	maximum
Attitude estimation error [deg]			
Mixed Filter	1.53	0.98	5.86
Vector Filter	1.35	0.81	5.25

D. Simulation Results - Figures

Here are several examples of INS filter estimations. These figures were taken for common conditions simulation.

Figure shows that both two filters tend to overact a bit. This is caused by $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrices set for extreme conditions for the EKF and rise β_{rise} and fall β_{fall} coefficients for the vector filter.

E. Simulation Results – Filter Complexity Calculations

As shown before both the MEKF and the vector filter can reach about the same estimation results and the MEKF can be replaced by the vector filter. The table below tries to show the main merit of the vector filter and this is its complexity (when it comes to number of arithmetic operations per filter step). The data are calculated for for filter frequency of 100 Hz and floating point operations are taken in single precision (one double precision FP operation was approximated as two single precision operations). Model calculations and ECEF to LLA transformations were omitted. Results were rounded.

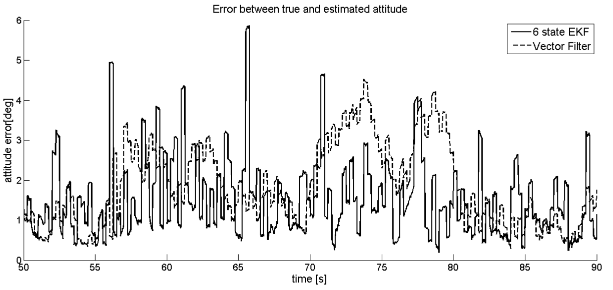


Fig 3. Example plot of attitude error for the mixed filter (6 state EKF) and the vector filter

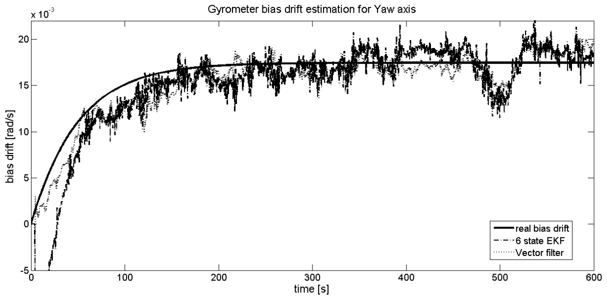


Fig 4. Example plot of a bias drift estimation

Matrix optimizations were calculated for the 12 state EKF only to show the huge difference between unoptimized and optimized calculations with sparse matrices. The table shows that the 12 state EKF needs really huge processing power and without optimizations and lowering filter frequency it is not suitable for miniature INS systems. The mixed solution and the vector filter are much better suited. The vector filter needs less than 4% of processing power compared to the (unoptimized) 12 state EKF and 20% of the processing power compared to the mixed solution.

F. Vector Filter - Higher Level of Control

The above simulations were done using the vector filter without higher level of parameter control – in this case the filter behaves like the corresponding EKF. As the vector filter calculates direction and distance of convergence to the measured values the higher level of control can take these data and for example during filter initialization it can estimate correct filter outputs in one or several consecutive steps.

V. CONCLUSION

The article shows that a standard tightly coupled GPS aided inertial navigation can be easily replaced by much more efficient solutions keeping the estimation as accurate as for

TABLE IV.
FILTER COMPLEXITY

Filter	FLOPS	FLOPS, matrix optimizations	FLOPS, full optimizations + 25 Hz filter frequency
12 state EKF	1896000	640000	190000
Mixed Filter	348000	-	-
Vector Filter	70000	-	-

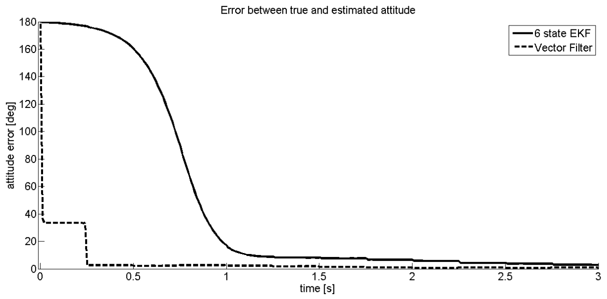


Fig 5. Example plot of a higher level control functionality (attitude estimation error). Compared vector filter and mixed solution.

the standard solutions. This may be done either by the hybrid solution where the KF is used to estimate some parameters while other are estimated by the custom filter. Or it may by done by the custom filter only. The main benefit of this approach is significant reduction in complexity (see table IV) and better system control-ability. Besides this benefit the nature of the vector filter implementation enables it to switch from tightly to loosely coupled system and back without any impact on the system which is vital in cases when GPS measurement is not present for longer period of time and INS attitude estimation would become corrupted due to present sensor errors.

These merits are vital especially in miniature embedded system where their price, size and power consumption is very limited (the system is being developed for the miniature UAV where the whole embedded system weights less than 100 grams and consumption is less than 1Watt (including wireless communication)).

References

[1] Floder J, Flying Object Control – Inertial Measurement Unit for an Embedded System, *PDES 2009*, pp. 108-113.

[2] Mohinder S. Grewal - Lawrence R. Weill - Angus P. Andrews, *Global Positioning Systems, Inertial Navigation and Integration*, ISBN 0-471-35023-X.

[3] Marins J. et al, An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors, *International Conference on Intelligent Robots and Systems,* 2001. *http://npsnet.org/~zyda/pubs/IROS2001.pdf*

[4] Bijker J., Steyn W., Kalman filter configurations for a low-cost loosely integrated inertial navigation system on an airship. *Elsevier Journal*, 2008. *http://linkinghub.elsevier.com/retrieve/pii/S0967066108000816*

[5] Markley F. L., Multiplicative vs. Additive Filtering for Spacecraft Attitude Determination, *NASA's Goddard Space Flight Center*.

[6] Xiaoying Kong., INS algorithm using quaternion model for low cost IMU, *Elsevier Journal,* 2004. *http://linkinghub.elsevier.com/retrieve/pii/S0921889004000119*

[7] http://en.wikipedia.org/wiki/Geographic_coordinate_system

[8] Wendel J., Trommer G., Tightly coupled GPS/INS integration for missile applications, *Elsevier Journal*, 2004. *http://linkinghub.elsevier.com/retrieve/pii/S1270963804000793*

[9] Ko J., Klein D., Fox D., Haehnel D., GP-UKF: Unscented Kalman Filters with Gaussian Process Prediction and Observation Models *http://citeseerx.ist.psu.edu*

[10] Addison Wesley Publishing Company, The Official Guide to Learning OpenGL, Version 1.1, pp. 458-60

Supervisory control and real-time constraints

Wojciech Grega

Department of Automatics, AGH University of Science and Technology, 30-059 Kraków, Al. Mickiewicza 30, Poland
wgr@ia.agh.edu.pl

***Abstract*—OPC (OLE for Process Control) protocol was developed as a solution which fulfills requirements of open data integration architecture for industrial control. OPC standard is not primarily intended for feedback control or communication with high-bandwidth hard real-time requirements. Adding OPC to a process could influence the dynamics of the control loop and could cause problems in controller design and implementation. The experiments presented in this paper have shown that OPC if properly configured, is capable of providing a loop time shorter than the time constants of many industrial processes.**

I. Introduction

THERE are three major trends observed in contemporary industrial control systems:

- distributing and decentralizing structures of automation,
- increasing integration of communication through all the levels of the control systems supported by application of wired and wireless networks,
- growing demand for application of IT standards.

Most of the contemporary industrial automation systems adopts multilevel, vertical control architecture [1], [2]. The hierarchical, multilevel approach to industrial process automation is a well accepted method able to cope with complexity of the process and multiple criteria of operation. The structure from Fig. 1 represents a functional decomposition, as it is based on defining functionally different control objectives for vertically dependent layers.

Logically, a typical complex, industrial control system is structured into three levels (Fig. 1), which are: the direct (device) control level, supervisory level and process optimization and management level. Basic objective of the direct (device) control level is to maintain the process states in a safe operations mode according to the prescribed set values. Device controller level provides interface to the hardware, either as separate modules or as microprocessors incorporated in the equipment to be controlled. A number of embedded control nodes and Programmable Logical Controllers (PLC) are used as the front-ends to take the control tasks.

The supervisory level comprises workstations and industrial PCs providing, the high-level program support, database support, graphic man-machine interface, alarming and general management of computing resources. The partial objective of this level is to keep set values of device controllers or process inputs on prescribed values in order to meet demands of product quality or other economical constraints.

The supervisory level often interacts with optimization level. The objective of this level is to calculate the best values of process inputs. Its structure can be very complex and include such modules as identification module, simulation module or optimization module [2].

Usually direct control level operates with an intervention frequency significantly higher (i.e. sampling period is shorter) than the intervention frequency of the supervising level. In direct control sampling intervals can be in the range of miliseconds, whereas sampling interval of the process optimization level can be in the range of seconds or even minutes. This effect is justified by slow-varying disturbances shifting optimal high- level set-points, when compared to the device level feedback control dynamics.

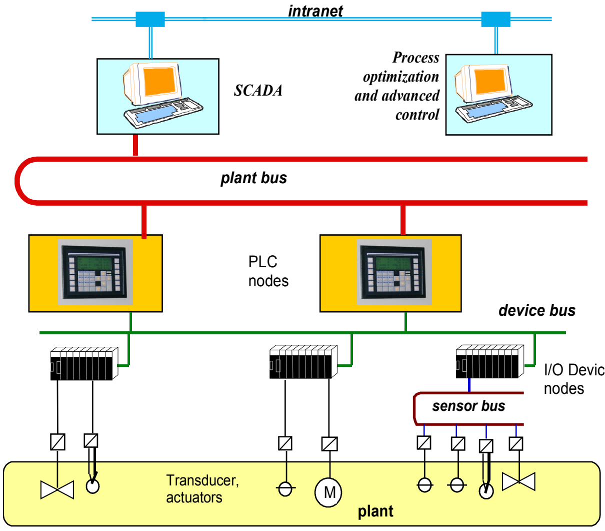


Fig.1 Multilevel structure of the industrial automation system

The ability to easily integrate information from control systems and „plant floor" measurements with supervisory and optimisation levels is a critical issue. It is very difficult to share data between various devices and software manufactured by different vendors without a common industrial communication protocol and integration standard to facilitate interoperability [3], [4].

This work was supported by the European Regional Development Fund Grant no. UDA-POIG.01.03.01-12-171/08/00.

The key is an open and effective communication and integration architecture concentrating on data access, not on the types of data. Ethernet comes as a solution to the data transmission problems mentioned above, while the OPC (OLE for Process Control) protocol was developed as a solution which fulfills requirements of open data integration architecture [5], [6].

In determining the capability for improving the plant operation through control methods implementation, it is important to understand where the delays occur and how these individual delay components contribute to the end-to-end communication. As a first step in this delay analysis, the impact of the network is assessed.

It is well known that the introduction of data transmission networks into the feedback loop in many cases violates conventional control theories assumptions such as non-delayed or evenly spaced sampling sensing and actuation. The computer network is characterized by its maximal throughput. This parameter limits the amount of data that can be sent within a time unit. Network-induced delays may vary depending on the network load and medium access protocol. Generally, networked control often introduces some additional temporal non-determinism. For distributed control systems variable queuing delays, transmission delays, transport layer ACK delays and the lost of data, leads to the deterioration of the quality of control [7]. Various methodologies were proposed for compensation of such the effects [8].

When using networks for control, it is often important to assess determinism as a QoS parameter. For example, evaluating whether end-to-end message communication times can be predicted exactly or approximately, and whether these times is bounded. Total end-to-end delay between network nodes is the sum of pre-processing time (microprocessor), waiting time (network protocol – MAC), transmission time (data rate & length), post-processing time (microprocessor).

Very little was said about the impact of OPC data processing on the dynamics of a control loop [9]. The overhead associated with OPC is included in pre (post) processing times and might be significant and variable for some configurations of this interface. Most of this delay is due to the software implementation of the OPC protocol, as OPC was not intended for hard real-time applications.

J. T. Parrott et al. from the University of Michigan [16] had investigated more deeply the delays induced by the application layer in three types of communication:

- UDP, User Datagram Protocol, a simple transport layer protocol,
- OPC, OLE for Process Control, an open application level communication protocol,
- VPN, virtual private networks, which create secure “tunnels” to transfer data between networks.

The Authors concluded that application layers delays are more important than network delays.

The main focus of this paper is in the usage of OPC in real-time control loop. It has already been mentioned that OPC has been designed to help with the interoperability problems inherent in a market with different propriety devices, protocols and industrial network standards. The question that needs to be asked is, which costs does this interoperability incur on the system in terms of performance? In order to achieve these goals a number of experiments was planned and performed. The results are given in the following sections.

II. Networks

Current communication systems for automation implements different protocols. This is a substantial disadvantage, leading to necessity of using vendor-specific hardware and software components, which increase installation and maintenance costs. Moreover, presently used fieldbus technologies make vertical communication across all levels of the automation systems difficult. Gateways need to be used to establish connections between different kinds of fieldbus systems used in the lower level, and Ethernet used in the upper levels.

Ethernet provides unified data formats and reduces the complexity of installation and maintenance, which, together with the substantial increase of the transmission rates and communication reliability over the last years, results in its popularity in the area of industrial communications.

Ethernet, as defined in IEEE 802.3, is non-deterministic and thus, is unsuitable for hard real-time applications. The media access control protocol, CSMA/CD with its backoff algorithm, prevents the network from supporting hard real-time communication due to its random delays and potential transmission failures. In real-time systems, delays and irregularities in data transmission can very severely affect the system operation. Therefore, various techniques and communication protocol modifications are employed, in order to eliminate or minimise the undesired effects.

To employ Ethernet in industrial environment, its deterministic operation must first be assured, which can be accomplished in several ways. Coexistence of real-time and non-real time traffic on the same network infrastructure remains the main problem. This conflict can be resolved in several ways, by [10]:

- embedding fieldbus or application protocol on TCP/IP – the fieldbus protocol is tunneled over Ethernet, and full openness for “office” traffic is maintained,
- using special Data Link layer for real-time devices – special protocol is used on the second OSI Layer, implemented in every device. The real-time cycle is divided into slots, one of which is opened for regular TCP/IP traffic, but the bandwidth available is heavily limited down,
- using application protocol on TCP/IP, direct MAC addressing with prioritization for real-time, and hardware switching for fast real-time,
- maintaining real-time on TCP/IP is achieved by prioritized messaging and time synchronization – the

synchronized devices assign higher priority and timestamp real-time messages,

All the specific techniques allow a considerable improvement in terms of determinism. The desire to incorporate a real-time element into this popular single-network solution has led to the development of different real-time Industrial Ethernet solutions, called Real-time Ethernet, as PROFINET, EtherCAT, Ethernet/IP [11], [12] and many more. The conditions for the industrial use of Ethernet are described by international standard IEC (*International Electrotechnical Commission*) IEC 61 784-2 *Real Time Ethernet,* Fig.2.

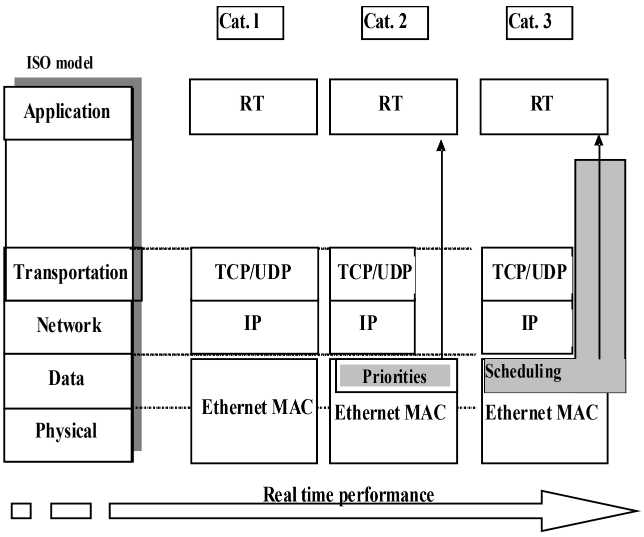


Fig.2 Classification of industrial Ethernet (IEC 61 784-2)

III. OPC

OPC [6] is a widely accepted open industrial communication standard that enables the exchange of data without any proprietary restrictions between multi-vendor devices and control software (horizontal integration) as well as between different software applications (vertical integration, Fig.3). Currently, most of plant-level control systems can be configured to be the OPC servers for supervisory control levels of the factory.

OPC is based on the Microsoft DCOM specification. Although OPC actually consists of many different data exchange specifications, its most commonly used form is Data Access (OPC DA), which supports both client-server and publisher-subscriber data exchange models. OPC DA deals only with current process data - not historical data or alarms.

An OPC client can be connected to several OPC servers provided by different vendors. Important feature of OPC is that the client is able to connect the OPC server which is located in other computer in the network. This is possible because of DCOM (Distributed COM) the COM standard extension which enables client running on one computer to create instances and invoke methods of servers on another computer within the network.

The OPC standard defines numerous ways to communicate between the server and its clients to satisfy different OPC applications. The client can specify that some operations should be performed on "cache" or "device". If "device" is chosen, operation directly on the physical device will be requested. If "cache" is selected data will be read from server's internal memory where server keeps a copy of device data received during the last OPC refresh cycle.

Both reading and writing can be done synchronously or asynchronously:

- OPC synchronous functions run to completion before returning. It means that the OPC client program execution is stopped till operation is over.
- OPC asynchronous functions use callback mechanism to inform the requested OPC operation is over. The client program execution is not stopped while waiting for OPC operation completed.

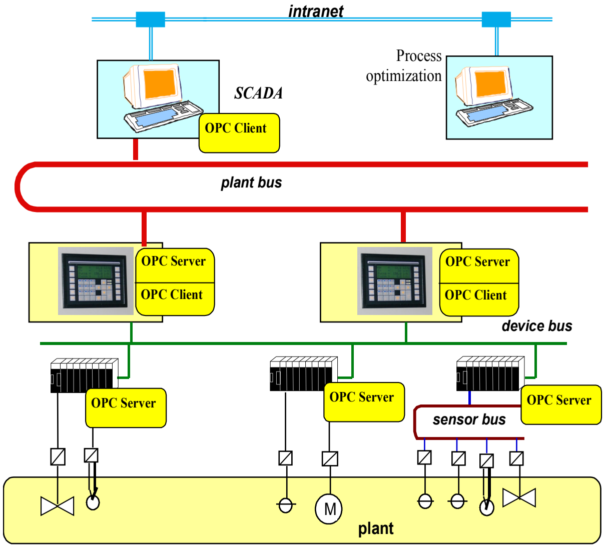


Fig. 3 Vertical integration with OPC

The OPC standard was not primarily intended for feedback control or communication with high-bandwidth, hard real-time requirements. However, many of I/O devices require real-time capabilities that are specific to the hardware and essential to proper operation of the high-level SCADA application. Therefore, one of the most important issues for communication between devices is to the assure proper configuration and synchronization.

IV. Experiments

The control algorithm must be designed according to a specification that provides the required performance when changes occur at the input or in the disturbances. For distributed control systems variable delays are considered to be the most important disturbances. For longer delays observed in the control loop it is much difficult to ensure the stability of the system within the response specification. Another factor that affects the operation of the control system is the variability of the delays, which again cannot be too large without making the system unstable.

The control system considered in this paper has two data processing modules (Fig. 4): the OPC server and OPC client. The main objective of these experiments was to investigate how fast the OPC server responds to a client request and what is uncertainty associated with such a data exchange? The configuration shown in Fig. 4 describes idea of the experiments. Analogue Process Simulator (PS) have been used as a model of real controlled process. The device can simulate a third order linear process with different time constants.

The PS has two current inputs (4..20 mA) and one current output. The process control value was calculated in real-time in Simulink and send to PLC (Siemens) during the next OPC server cycle. In this configuration PLC was only a bridge between PC and Process Simulator. During each PLC controller program cycle process variables are read from PS and control variables are send to PS. The OPC server makes the variables available for OPC clients operating in the upper layer.

The Matlab OPC Toolbox™ was applied at the upper layer. It is a set of functions which make able to connect with an external OPC server from Matlab/Simulink programming environment. For the experiments described in in this paper only Simulink part of the OPC Toolbox was used. The PID controller block of Simulink was running using *pseudo real-time simulation* mode.

To configure the reading or writing block of the OPC client it is also important to decide the method how data will be captured from the OPC server. We had selected one of two options:

Synchronous read – MATLAB stops simulation and wait for the server to return data from a read request before continuing processing. The data returned by the server can come from the server's *cache*, or the server read values directly from the *device* that the server item refers to. Reading from the device can be slower but data are more time accurate.

Asynchronous read – MATLAB client sends a request to read data from the server. Ones the request is accepted it continue processing without waiting to receive values from server. Server will use callback to inform MATLAB about new data occurrence. MATLAB will handle that event as soon as it is able to perform that task.

With each OPC data read time stamp value is associated. Theoretically, distinction in time between successive time stamps should be equal to OPC item refresh rate (T_{ITEM}). In reality it varies depending synchronization method and CPU load.

Default server's cycle time T_{SERVER} for Siemens PLC was set as 00ms. OPC item refresh rate is specified in OPC Client and should be multiplication of OPC Server cycle time. OPC performance was calculated as sum of absolute errors between two successive OPC item read time stamps and OPC item refresh rate (T_{ITEM}) and divided by number of time stamps. The results of this calculation will presents the average time deviation of OPC item reading:

$$OPC_{Perform}=\frac{1}{N}\sum_{i=1}^{N-1}|(TimeStamp_{i+1}\text{-}TimeStamp_i-T_{ITEM})|$$

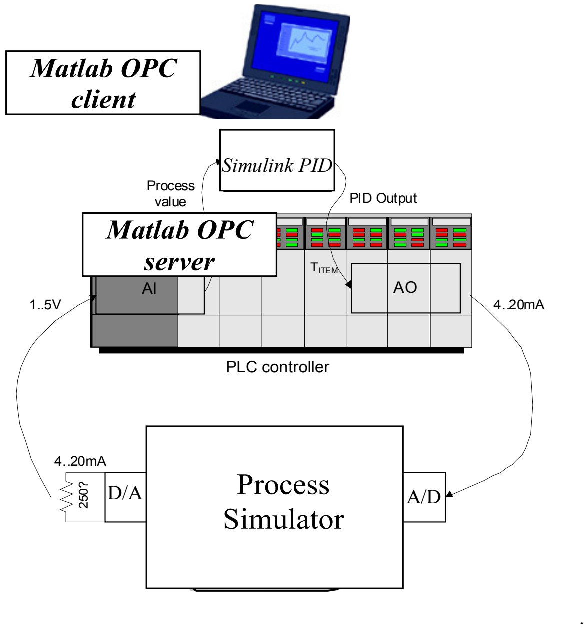


Fig. 4 Laboratory control system

The results of the above experiment are summarized in Table 1.

TABLE 1. OPC PERFORMANCE [13]

Configuration	OPC item rate T_{ITEM}=100ms, OPC server rate T_{SERVER}=100ms
	OPC performance
Synchronous cache read, synchronous write	8,72
Asynchronous cache read, asynchronous write	162,80
Asynchronous device read, synchronous write	162,39
Synchronous cache read, asynchronous write	13,31
Synchronous device read, synchronous write	6,99
Synchronous device read, asynchronous write	6,39

It can be seen for the delay of 100ms between readings, that the values of jitter (i.e. average of the deviation from the data mean delay) obtained for synchronous read are much lower compared to asynchronous operations. As it can be seen in Fig.5d the OPC client very often needs 500ms or even 700ms to asynchronously read data from the PLC (see Fig.6b). Notice, that while waiting for this event processing the controller is running and obsolete data are used for control. If one looks in details into asynchronous time stamps vector, he will recognize also, that several items have the same time stamp.

The lower jitter values obtained for synchronous mode indicate the significance of proper configuration of OPC interface.

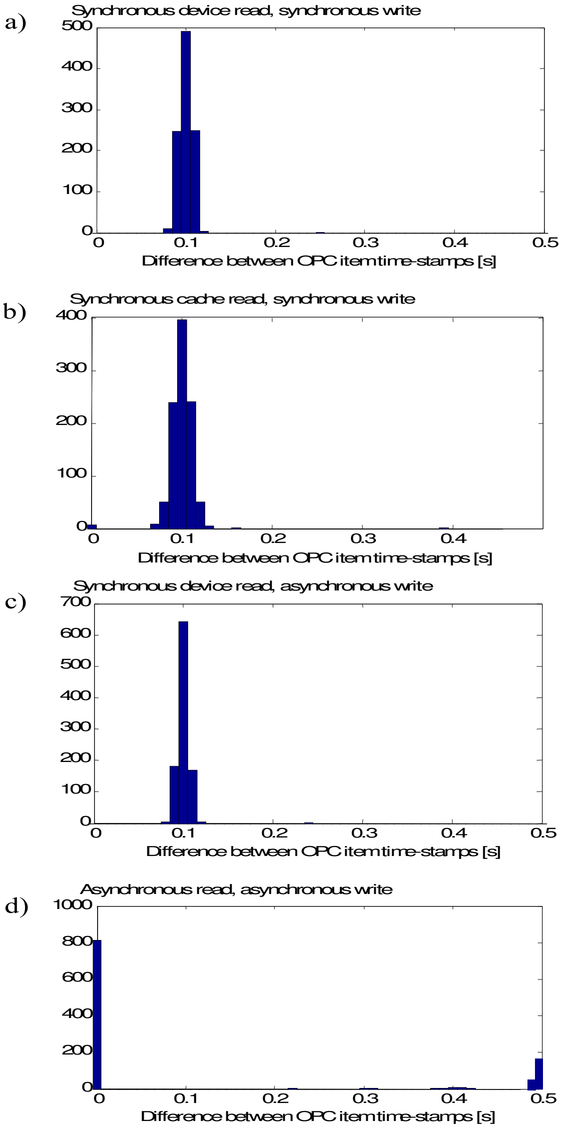


Fig.5 OPC jitter

V. Influence of network traffic

In this part impact of network traffic on data delivery was tested. PLC and PC with MATLAB were connected each other via hub and additionally, three other computers were connected to the hub. Between additional computers artificial TCP traffic was generated. OPC Server refresh rate and OPC item refresh rate were set to $T_{ITEM}=$ $=T_{SERVER}=100$ms. One of the tests was performed using wireless (WiFi) connection. The results are presented in Table 2.

Digital control systems are based on periodic operations. A shortest sampling period is limited by OPC server refreshment rate. It must guarantee acceptable control performance. The network communication delays and jitters will be added to the OPC delays and jitters, creating total delays and uncertainty in the control loop. General conclusion after this experiment is, that low network traffic is not influencing very much the time accuracy of data delivery. In this case OPC jitter is more significant.

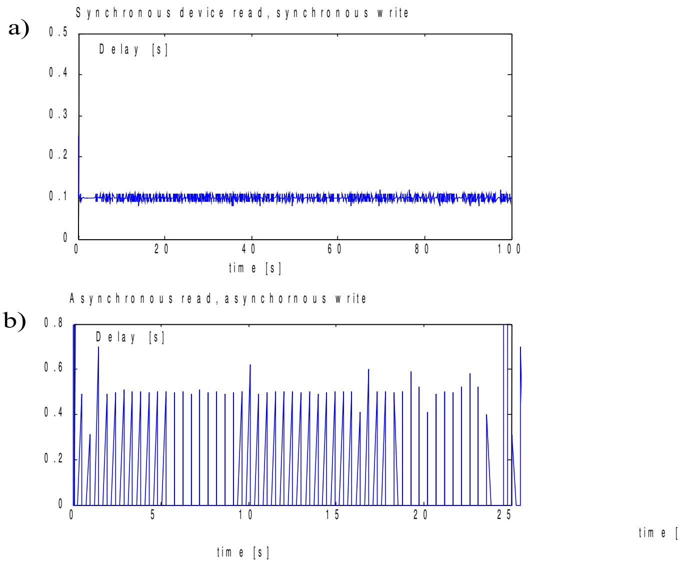


Fig.6 Time delays for synchronous and asynchronous modes

Table 2. Jitter introduced by network [13]

Network	Synchronous		
	Mean jitter [ms]	Jitter highest [ms]	Jitter lowest [ms]
Ethernet	0.4231	0.4323	0.493
Wireless	0.7222	0.8889	1.7778
TCP Ethernet with traffic 50%	5.7105	20.7895	24.4737

Network type jitters and delays dominate for heavy traffic. In this case the highest jitter gets 20% of T_{ITEM} time and might influence on control quality i.e. to violate conventional control theories assumptions such as evenly spaced sampling sensing and actuation.

VI. Conclusions

Various requirements for distributed, real-time control of a process in process automation can be formulated, for example stability, speed of response etc. Most important is to ensure that the data required by the controller are delivered to the controller and for the data from the controller to the process again, in less time that it takes for the any appreciable change to occur in the output of the process. With the industrial network speeds that increase application layer delays can become more important than network delays. As a consequence, application layer delays must be taken into account but they are often not specified.

OPC is a standard that has been developed to help with inoperability problems with the number of different devices and communications protocols in the process industry. OPC was never intended for a hard real-time operation. The latency and jitter associated with OPC (and DCOM in general) is significant. Even though data read from an OPC server bears timestamps, there is no guarantee that the client will receive them in a reasonable time. Therefore, adding OPC to a process could influence the dynamics of the control loop and could cause problems in controller design and implementation.

There are a number of researches focused on COM technology to demonstrate that the COM/DCOM model after some extensions can guarantee real-time communication to industrial control systems [14], [15].

Our experiments have shown that standard OPC DA in asynchronous mode has slow and unpredictable read times. However, if properly configured, the OPC is capable of providing a loop time less than the time constants of many industrial processes. Especially, it is useful for supervisory control level, where sampling intervals can be in the range of seconds or even minutes. It can be also used to send set-point commands down from the SCADA to the device level. OPC standard has its place in the distributed control configurations of all processes that do not have fast responding outputs.

Acknowledgment

Author thanks Mr. Marcin Bajer for valuable experiments he performed as a ;part of his Master Thesis [13].

References

[1] P. Tatjewski, *Advanced Control of Industrial Processes*. Springer-Verlag, London, 2007.

[2] W. Grega, W. Byrski and J. Duda, "InStePro – Integrated Production Control", Available (2010): http://www.InStePro.agh.edu.pl .

[3] W. Grega, "Design of Distributed Control Systems: from Theoretical to Applied Timing", in *Recent advances in control and automation*, Warszawa, pp. 343–352, 2008.

[4] L. Almeida, "Networks for Embedded Control Systems", in: *ARTIST2 China Summer School, Shanghai*. Available ((2010): http://www.artist-embedded.org/docs/Events/08/China.

[5] [Online]. Available (2010): http://www. opcfoundation.org.

[6] [Online]. "OPC DA 3.00 Specification. OPC Foundation", Available (2010): http://www. opcfoundation.org.

[7] W. Zhang, M. Branicky and S. Philips, "Stability of Networked Control Systems", *IEEE Control System Magazine*, vol. 21, pp. 84-99, 2001.

[8] W. Grega, "Stability of Distributed Control Systems with Uncertain Delays", in *8th IEEE International Conference on Methods and Models in Automation and Robotics*, pp. 303 – 307, 2002.

[9] N. Kalappa, J Moyne, J Parrott and Y.Shian Li, "Practical Aspects Impacting Time Synchronization Data Quality in Semiconductor Manufacturing", In: *Proceedings of the IEEE 1588 Conference*, October 2006.

[10] L. Larsson, Technical article: Fourteen Industrial Ethernet solutions under the spotlight, *The Industrial Ethernet Book –* 2005 – 28 – Available (2005): http://ethernet.industrialnetworking.com/ articles/articledisplay.asp ?id=854.

[11] M. Felsner, "Real-Time Ethernet – Industry Prospective", in: *Proceedings of the IEEE*, vol. 93, pp. 1118- 1129, June 2005.

[12] D. Jansen and H. Buttner, "Real-time Ehernet the EtherCAT Solution", *Computing & Control Engineering Journal*, vol. 15, pp. 16 – 21, Jan. 2006.

[13] M. Bajer, "Control Systems Integration using OPC Standard", AGH Master Thesis, W. Grega -Supervisor, Krakow &Antwerp 2008.

[14] Paul Fischer, "Real Time Extensions to OPC", *Real Time Magazine*, vol. 98Q3, p .76.

[15] Y. Veryha, "Going beyond performance limitations of OPC DA implementation", In *Proceedings of the 10th IEEE Conference on Emerging Technology and Factory Automation*, pp. 47−53, 2005.

[16] 16. J. T. Parrott, J. R. Moyne, and D. M. Tilbury, "Experimental Determination of Network Quality of Service in Ethernet: UDP, OPC, and VPN", In *2006 American Control Conference, ACC'06*, Minneapolis, USA, June 2006.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 797–804

ISBN 978-83-60810-27-9
ISSN 1896-7094

Integration of Scheduling Analysis into UML Based Development Processes Through Model Transformation

Matthias Hagner and Ursula Goltz
Institute for Programming and Reactive Systems,
TU Braunschweig,
Mühlenpfordtst. 23,
38106 Braunschweig, Germany,
{hagner, goltz}@ips.cs.tu-bs.de

***Abstract*—The complexity of embedded systems and their safety requirements have risen significantly in recent years. Models and the model based development approach help to keep overview and control of the development. Nevertheless, a support for the analysis of non-functional requirements, e.g. the scheduling, based on development models and consequently the integration of these analysis technologies into a development process exists only sporadically. The problem is that the analysis tools use different metamodels than the development tools. Therefore, a remodeling of the system in the format of the analysis tool or a model transformation is necessary to be able to perform an analysis. Here, we introduce a scheduling analysis view as a part of the development model, which is a MARTE annotated UML model to describe a system from the scheduling behavior point of view. In addition, we present a transformation from this annotated UML model to the scheduling analysis tool SymTA/S and a treatment of the analysis results to integrate scheduling analysis into a development process. With our approach it is not necessary to remodel the system in an analysis tool to profit from the analysis and its results. Additionally, we illustrate our approach in a case study on a parallel robot controller.**

I. Introduction

Model based development is widely appreciated in the embedded systems domain to cope with the complexity. As a generic and standardized approach, UML [1] has established as one of the most important notations for modeling software. The MARTE profile (UML Profile for Modelling and Analysis of Real-Time and Embedded systems) [2] makes domain specific ideas, relevant for real-time and embedded systems design, available in a unified modeling framework. Based on this profile, we have defined the UML scheduling analysis view [3] as a part of the design model that concentrates on the scheduling analysis aspects and leaves out all unnecessary information to help the developer focusing on these aspects. This approach is based on the cognitive load theory [4], which states that human cognitive productivity dramatically decreases with the amount of different dimensions to be considered at the same time.

Besides specification and tracing of timing requirements through different design stages, the major goal of enriching models with timing information is to enable early validation and verification of design decisions. As designs for an embedded or safety critical systems may have to be discarded if deadlines are missed or resources are overloaded, early timing analysis has become an issue and is supported by a number of specialized analysis tools like SymTA/S [5], MAST [6], and TIMES [7]. However, the metamodels on which analysis models of the tools are based differ from each other and in particular from UML models used for design, especially the UML scheduling analysis view. Thus, to make an analysis possible and to integrate it into a development process, the developer has to remodel the system in the analysis tool. To avoid this major effort, an automatic model transformation is needed to build an interface that enables automated analysis of a MARTE extended UML model using existing RT analysis technology.

In this paper, we introduce a method to integrate the scheduling analysis into a UML based development process. With our approach it is possible to add scheduling parameters to a UML model and to perform scheduling analysis based on this model. Therefore, we demonstrate a model transformation (realized with the Atlas Transformation Language - ATL [8]) from our proposed UML scheduling analysis view as an extension of Papyrus for UML[1] to the scheduling analysis tool SymTA/S. After the analysis, the results are published in the UML development model to give the developer a feedback concerning the scheduling behavior such that he/she can draw the right conclusions. Consequently, the developer does not need to see SymTA/S or to know how to use the analysis tool to profit from scheduling analysis.

This paper is structured as follows: Section II gives an introduction over the integration of scheduling analysis into the development process, Section III presents the UML scheduling analysis view, in Section IV, SymTA/S

[1] http://www.papyrusuml.org

978-83-60810-27-9/09/$25.00 © 2010 IEEE

is explained, Section V and VI present the transformation from the UML scheduling analysis view to SymTA/S and the recirculation and publication of results of the analysis. In Section VII we apply our approach to a case study of our Collaborative Research Centre 562. Section VIII concludes the paper.

II. Integration of Scheduling Analysis into a Development Process

The idea of performing scheduling analysis based on UML models assumes that all the information that is needed for the analysis is already part of the UML model. There is no scheduling analysis tool that is based on UML models or that uses UML models as an input. Therefore, the transformation described in Section V is necessary. But for this transformation, it is necessary that all information needed is already part of the UML development model. If this is the case, the data/necessary information of the system is transformed into the format of the analysis tool.

After the transformation is done, the analysis examines the response times of the tasks and the utilization of the resources. It checks if task chains are executed fast enough or deadlines are missed. Following the analysis, the results are published again in the UML scheduling analysis view (see Figure 1 for the workflow). These steps are done automatically and the developer only has to start the workflow and gets back the analysis results immediately. Afterwards, he/she just has to interpret the analysis results. Consequently, the developer does not need to see or use the analysis tool, as he/she can model and parameterize the system in the UML development model. Even the analysis results are published using UML and the MARTE profile.

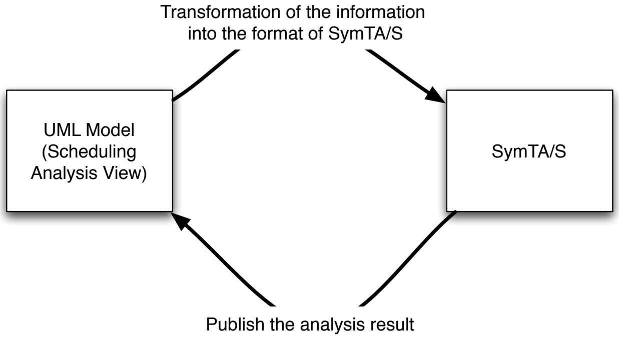

Transformation of the information into the format of SymTA/S
UML Model (Scheduling Analysis View)
SymTA/S
Publish the analysis result

Fig. 1. Illustration of the transformation flow

This approach is independent from the point in time during the development process, as long as the system under consideration is completely described to meet the criteria of a scheduling analysis. It is useful at the beginning of a development process to help making decisions concerning the number of resources or the distribution, even if at this time mainly estimated values must be used and the system is not completely designed (e.g. tasks are still combined and not yet broken down). At a later stage of development, more exact/measured (execution) times can be used to determine more accurate analysis results. Regardless of the development stage, the workflow, described in Figure 1, is the same.

III. The UML Scheduling Analysis View

The UML scheduling analysis view [3] was designed regarding the information required by a number of scheduling analysis tools (e.g. SymTA/S). Therefore, it concentrates on and highlights timing and scheduling aspects. Unlike usual design models, it contains all necessary information for an analysis (priorities, scheduling algorithms, task execution times, etc.).

The view consists of three different UML diagram types and a selection of MARTE stereotypes and tagged values (see Figure 2). The diagram types are class diagrams as an architectural view (to describe the structure, associations, and allocations of the systems' elements), object diagrams as a view on the runtime system (to instantiate the concrete system that should be analyzed based on the architectural view), and activity diagrams as a workload view (to describe workload situations, flows, and dependencies of tasks).

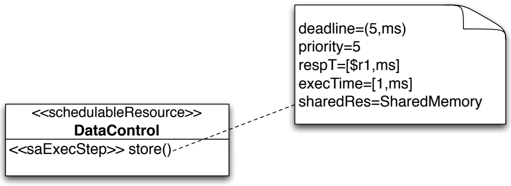


Fig. 2. Example of MARTE stereotypes and tagged values

The stereotypes and the tagged values are based on the MARTE UML profile. We only used a small amount of the defined elements for the scheduling analysis view. One goal of the view is to keep it as simple as possible. Therefore, only elements are used that are necessary to describe all the information that is needed for an analysis. In Table I all used stereotypes and tagged values are presented.

Class diagrams are used to describe the architectural view/the structure of the modeled system. The diagrams show resources, tasks, and associations between these elements. Furthermore, schedulers and other resources, like shared memory, can be defined. Figure 3 shows a class diagram of the scheduling analysis view that describes the architecture of a sample system. The functionalities/the tasks and communication tasks are represented by methods. The tasks are described using the «saExecStep» stereotype. The methods that represent the communication tasks (transmitting of data over a bus) are extended with the «saCommStep» stereotype. The tasks or communication tasks, represented as methods, are part of schedulable resource classes (marked with the «schedulabeResource» stereotype), which combine tasks

TABLE I
ELEMENTS OF THE SCHEDULING ANALYSIS VIEW

Stereotype	used on	Tagged Values
«saExecHost»	Classes, Objects	Utilization, mainScheduler, isSched
«saCommHost»	Classes, Objects	Utilization, mainScheduler, isSched
«scheduler»	Classes, Objects	schedPolicy, otherSchedPolicy
«schedulableResource»	Classes, Objects	
«saSharedResources»	Classes, Objects	
«saExecStep»	Methods	deadline, priority, execTime, usedResource, respT
«saCommStep»	Methods	deadline, priority, execTime, msgSize, respT
«saEndToEndFlow»	Activities	end2endT, end2endD, isSched
«gaWorkloadEvent»	Initial-Node	pattern
«allocated»	Associations	

or communications that belong together, e.g. since they are part of the same use case or all of them are service routines. Processor resources are represented as classes with the «saExecHost» stereotype and bus resources are classes with the «saCommHost» stereotype. The tasks and communications are mapped on processors or busses by using associations between the schedulable resource and the corresponding bus or processor resource. The associations are extended with the «allocated» stereotype. Scheduling relevant parameters (like deadlines, execution times, priorities, etc.) are added to the model using tagged values.

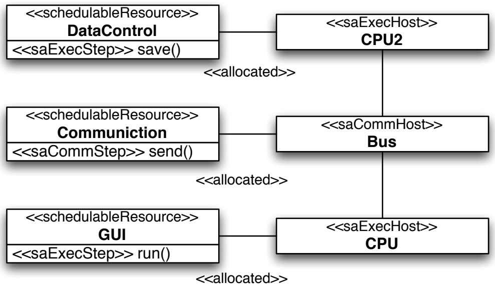


Fig. 3. The architectural view of the scheduling analysis view

The object diagram or runtime view is based on the class diagram/architectural view of the scheduling analysis view. It defines how many instances are parts of the runtime system respectively what parts are considered for the scheduling analysis. It is possible that only some elements defined in the class diagram are instantiated. Furthermore, some elements can be instantiated twice or more (e.g. if a processor is redundant). Only instantiated objects are taken into account for the scheduling analysis and, consequently, for the model transformation.

Activity diagrams are used to describe the behavior of the system. Therefore, workload situations are defined that outline the flow of tasks that are executed during a certain mode of the system. The dependencies of tasks and the execution order are illustrated. The «gaWorkloadEvent» and the «saEnd2EndFlow» stereotypes and their corresponding tagged values are used to describe the workload behavior parameters like the arrival pattern of the event that triggers the flow or the deadline of the outlined task chain. For example, in Figure 4, it is well defined that at first *cpu.run()* has to be completely executed, before *communication.send()* is scheduled etc.. There are restrictions like that no decision nodes are allowed, because the analysis tool SymTA/S does not support these model elements.

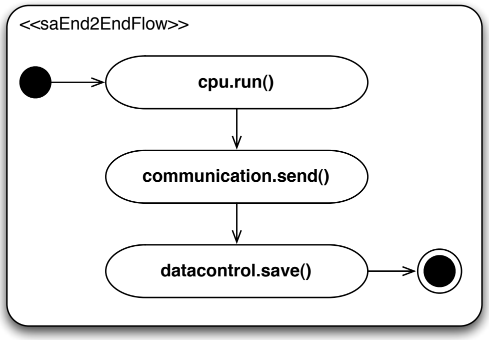


Fig. 4. A workflow description of the scheduling analysis view

The scheduling analysis view can be easily be extended, if necessary. If a scheduling analysis tool offers more possibilities to describe or to analyze a system (e.g. a different scheduling algorithm) and needs more system parameters for it, these parameters have to be part of the scheduling analysis view. Therefore, the view can be extended with new tagged values that offer the possibility to add the necessary parameters to the system description (added to Table I).

Another advantage of using the scheduling analysis view is that the tagged values help the developer to keep track of timing requirements during the development, as these parameters are part of the development model. This especially helps to keep considering them during refinement.

IV. SCHEDULING ANALYSIS WITH SYMTA/S

We use SymTA/S (**Sym**bolic **T**iming **A**nalysis for **S**ystems) [4] for the scheduling analysis. The example depicted in Figure 5 is the SymTA/S representation of the system described in Section III and illustrated in Figure 3 and Figure 4. There is one source (trigger), two CPUs (CPU and CPU2), which run two tasks (run and save), and a bus (Bus) with one communication task (send). All tasks are connected using event streams, representing task chains.

SymTA/S links established analysis algorithms with event streams and realizes a global analysis of distributed systems. At first, the analysis considers each resource on its own and identifies the response time of the mapped tasks. From these response times and the given input event model it calculates the output event model and propagates it by the event stream. If there are cyclic dependencies, the system is analyzed from a starting point iteratively until reaching convergence.

V. Transformation and Analysis

During the transformation, the information is taken from the UML metamodel of the scheduling analysis view and is placed in the metamodel of SymTA/S. The scheduling analysis view is modeled in Papyrus for UML and saved as an XMI[9] file. SymTA/S files are also based on XML. Therefore, the ATL transformation [8] has to read the XMI file and create a, for SymTA/S valid, XML file containing the system represented in the scheduling analysis view.

First, the transformation checks whether the model is suitable at all for a transformation. After that, the hardware components are transformed. The transformation searches for all objects with the «saExecHost» stereotype and generates CPU objects for them in the SymTA/S model. The same happens with objects with the «saCommHost» stereotype for busses.

Thereafter, the activity diagrams/workflow descriptions are considered, and, depending on the parameters of the initial nodes, corresponding source elements in the SymTA/S model are created. In the next step, all tasks and communication tasks (all methods with a «saExecStep» or «saCommStep» stereotype) are created as often, as the class/schedulable resource they belong to is instantiated in the scheduling analysis view (e.g. if a class "Calculate" with the «schedulableResource» stereotype has a method *calculateValues()* that has the «saExecStep» stereotype, and the "Calculate" class is instantiated two times in the scheduling analysis view, there will be two calculateValues tasks in the SymTA/S model).

Analyzing the «allocated» stereotypes does the mapping: For example, if there is an «allocated» marked association between a resource and a class with a «schedulabeResource» stereotype, all methods/tasks of the «schedulabeResource» class are mapped on this resource in the SymTA/S model. The corresponding parameters of the tasks (annotated with tagged values in the UML scheduling analysis view) are entered into the SymTA/S model.

In the last step of the transformation, the event streams in SymTA/S are created based on the dependencies described in the activity diagrams/workflow descriptions. The activity diagrams of the UML scheduling analysis view are observed and the dependencies are added to the SymTA/S model. It is only necessary to detect which actions are connected in the UML view and to reproduce these connections using Event Streams in SymTA/S. As SymTA/S does not offer decision nodes, these elements are not supported during the transformation.

After the complete transformation, SymTA/S is used for the analysis. For this, the tool is started with the created XML file as a parameter. SymTA/S analyses the system and writes back the results into the XML file.

VI. Publishing the Analysis Results in the Scheduling Analysis View

After the analysis is finished, the results are published in the scheduling analysis view. The developer gets the information whether there are tasks or paths/task chains that are not schedulable or that miss their deadlines. Some of the tagged values of the scheduling analysis view are used to give the developer a feedback about the analysis results. For example, in Figure 2 the respT tagged value is empty before the analysis and has a variable ($r1), which means that the response time of the corresponding task is entered at this point after the analysis. There are also other parameters, which give a feedback to the developer (see also Table I):

a) respT: The respT tagged values gives a feedback about the response time of the (communication) tasks and is offered by the «saExecStep» and the «saCommHost» stereotype.

b) end2endT: As the respT, the end2endT tagged values offers the response time, in this case of a path/task chain and is offered by the «saEnd2EndFlow» stereotype. It is not a summation of all response times of the tasks that are part of the path, but a worst case calculated response time of the whole path examined by the scheduling analysis tool (for more details see [5]).

c) Utilization: The «saExecHost» and the «saCommHost» stereotype offer a utilization tagged value that gives a feedback about the load of the CPU or the bus. The value is given in percent. If the value is higher than 100%, it is obvious that this resource is not schedulable (and the isShed tagged value is false too), but even if the value is only slightly under 100% , this is a warning for the developer that this resource is nearly overloaded.

d) isShed: This tagged value gives a response whether the tasks mapped on this resource are schedulable or not (false or true) and is offered by the «saExecHost» and the «saCommHost» stereotype. The tagged values are connected to the Utilization tagged value (e.g. if the utilization is higher than 100%, the isShed tagged value is false). This tagged value is also offered by the «saEnd2EndFlow» stereotype. As the «saEnd2EndFlow» stereotype defines parameters for a path/task chain, the

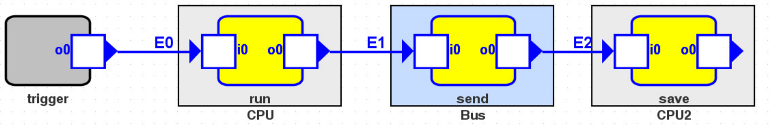


Fig. 5. A sample of a system described in SymTA/S

isShed tagged value gives a feedback whether the deadline for the path is missed or not.

After an analysis is finished, the developer can check if the system is schedulable. For this, the developer checks if the paths/tasks chains are schedulable (isShed tagged value of the «seEnd2EndFlow» stereotype). If this is false, the developer has to find the reason why the scheduling failed. The end2EndT tagged value shows to what extent the deadline is missed, as it gives the response time of the path/task chain. The response times of the tasks and the utilization of the resource give also a feedback where the bottleneck might be (e.g. a resource with a high utilization and tasks scheduled on it with long response times is more likely a bottleneck as resources with low utilizations).

VII. Case Study

The aim of the Collaborative Research Centre 562 (CRC 562)[2] is the development of methodological and component-related fundamentals for the construction of robotic systems based on closed kinematic chains (parallel kinematic chains - PKMs), to improve the promising potential of these robots, particularly with regard to high operating speeds, accelerations, and accuracy [10]. This kind of robots features closed kinematic chains and has a high stiffness and accuracy. Due to low moved masses, PKMs have a high weight-to-load-ratio compared to serial robots. The demonstrators which have been developed in the research center 562 move very fast (up to 10 m/s) and achieve high accelerations (up to 100 m/s^2). The high velocities induced several hard real-time constraints on the software architecture that has been designed to control the robots. The latest version is *PROSA-X* (**P**arallel **R**obots **S**oftware **A**rchitecture - e**X**tended) which uses multiple control PCs to distribute its algorithmic load. *PROSA-X* is a generic architecture which is used to control several different robots. A specifically tailored middleware (*MiRPA-X*) and a bus protocol that operates on top of a FireWire bus (*IAP*) realize communication satisfying the hard real-time constraints [11]. The architecture is based on a layered design with multiple real-time layers within QNX[3] to realize e.g. a deterministic execution order for critical tasks [12].

The robots are controlled using cyclic frequencies between 1 and 8 kHz. If these hard deadlines are missed, this could cause damage to the robot and its environment. To avoid such problems, a scheduling analysis on the basis of models ensures the fulfillment of real-time requirements. The case study presented below has been performed in this context and is based on [13].

In Figure 6 a scheduling analysis view representation of the software architecture, PROSA-X [14], is presented. It consists of a control PC ("Control_PC1"), which performs various computing tasks ("CP1_Tasks"). The control PC is connected via a FireWire (IEEE 1394) data bus with a number of other processors ("DSP_1-7"). The DSPs supervise and control the machine.

The methods represent the following tasks:

- *IAP_D*: This instance of the *IAP* bus protocol receives the *DDTs (Device Data Telegram)* that contain the instantaneous values of the DSP nodes over the FireWire bus.
- *HWM*: The *Hardware Monitoring* takes the instantaneous values received by the *IAP_D* and prepares them for the control.
- *DC*: The *Drive Controller* operates the actuators of the parallel kinematic machine.
- *SMC*: The *Smart Material Controller* operates the active vibration suppression of the machine.
- *IAP_M*: This instance of the bus protocol *IAP* sends the setpoint values, calculated by DC and SMC, to the DSP node.
- *CC*: The *Central Control* activates the currently required sensor and motion modules (see below) and collects their results.
- *CON*: *Contact Planner*. Combination of power and speed control. For the end effector of the robot to make contact with a surface.
- *FOR*: *Force Control*, sets the force for the end effector of the robot.
- *CFF*: Another *Contact Planner*, similar to CON.
- *VEL*: *Velocity Control*, sets the speed for the end effector of the robot.
- *POS*: The *Position Controller* sets the position of the end effector.
- *SAP*: The *Singularity Avoidance Planner* plans paths through the work area to avoid singularities.
- *SEN*: An exemplary *Sensor Module*.

Next we explain the real-time demands on the system. There is no deadline on a specific task, but there are deadlines on task chains/workflows. The first task chain receives the instantaneous values and calculates the new setpoint values (IAP_D, HWM, DC, SMC). The deadline

[2] http://www.tu-braunschweig.de/sfb562
[3] QNX Neutrino is a micro kernel real-time operating system.

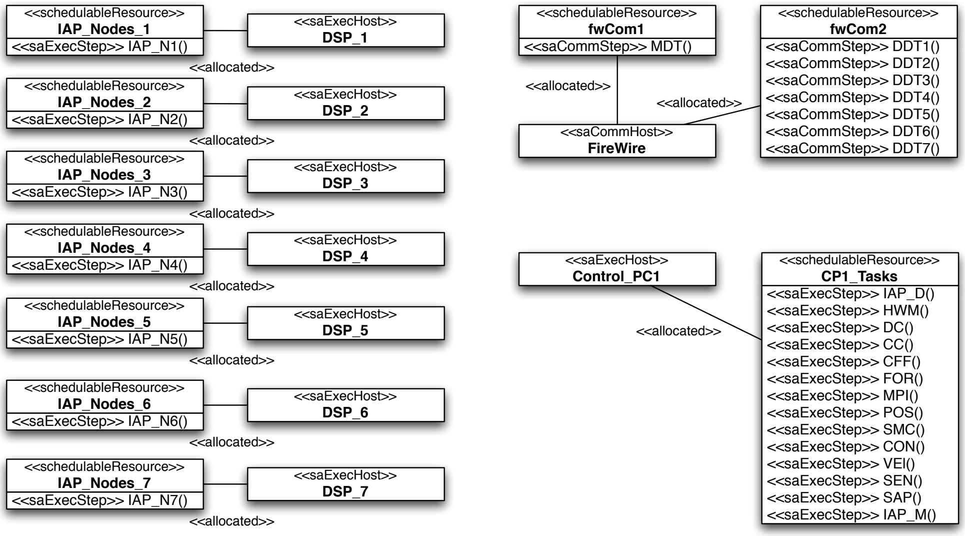


Fig. 6. The architectural view of the PROSA-X system

for this is after 250 microseconds (see Figure 7 for the scheduling analysis view description).

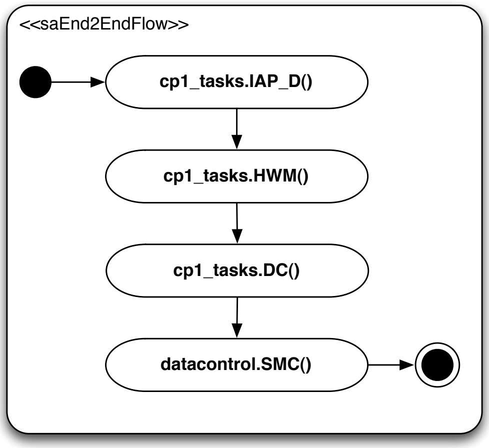


Fig. 7. The receiving of the instantaneous values and the calculation of the new setpoint values

The second task chain contains the sending of the setpoint values to the DSPs and their processing (IAP_M, MDT, IAP_N1, …, IAP_N7, DDT1, …, DDT7). This must be finished within 750 microseconds (see Figure 8).

Finally, the third chain comprises the control of the sensor and motion modules (CC, CON, FOR, CFF, POS, VEL, SEN, SAP) and has to be completed within 1945 microseconds (see Figure 9).

The system was modeled in Papyrus for UML as a Scheduling Analysis View. Every element of the architectural view (depicted in Figure 6) is instantiated once in the runtime view.

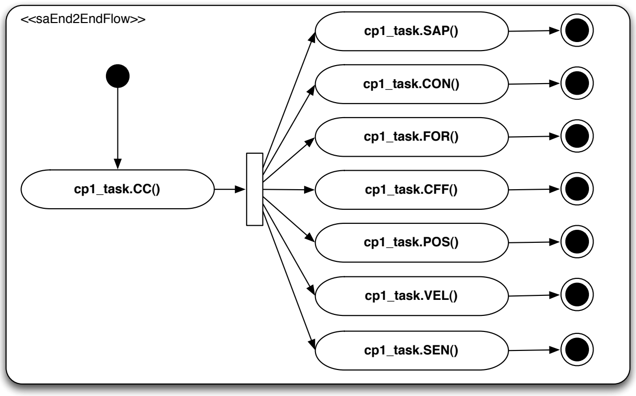


Fig. 9. Control of the sensor and motion modules

The ATL transformation creates a corresponding SymTA/S model and makes it possible to analyze the system. The transformation was successful, the output model was analyzed by SymTA/S and confirms to the expectations: The analysis was successful, all paths keep their real-time requirements, the resources are not overloaded. The SymTA/S model is depicted in Figure 10.

After the successful analysis, the results are published back into the scheduling analysis view (see section VI).

The analysis has confirmed that the system is schedulable. We also integrated this analysis into our approach for finding the best distribution using graph partitioning [15].

VIII. Conclusion and Further Work

We have presented a model transformation from a MARTE annotated UML model to the scheduling analysis tool SymTA/S to make a scheduling analysis based

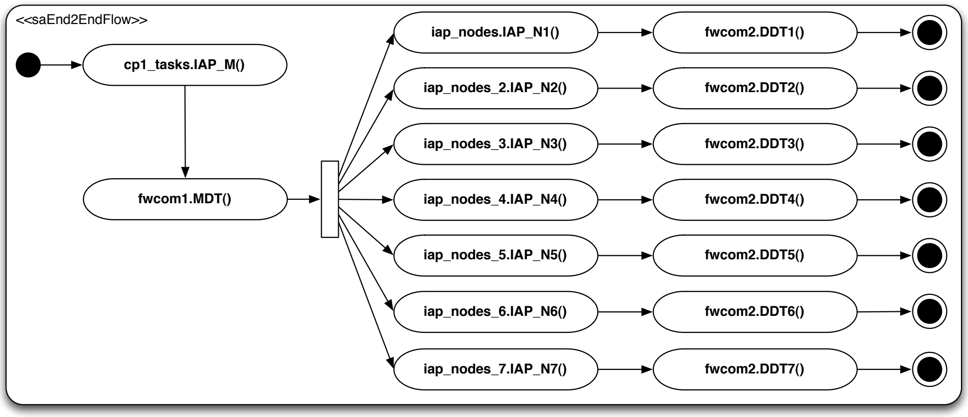


Fig. 8. Sending of the setpoint values to the DSPs

on UML design models possible and to limit the effort for a developer to get feedback concerning the timing/scheduling behavior. We have demonstrated this transformation on a parallel robot controller of our CRC 562. Further steps would be to create more model transformations for other non-functional parameters to corresponding analysis tools.

Another extension of this approach would be to describe a method how to create the scheduling analysis view and how to add the necessary and typically missing parameters to this view to make a scheduling analysis possible (we have made first approaches in [16] and [17]). It is also possible to integrate further tools into the method (e.g. aiT [18]).

Furthermore, a method to help finding the bottleneck if an analysis fails would be of great benefit.

Acknowledgment

The authors would like to thank Symtavision[4] for the grant of free licenses.

References

[1] OMG Object Management Group, “Unified modeling language specification,” 2003.

[2] ——, “UML profile for modeling and analysis of real-time and embedded systems (MARTE),” 2009.

[3] M. Hagner and M. Huhn, “Tool support for a scheduling analysis view,” in *Design, Automation and Test in Europe (DATE 08)*, 2008.

[4] J. Sweller, “Evolution of human cognitive architecture,” in *The Psychology of Learning and Motivation*, vol. 43, 2003, pp. 215–266.

[5] R. Henia, A. Hamann, M. Jersak, R. Racu, K. Richter, and R. Ernst, “System level performance analysis - the SymTA/S approach,” *IEEE Proceedings Computers and Digital Techniques*, vol. 152, no. 2, pp. 148–166, March 2005. [Online]. Available: citeseer.ist.psu.edu/jersak05system.html

[6] M. G. Harbour, J. J. G. García, J. C. P. Gutiérrez, and J. M. D. Moyano, “Mast: Modeling and analysis suite for real time applications,” in *ECRTS ’01: Proceedings of the 13th Euromicro Conference on Real-Time Systems*. Washington, DC, USA: IEEE Computer Society, 2001, p. 125.

[7] E. Fersman and W. Yi, “A generic approach to schedulability analysis of real-time tasks,” *Nordic J. of Computing*, vol. 11, no. 2, pp. 129–147, 2004.

[8] OMG Object Management Group, “MOF 2.0, query / views / transformation ad/2002-04-10, revised submission, version 1.0, 2003/08/18, OpenQVT.”

[9] ——, “XML model interchange(XMI),” 1998.

[10] J.-P. Merlet, *Parallel Robots.* Kluwer Academic Publishers, 2000.

[11] N. Kohn, J.-U. Varchmin, J. Steiner, and U. Goltz, “Universal communication architecture for high-dynamic robot systems using QNX,” in *Proceedings of International Conference on Control, Automation, Robotics and Vision (ICARCV 8th)*, vol. 1. Kunming, China: IEEE Computer Society, December 2004, pp. 205–210, iSBN: 0-7803-8653-1.

[12] J. Maass, N. Kohn, and J. Hesselbach, “Open modular robot control architecture for assembly using the task frame formalism,” *International Journal of Advanced Robotic Systems*, vol. 3, no. 1, pp. 1–10, 2006, iSSN: 1729-8806.

[13] A. Bragenheim, “Realisierung einer Modelltransformation zur Schedulability-Analyse von UML-Modellen mit SymTA/S,” 2009.

[14] J. Steiner, U. Goltz, and J. MaaSS, “Dynamische verteilung von steuerungskomponenten unter erhalt von echtzeiteigenschaften,” in *6. Paderborner Workshop Entwurf mechatronischer Systeme*, 2009.

[15] J. Steiner, A. Amado, U. Goltz, M. Hagner, and M. Huhn, “Engineering self-management into a robot control system,” in *Proceedings of 3rd International Colloquium of the Collaborative Research Center 562*, 2008.

[16] M. Hagner and M. Huhn, “Modellierung und analyse von zeitanforderungen basierend auf der uml,” in *Workshop*, ser. LNI, H. Koschke, Ed., vol. 110, 2007, pp. 531–535.

[17] M. Hagner, M. Huhn, and A. Zechner, “Timing analysis using the MARTE profile in the design of rail automation systems.” in *4th European Congress on Embedded Realtime Software (ERTS 08)*, 2008.

[18] C. Ferdinand, R. Heckmann, M. Langenbach, F. Martin, M. Schmidt, H. Theiling, S. Thesing, and R. Wilhelm, “Reliable and precise wcet determination for a real-life processor,” in *EMSOFT ’01: Proceedings of the First International Workshop on Embedded Software.* London, UK: Springer-Verlag, 2001, pp. 469–485.

[4] http://www.symtavision.com

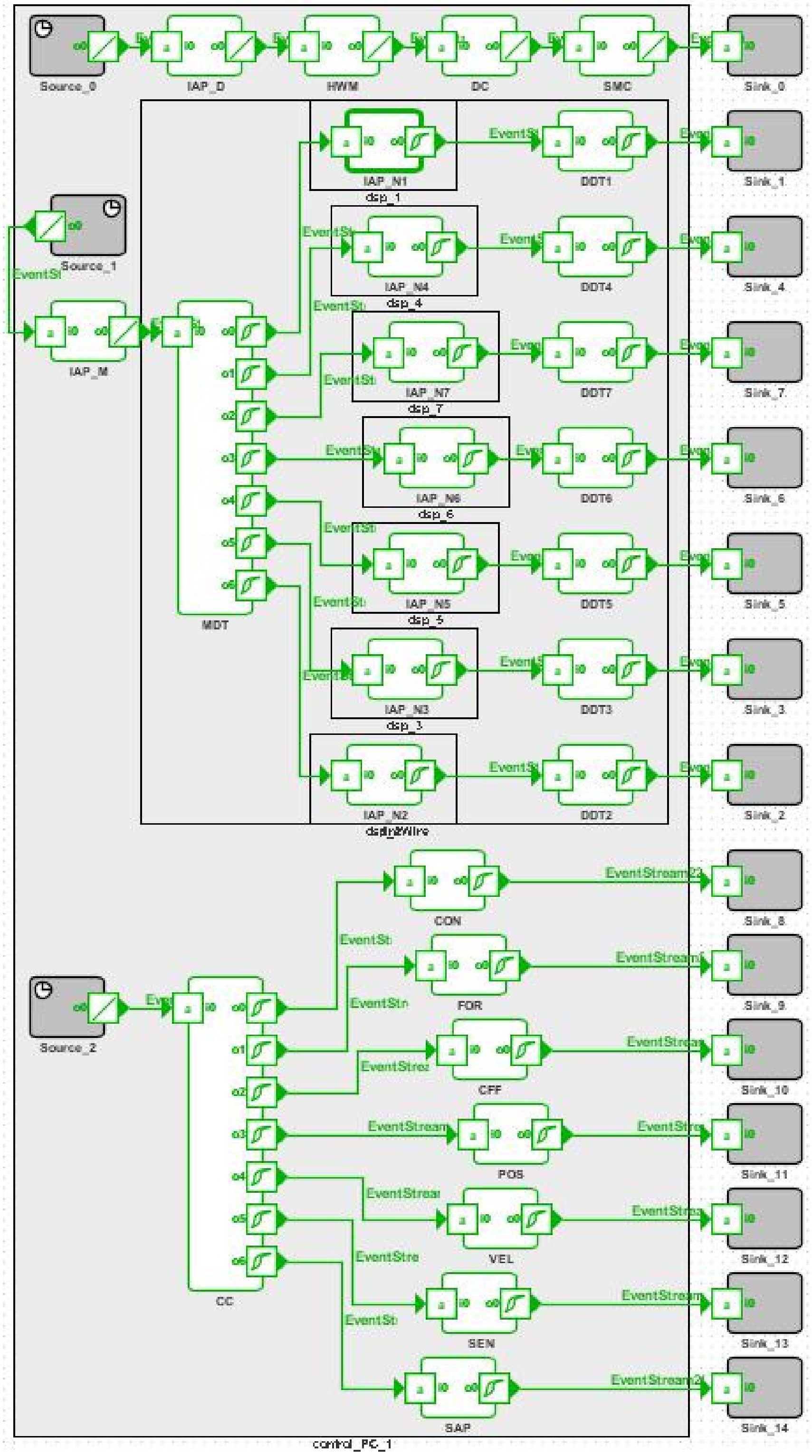


Fig. 10. The SymTA/S description of the PROSA-X system

ISBN 978-83-60810-27-9
ISSN 1896-7094

Laboratory real-time systems to facilitate automatic control education and research

Krzysztof Kołek,
Andrzej Turnau,
Krystyn Hajduk
AGH University of Science and Technology al. Mickiewicza 30,
30-059 Kraków, Poland
Email: {kko, atu,kha}@agh.edu.pl

Paweł Piątek, Mariusz Pauluk,
Dariusz Marchewka
AGH University of Science and Technology al. Mickiewicza 30,
30-059 Kraków, Poland
Email: {ppi, mp,dmar}
@agh.edu.pl

Adam Piłat, Maciej Rosół,
Przemysław Gorczyca
AGH University of Science and Technology al. Mickiewicza 30,
30-059 Kraków, Poland
Email: {ap, przemgor,mr}
@agh.edu.pl

Abstract—**The paper is an attempt to interest the reader how to control real-time mechatronic systems under the MS Windows operating system. The authors refer to solutions that combine a software part and a hardware using the FPGA technology, together forming a comprehensive platform for the control purposes in the real-time. The main emphasis is placed on the authors' own designs and constructions. Lectures and laboratory experiments must be conducted hand in hand. Such is the message to facilitate the *Automatic Control* education. Research works have been carried out in the Department of Automatics at the University of Science and Technology (AGH).**

I. Introduction

THE paper presents important aspects of education provided with the use of laboratory experiments. How to facilitate the teaching of control theory which can not go without a control experiment performed in the real-time? Researcher, scientist, teacher, engineer should use the laboratory computer controlled systems. That is the systems that we create and by which we teach young people. These systems are the subject of further considerations. The work focuses on:

❍ development of software to enable a control in real time involving a popular MS Windows operating system,

❍ construction of simple and also advanced mathematical algorithms for control purposes in the real-time,

❍ implementation, verification and tests of the developed algorithms in the self-designed and constructed complete mechatronic systems such as: the gantry crane, tower crane, magnetic levitation, magnetic bearings, anti-lock braking system, pendulum on a cart, the multi tank, etc.,

❍ usability of the same software and hardware platform to perform simulations and real-time experiments.

While teaching the automatic control it is reasonable to intertwine lectures and laboratory experiments. In particular this field of knowledge requires to be exposed through demonstrative laboratory tools. A prominent pedagogue equipped only with chalk and his knowledge can be under a misapprehension that a pure theory presented at the lecture is just what was expected by the audience. In fact, such a lecture may become as interesting as watching grass grow. One has to be far from impression that he is just presenting electro-mechanical toys in motion even quite complex due to involved preprogrammed control algorithms. In fact, one may notice a smell of mechatronic systems and it is not bad. However, our address has a different goal, namely how to facilitate the automatic control education.

II. Real Time Control

A. Real-time services in MS Windows systems

MS Windows can not be considered as robust hard real-time operating systems. However some features of the system can be applied to develop a soft real-time platform. The main parameter of the real-time control systems is the sampling time period – its maximum frequency and accuracy. The timing services built into the MS Windows systems are:

❍ system timer message (WM_TIMER),

❍ multimedia timer events,

❍ kernel mode services. The RTWT kernel driver is considered in this comparison.

The system timer is posted to a thread's message queue when a timer expires. The minimum sampling period of this service is 1 millisecond. In fact, the system processes the WM_TIMER message at low priority influencing the performance. Also, the minimum sampling in multimedia services is 1 ms. But the performance, considered as the timer jitter, seems to be much better. The best performance, both in the sampling period and the jitter, is achieved by kernel mode services.

Fig. 1, presents the histograms of three services. In all cases a 10 millisecond timer event has been started. The horizontal axis presents the duration of the sampling period. The duration was measured by a hardware timer with a 25 ns resolution. The vertical axis shows the number of sampling periods of a given duration.

One can observe that the WM_TIMER service behaves in a very poor way. The period was set to 10 ms but most of the sampling periods last much longer. Even a 260 ms period was observed. It is unacceptable for electrical and mechanical control systems. On the opposite side of the time regimes appear the RTWT service. Here the accuracy and jitter of the sampling periods i less then 15 ms. The multimedia timer can be located as an in-between solution. The

This work was supported by AGH. Project no. 11.11.120.768

average value of the sampling period follows the timer period setup, but a few millisecond jitter can be observed as well.

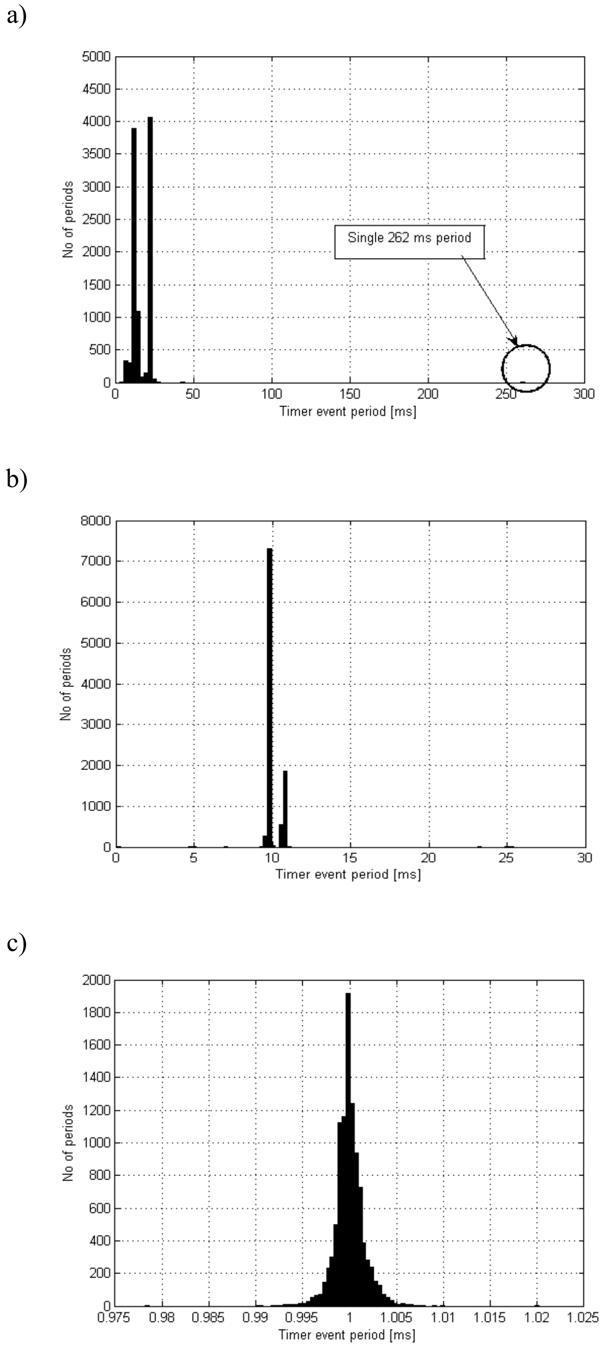


Fig.1. Histograms of the durations of the timing services: a) WM_TIMER, b) multimedia, c) RTWT timer

The timer services are executed in interrupt mode between operating system services and usually are not allowed to call all operating system functions. In the investigated timers the exception is the WM_TIMER procedure, where any OS call can be executed. The multimedia service disallows call only to a very limited set of OS functions. Such API functions like file I/Os, network functions or access to the USB stack, which are crucial to applications of measurement and control systems, can be executed in a timer service routine. The most restricted policy enforces RTWT timer. In this case the timer routine is executed at the kernel level and most of the OS API function may crush the system. The RTWT platform can be applied to cooperate with I/O boards plugged-into PCI or ISA buses but Ethernet or USB devices are unavailable.

Some functions of the control system may require a reaction faster than the response time of the presented timers. An interface to incremental encoder of safety functions can be given as an example. In such a case an additional hardware has to be used. The FPGA technology seems to be an ideal solution due to its flexibility.

III. Hardware implementation

A. RTDAC-Board

Standard PC computers are used most often as a control hardware in educational field. This solution has some advantages and disadvantages. The most important disadvantage is that PC hardware is not dedicated for control application. In most cases an additional hardware is needed. A control and/or measurement PCI or USB board is a good choice for educational usage. Control and measurement board should provide some basic features like analog inputs, analog outputs, digital inputs, digital outputs and communication with computer operating system. Some extra features e.g., incremental encoders counters are necessary for special mechatronic system like Inverted Pendulum or Crane.

Typically control and measurement hardware is used with many dynamical systems. This is the reason, that some kind of reconfigurability of the measurement board is necessary. This feature can be provide by using FPGA techniques. One of the FPGA control board is RT-DAC4/PCI [5] presented in Fig. 2.

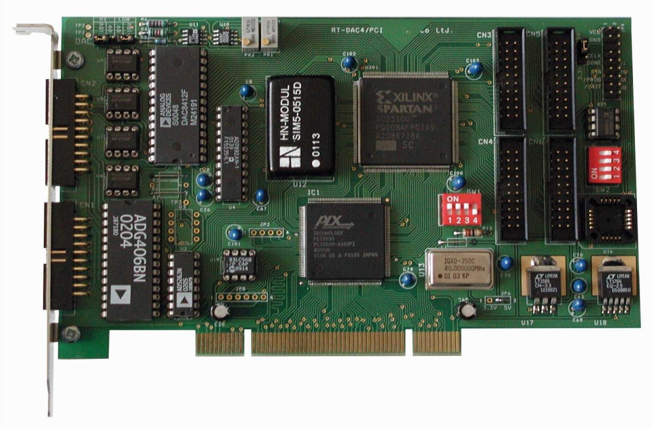


Fig. 2. RT-DAC PCI I/O board

The block diagram of this board is presented at the Fig. 3. The FPGA circuit can be reprogrammed from the PC computer level. This unique function guarantees flexibility of the device and a whole control system. User can manage configuration of the FPGA circuit by special software executed in the PC computer. Hardware functions can be modified by changing FPGA configuration.

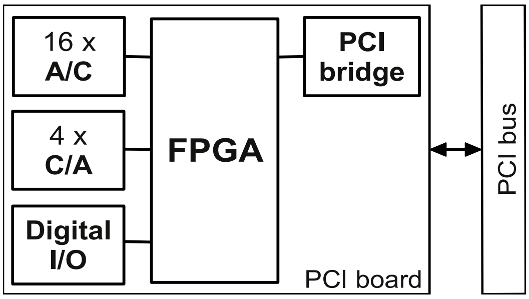


Fig 3. RT-DAC4/PCI block diagram.

FPGA control and measurement boards can work in three main modes that can be changed by reprogramming FPGA.

1. Basic mode. In this mode the board is equipped only with elementary functions. They support only simple operations with analog to digital and digital to analog converters and digital I/Os.
2. Advanced measurement mode. In this mode, the board is equipped with analog I/O operation functions, digital I/O operation functions, and some advanced measurements and control signal operation functions like incremental encoder counters, PWM functions, digital filter blocks, linearisation operation for analog signals. The control loop is closed by PC computer in this case. A designer can save additional processor time for calculations of the most advanced control algorithms in the computer by moving some operations from the software layer (PC computer control task) to the hardware layer (FPGA circuit).
3. Hardware control mode. In this type of operation the hole control task is located in the hardware layer. This is useful for time-critical dynamical systems. A high-speed control system is necessary in this case and PC computers are to slow to execute control tasks [9][8]. In this mode, only monitoring and supervisory operations are performed in a PC computer. Hierarchical high-speed control system can be also build with measurement and control board equipped with FPGA circuit. Direct control algorithms can operate in a hardware layer (FPGA) and optimisation or adaptation algorithm can operate in a software layer (PC software).

B. *Power Interface*

The power interface is an electronic device that provides connections between mechanical system and the controller (computer, PLC, microcontroller etc.). The typical structure of the power interface is shown in Fig. 4. It contains:

❍ power controller to activate the actuators,
❍ electronics for measurement signal conditioning,
❍ other electronics including optical insulation, protection against overvoltage and short circuit.

The main function of the power controller is to guarantee the required current and voltage range for actuators (DC motors, valves). The power controller operating in the PWM mode utilizes integrated bridge circuits supported by a control logic and protection system. Current and temperature are measured by integrated sensors. Depending on the controlled systems is used from one (the cart & pendulum case) to four (the tanks case) power modules. The power module provides required by the actuators voltages in the 12-24 V range and currents up to 16 A (the ABS case).

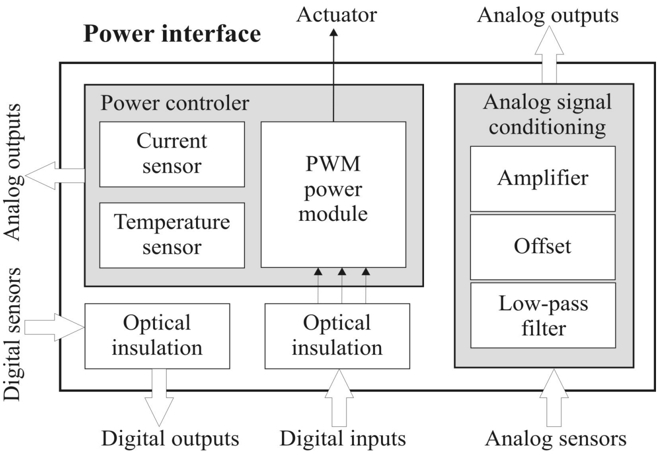


Fig. 4. Structure of the power interface

The analog sensors are connected to conditioning analog circuits. They provide the required measurement signal properties for a high quality A/D conversion. The analog signal conditioning block has an independent bipolar stable supply. Analog output signals from the conditioning block can be established as a unipolar or bipolar in the 0-10 V or ±10 V range respectively. These circuits are utilized to:

❍ increase or decrease the amplitude of the signal,
❍ filter the signal,
❍ decrease the signal output impedance,
❍ reduce the measurement signal bandwidth,
❍ provide a variable gain and offset control.

Optical insulation is used to protect the external I/O interface from overvoltage and overcurrent. It also provides a matching voltage levels.

In addition to these electronics circuits, power interface may also contains circuitry to handle voltage, pressure and strain gauge sensors.

The following sections present implementations of mechatronic systems that use the hardware just described.

IV. One Rotor Aerodynamical System (ORAS)

ORAS shown in Fig. 5 is a laboratory set-up designed for control experiments performed in the real-time. In certain aspects its behavior resembles the special type of one rotor helicopter. From the control point of view it exemplifies a high order nonlinear system with significant cross-couplings. ORAS consists of a beam pivoted on its base in such a way that the beam can rotate freely both in the horizontal and vertical planes.

At the end of the beam there is the rotor driven by a DC motor. The rotor position can be changed by the geared DC motor (see Fig. 6 and 7).

The state of the beam is described by four process variables: horizontal and vertical angles measured by encoders, and two corresponding angular velocities. Two additional

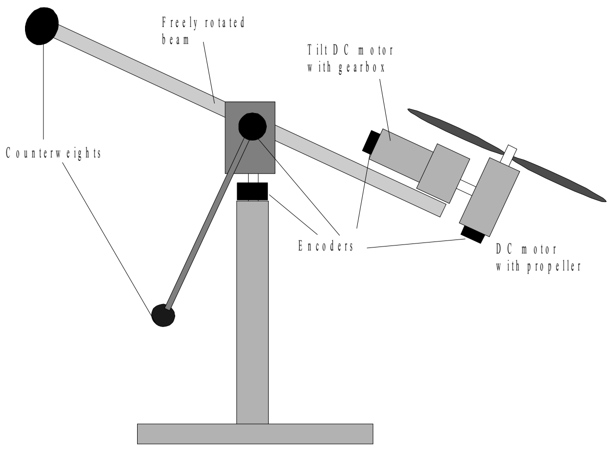


Fig.5 **O**ne **R**otor **A**erodynamical **S**ystem

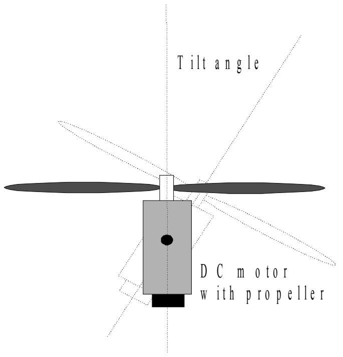


Fig. 6 Propeller tilt angle

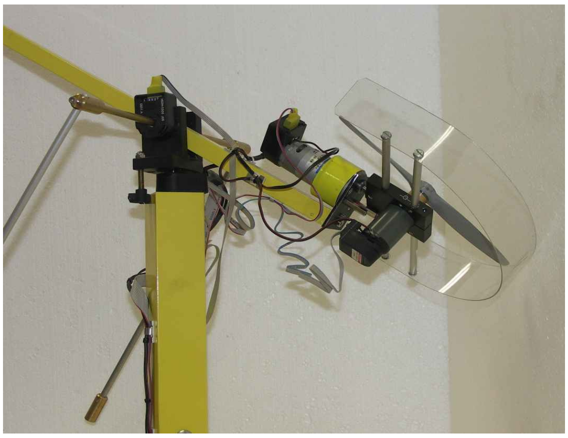


Fig.7 Laboratory set

state variables are the angular velocity of the rotor and the tilt angle. The ORAS system has been designed to operate with an external, PC-based digital controller. The control computer communicates with the position, speed sensors and motors by a dedicated I/O board and power interface. The I/O board is controlled by the real-time software which operates in the MATLAB/Simulink RTW/RTWT environment.

V. Tower Crane

Fig. 8 illustrates a general view of the model. The crane may hoist or lower a suspended payload and also move the payload along the rail and around the basis. The crane is controlled in the real-time in the MATLAB & Simulink environment [6][7].

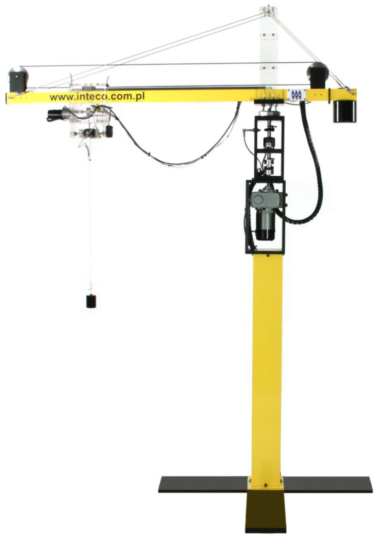


Fig. 8 General view of the tower crane

A PC computer is equipped with an analog-digital board (RT-DAC USB [5]) to transfer data between the tower crane and a controller running on the PC. Digital outputs of RT-DAC are connected to the crane power interface, where the calculated by the control algorithm value of control is converted into a PWM type voltage signal and then distributed to one among three DC-gear motors.

There are two encoders mounted on the shafts to measure rotary positions. Subsequent two encoders are placed in the trolley mechanism for measuring a deviation of the rope from the vertical position. The measurements are performed in two planes (see Fig. 9). The third gear motor is placed directly inside the crane body. The shaft that transfers the motor torque is equipped with the next encoder that measures rotary position of the crane arm with respect to the basis. The Simulink driver has three inputs: XPWM, TPWM and ZPWM to control three motions: a trolley progressive, crane rotary and payload up or down. The control values may vary from 0 to 1. The value 0 refers to no control, value 1 means full control. The control is the PWM type. A value between 0 and 1 refers to the duty cycle of the control square wave. The switch “Reset” sets the encoder counters to value 0. It is used for calibration purposes.

There are five outputs: *X Position* is the trolley position related to the jib length, *T Angle* is the jib angular position related to the crane basis, *Z Position* is the rope length of the suspended payload, *X Angle* is the angle deviation of the

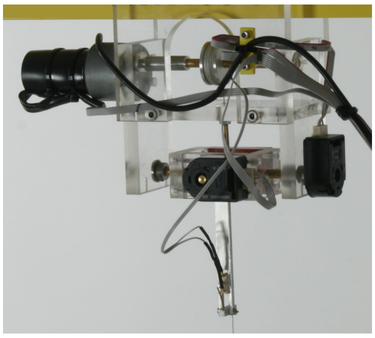


Fig. 9. Tower crane trolley

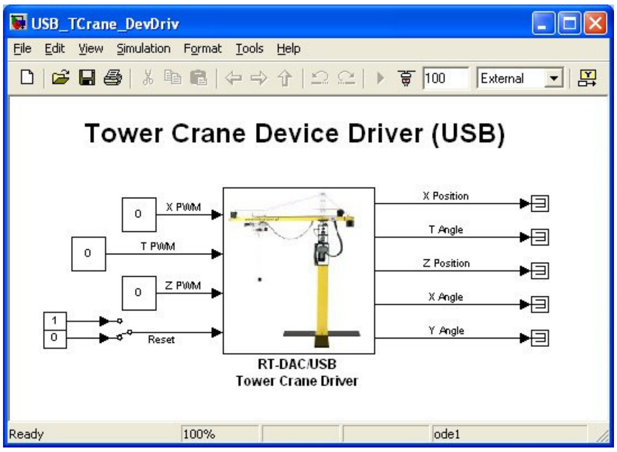


Fig. 10. Tower crane Simulink device driver

payload in the jib plane, *Y Angle* is the angle deviation of the payload in the plane, directed perpendicularly to the jib.

The laboratory model is not a copy of any existing industrial tower crane. It is a tool for research to examine phenomena that occur during motion of a suspended payload and to design control algorithms assuring a safe transport.

VI. Active Magnetic Levitation Systems (AMLS)

AMLS [14] represent structurally unstable systems where the system dynamics is adjusted by the operating controller. The performance of the controlled AMLS depend on the applied hardware and realized control strategy. A key point is to satisfy real-time conditions that allows to close the control loop with frequencies higher than 200Hz depending on the applied hardware and control tasks. Nowadays, AMLS are controlled in the digital form usually, where a number of hardware-software architectures is used, to satisfy mentioned requirements. The real-time controller processing time, computational effort and numerical representation of the processed digital system must be considered for the real-time application.

The AMLS candidates are available now in two designs: single and dual electromagnet (Fig. 11). Both of them contain the electromagnet, ferromagnetic object, position sensors, signal conditioning unit and power interface. The electromagnetic actuator is driven by PWM or current controller. In the first case the high frequency voltage signal is applied to the coil, while in the second one the coil current is kept at the desired level by the hardware current feedback. Both solutions affect the real-time control architecture. For the current driven systems the lower sampling frequency (200-400Hz) allows to control the AMLS using UBS based board [5]. The AMLS are connected to the PC via PCI based board usually equipped with the FPGA unit where the custom logic is implemented [5]. For the real-time control purposes the parallel signal processing board has been developed. The real-time control system performance can increase due to the parallel sampling. [10].

The basic principle of AMLS operation is to control an electro-magnet to keep a ferromagnetic object levitated. The object position is determined through a distance sensor or state observers. The equilibrium stage of two forces (the gravitational and electro-magnetic) has to be maintained by the controller to keep the sphere in a desired distance from the magnet. A number of real-time controllers has been developed to realize stabilization, program system dynamics (see Fig. 13), tracking and satisfy robustness.

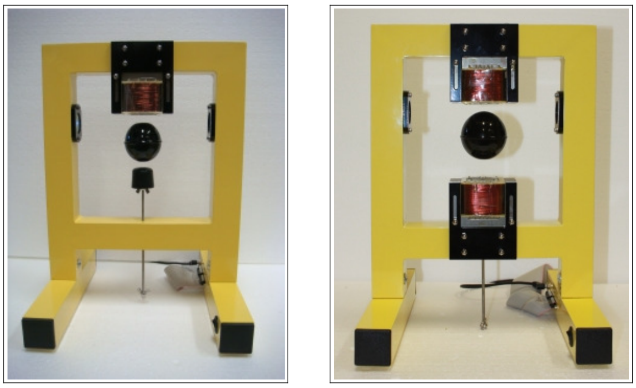


Fig. 11. AML Systems in a single and dual electromagnets versions [3].

From the real-time control point of view a time slot reserved for the application controller vary with respect to the controller architecture. The computational effort, control algorithm structure and numerical representation limit the execution time. For example a nonlinear robust fuzzy based controller [12] consumes more processing power that self tuning neural controller [11] and optimally tuned PID. The dual electromagnets AML can be used to check performance of the real-time controller. The lower electromagnet suits as an extra excitation signal (see Fig. 13) or increases the electromagnetic forces to speed up the system dynamics. The digital hard real-time realized in the form of custom FPGA embedded PID controller is mostly limited by the A/D converters speed. For the analog sampled hard-real time control the Dynamically Programmable Analog Signal Processor has been applied [13]. This solution allows to process sampled analog signals with tunable option of gains fast sampling rate up to 2MHz limited by the controller architecture.

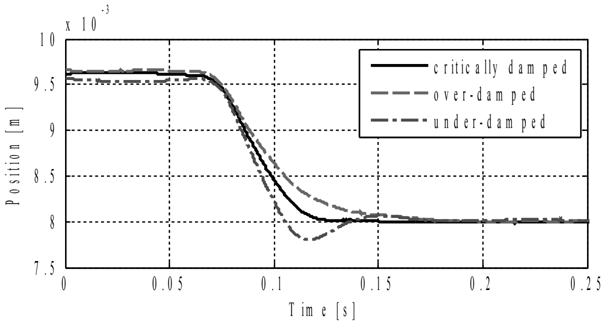


Fig. 12. Programmable dynamics of the AML System

AMLS are one of best tools to learn the control theory and real-time control due to the time-critical execution requirements of AMLS and its dynamics programming ease. Note, that limits of the real-time control systems or limited sampling frequency could make the system unstable.

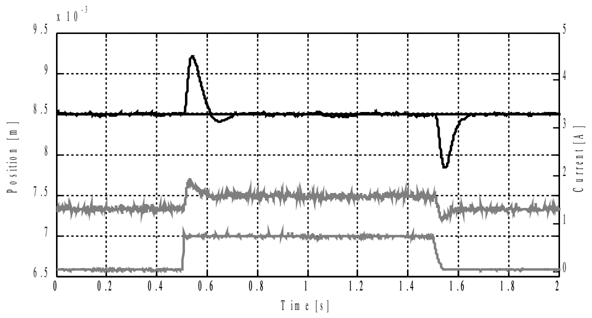


Fig. 13. Object stabilization at external square pulse excitation realized by the lower electromagnet [11]

For more information visit www.maglev.agh.edu.pl.

VII. Multitank system

The Multitank System consists of three water tanks placed above each other (Fig. 14). The uppermost tank is rectangular, the middle one is prismatic and the lowest is a quarter of a cylinder. The first tank thus has a constant cross section, while the cross sections of the two others vary with the water level. Water is pumped into the upper tank from a supply tank by a pump driven by a DC motor. The water flows out from the tanks only due to gravity. The orifices act as flow resistors. The outflow rate from each tank can be adjusted by a manual valves or proportional.

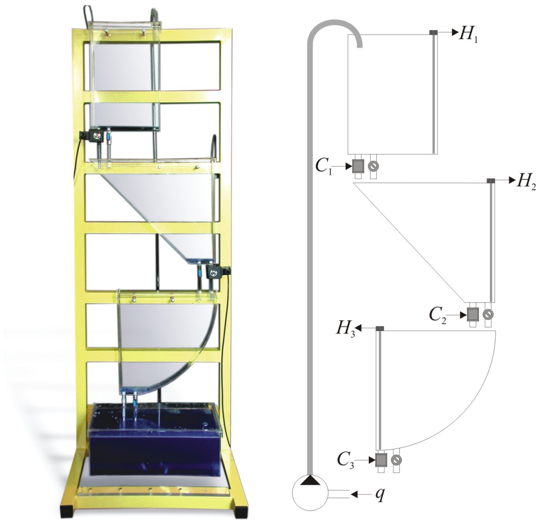


Fig. 14. General view of the tanks system

The levels in the tanks are measured with pressure transducers, which offers analog (0-10) V or digital interface (frequency signal from 100-200 kHz). The speed of the pump motor and proportional valves are controlled by PWM signals via power interface [4].

The goal of the Multitank System design is to study and verify in practice linear and nonlinear control methods. The general objective of the control is to reach and stabilise the level in the tanks (mainly the lower tank) by an adjustment of the pump operation or/and valves settings. The other control problems are: minimizing of the fluid level oscillations and stabilization of the outflow from the tank. These control problems can be solved by a number of level control strategies ranging from PID to adaptive and fuzzy logic controls [1][2].

The Multitank System has been designed to operate with following hardware control platforms:

❍ PC-based equipped with dedicated I/O board,

❍ PLC/PAC controller provided with PWM generator and analog (or high-speed counter) input modules,

❍ other FPGA/microcontroller based platform able to measuring analog or high-speed frequency signals and generating PWM signals.

Such a platform works as an external controller. The control system communicates with the level sensors, valves and pump by a dedicated I/O interface and the power interface.

The mathematical model of the process can be obtained by means of mass balance [15]:

$$\frac{dH_1}{dt}=\frac{q-C_1\sqrt{H_1}}{\beta_1(H_1)} \quad (1)$$

$$\frac{dH_2}{dt}=\frac{C_1\sqrt{H_1}-C_2\sqrt{H_2}}{\beta_2(H_2)} \quad (2)$$

$$\frac{dH_3}{dt}=\frac{C_2\sqrt{H_2}-C_3\sqrt{H_3}}{\beta_3(H_3)} \tag{3}$$

where: H_1, H_2 and H_3 are the water levels; $\beta_1(H_1)$, $\beta_2(H_2)$ and $\beta_3(H_3)$ are the cross sectional area of the tanks; C_1, C_2 and C_3 are the outflow coefficients and q is the flow into the first tank (control variable). The tank cross sectional areas are:

$$\beta_1(H_1)=a\cdot w \tag{4}$$

$$\beta_2(H_2)=w(c+b\frac{H_2}{H_{2\max}}) \tag{5}$$

$$\beta_3(H_3)=w\sqrt{R^2+(H_{3\max}-H_3)^2} \tag{6}$$

where a, b, c, w, $H_{2\max}$, $H_{3\max}$ and R are constants. The control and state components are bounded:

$$0\leqslant q(t)\leqslant 200\,cm^3/s \tag{7}$$

$$0\leqslant H_1(t),H_2(t),H_3(t)\leqslant 40\,cm \tag{8}$$

For the model (5), for fixed q=q_0 we can define an *equilibrium state (steady-state points)*, given by:

$$q_0=C_1\sqrt{H_{10}}=C_1\sqrt{H_{20}}=C_1\sqrt{H_{30}} \tag{9}$$

Figure 15 shows the equilibrium states for the real tank system.

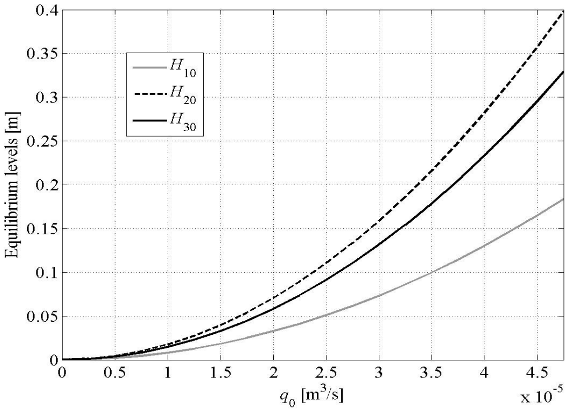


Fig. 15. Equilibrium levels for real system

Several issues have been recognised as potential difficulties for a high accuracy control of the tanks level or flow:

❍ nonlinearities caused by the tank shapes, the valve geometry and flow dynamics, the pump and valves input/output characteristic curve,

❍ state constraints, introduced by the maximum and minimum allowed levels in the tanks.

VIII. Pendulum on a cart system

A favourite laboratory system is the pendulum on a cart (see Fig. 16). We may have a lot of fan while experiment with it.

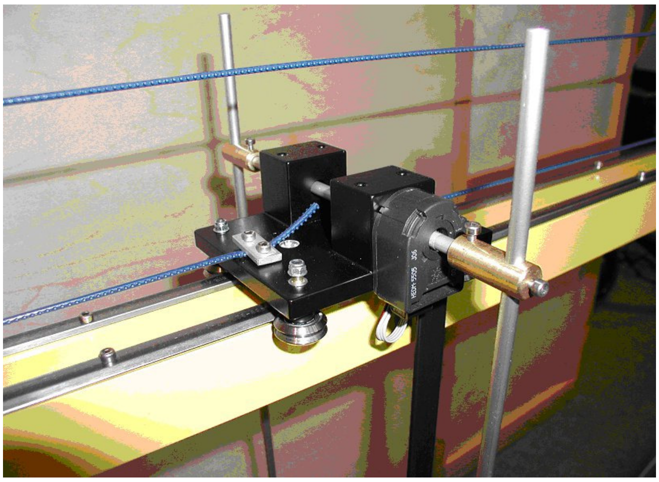


Fig.16 Pendulum on a cart system

To swing and to balance the pendulum the cart is pushed back and forth on a rail of limited length. The purpose of the control algorithm is to apply a sequence of forces of constrained magnitude to the cart such that the pendulum starts to swing with an increasing amplitude and the cart does not override the ends of the rail. The pole is swung up to achieve a vicinity of its upright position. Once this has been accomplish, the controller is maintaining the pole vertical and is bringing the cart back to the center of the rail. The pendulum in its upright position behaves as a circus acrobat. We have to be aware also that this non-trivial fourth order, unstable mechanical system can be used to conduct very serious research corresponding to the complex time-optimal control algorithm. To facilitate education does not mean to give a facile example. Hence, a short presentation of time-optimal and rule-based controls is shown.

The time-optimal controller requires analysis and synthesis based on a mathematical model [16][17]. The approach is laborious and time consuming. However, the good quality of control can be a reward. The so called canonical equations – state (forward) and conjugate (backward) – are solved. A variable parameter optimization method is used. The horizon, a number of switchings and the sign of the first "bang-bang" control are the variables of the quadratic performance index. A final snapshots of the state trajectories, control and anti-gradient of the performance index corresponding to the numerical optimization are shown in Fig. 17. The numbers correspond to the following variables: 1 → x_1 is the cart position with respect to the rail center, 2 → x_2 is the angle between the vertical upright direction and a current angular position of the pendulum, 3 → x_3 is the cart velocity, 4 → x_4 is the pendulum angular velocity, 5 → u is the control force acting horizontally on the cart and 6 → ψ is the anti-gradient of the performance index.

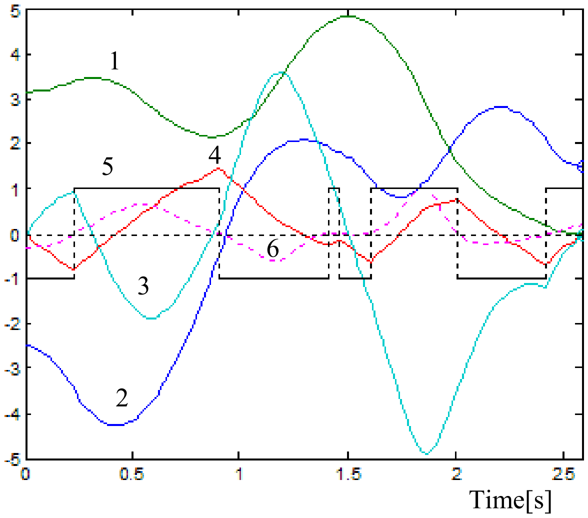


Fig. 17 The state trajectories, control and anti-gradient of the performance index at the end of the numerical optimization

The rule-based controller in a simple form is shown below.

```
                    Stabilization
If  $|x_2|-S<0$   Is pendulum in the stabilization zone?
   then  $u_r=K_1(x_1-x_1^f)+K_2x_2+K_3x_3+K_4x_4$
              Calculate an auxiliary linear control $u_r$.
    if  $|u_r|+F_s>u_{max}^{STAB}$   Does the auxiliary control plus
              friction exceed the limit?
    then  $u=u_{max}^{STAB}\operatorname{sign}u_r$   Calculate the ultimate control
                  that attains the limits.
    else  $u=u_r+F_s\operatorname{sign}u_r$   Calculate the ultimate linear
                  control contained in the limits.
   end
          Enlarging magnitude of the pendulum
elseif
   $\frac{1}{2}x_4^2+9.81\cdot 3.2(\cos x_2-1)>0$   Is pendulum kinetic energy
                  larger then potential energy?
  then u = 0   Set control to zero – perform soft landing in
               the stabilization zone.
  else
     $u=-u_{max}^{POST}\operatorname{sign}\left[x_4\left(|x_2|-\frac{\pi}{2}\right)\right]$   Apply the „bang-bang”
                  control to enlarge oscillations.
end
```

where: $K_{1,} K_{2,} K_{3,} K_4$ are constant gains, x_1^f is the final cart position, F_s is the static friction force and u_r is the auxiliary control.

Alas, rule-based control cannot predict the time-optimal strategy. In turn the time optimal control is sensitive to disturbances and without the soft landing (the control is set to zero when there is an excess of kinetic energy of the pendulum) it would be difficult to achieve the control goal.

IX. Conclusions

There are several critical terms to become an effective educator in the *Automatic Control* field. One must has access to facilities generally called mechatronic systems. Algorithms constructed on the basis of mathematical simulation models should be used in the same software environment to control the actual devices. It is a known complication of such a transfer, namely the real-time control regime. This is the biggest challenge in the education of engineers. It is satisfied mainly due to parallel (FPGAs) and software extensions.

X. References

[1] Cheung Tak-Fal, Luyben W.L., “Liquid Level Control in Single Tanks and Cascade of Tanks with Proportional-Only and Proportional-Integral Feedback Controllers”, *Ind. Eng. Chem. Fundamentals*, vol. 18, No. 1, 1979, pp. 15-21.

[2] Galichet S., Foulloy L., “Fuzzy Logic Control of a Floating Level in a Refinery Tank”, Proc. Of *3rd IEEE Int. Conference on Fuzzy Systems*, Orlando, June 1994, pp. 1538-1542.

[3] InTeCo Ltd., *Magnetic Levitation System*, Users Guide, Inteco 2003

[4] InTeCo Ltd., Multitank System. User's Manual, Inteco, 2010.

[5] InTeCo Ltd., *RT-DAC4/PCI User's Manual*, Inteco, 2005.

[6] Pauluk M. “Robust control of 3D crane”, MMAR 2002 : proceedings of the *8th IEEE international conference on Methods and Models in Automation and Robotics* : 2–5 September 2002 Szczecin. Wydawnictwo Uczelniane Politechniki Szczecińskiej.

[7] Pauluk M., A. Korytowski, A. Turnau and M. Szymkat (2001): *Time Optimal Control of 3D Crane*, "Methods and Models in Automation and Robotics" - proceedings of the 7th IEEE International Conference - MMAR 2001, pp. 927-932, 28 - 31 August 2001 Międzyzdroje, Poland

[8] Piątek P., W. Grega, „Seed analysis of a digital controller in time critica+l applications”, *Journal of Automation, Mobile Robotics & Intelligent Systems (JAMRIS)* vol. 3 No 1, 2009, p.57-61. ISSN 1897-8649

[9] Piątek, P. *Application of Specialized Hardware Architectures for Realization of Time-critical Control Tasks* (in Polish), Grega W. - supervisor. Ph.D. thesis, AGH University of Science and Technology, Department of Automatics, Poland 2007.

[10] Piłat A., Piątek P.: *Multichannel control & measurement board with parallel data processing*, Recent advances in control and automation Warsaw : Academic Publishing House EXIT, The Committee on Automatic Control and Robotics of the Polish Academy of Sciences, pp. 373÷380

[11] Piłat A., Turnau A.: “Neural adapted controller learned on-line in real-time”, *14th International conference on Methods and Models in Automation and Robotics* : 19–21 August 2009, Międzyzdroje, Poland

[12] Piłat A., Turnau A.: “Self-organizing fuzzy controller for magnetic levitation system”, *Computer Methods and Systems*, 14–16 November, 2005, Kraków, Poland, pp. 101÷ 106

[13] Piłat A.: “Programmable analog hardware for control systems exampled by magnetic suspension”, *Computer Methods and Systems*, 14–16 November 2005, Kraków, Poland, pp. 143÷ 148

[14] Piłat, A.: *Control of magnetically levitated systems*. Ph.D. Dissertation (in Polish), AGH University of Science and Technology, 2002, Kraków, Poland

[15] Rosół M., *Control of Nonlinear Liquid Flow Process*, PhD Thesis – in Polish, (Grega W. supervisor), AGH University of Science and Technology, Department of Control, Krakow, 2001.

[16] Szymkat M., Korytowski A., Turnau A.: “Extended Variable Parameterization Method for Optimal Control”. Proc. 2002 *IEEE Int. Symposium. on Computer Aided Control System Design*, Glasgow, Scotland, U.K. September 18-20 2002, pp. 175-180.

[17] Turnau A. “Model identification dedicated to the time-optimal control.” *17 IFAC World Congress*, 7-11 July, Seoul, 2008, pp. 2655-2660.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Methods of Computer-Assisted Manual Control of Wheeled Robots

Viktor Michna, Petr Wagner, Jiri Kotzian
VŠB – Technical University of Ostrava
Department of Measurement and Control, Faculty of Electrical Engineering and Computer Science
17. listopadu 15, Ostrava-Poruba 708 33, Czech Republic
Email: {viktor.michna, petr.wagner, jiri.kotzian}@vsb.cz

Abstract*—This paper deals with possibilities of manual control of wheeled robots. The problem is tested on a robot-soccer application. The manual computer-assisted control module creates an interface between game controller and controlled robot. There are several ways of steering implementation. The simplest is a differential steering. The module currently uses semi-automatic steering which provides a capability to move with the robot as a point. This functionality requires interaction with vision system, usage of Kalman or similar filter and compensators. Next logical step is a fully-assisted steering. This steering system interacts with strategy and motion control module and provides additional functionality such as tracking the ball and shot. The objective is the control of many robots by only one player.

I. Introduction

The robot-soccer is a popular application that has ability to attract young people and to popularize technical branches on the faculties of mechanical engineering, electrical engineering and computer science. It is also used as a test bed to evaluation and testing of new methods, algorithms and solutions in different technical branches.

Current solution of robot-soccer application consists of the Vision module, the Strategy module (divided into Higher strategy and Motion control), the Wireless transceiver and the Communication module. The Communication module provides interface for distributed data exchange among the other modules of the system [1].

This paper describes the development of computer-assisted Manual control module as the part of the 3rd generation of robot-soccer strategy system for VŠB-TUO robot-soccer team. The module has been developed for two reasons – strategy testing against human opponents and for attraction of young people to study at technical universities.

II. Manual control module interface

The manual control module allows a player to control the robots using connected game controllers or joysticks. It cooperates with other robot-soccer modules so it makes an interface between the game controller and controlled robot.

A. Wheeled robot

The robot with basic parameters is in the Fig. 1. The relationship between velocities, angular velocity and radius of rotation is defined [2]

$$v=\frac{vl+vr}{2}\ ,\ \ \omega=\frac{v}{r}=\frac{vr-vl}{d} \qquad (1)$$

B. Game controller

After testing several types of controllers, the gamepad was chosen. It allows quick control of associated values.

The selected gamepad (Fig. 2) has

- 2 analog sticks
- 8-way POV (point of view) hat
- 12 buttons.

These controls can be utilized to comfortably support both left-hand and right-hand mode.

The position of main stick has coordinates $jx, jy \in \langle -1;1 \rangle$ in axis x, y respectively and button values $b_0, b_1, .. b_N \in \{0;1\}$.

III. Analysis of control methods

There are several ways of how manually control the robot – differential steering, semi-automatic steering, fully-assisted steering and group-assisted steering, from the simplest one to the most advanced.

A. Differential steering

The first generation of manual control used the simple differential steering. Its principle is based on direct

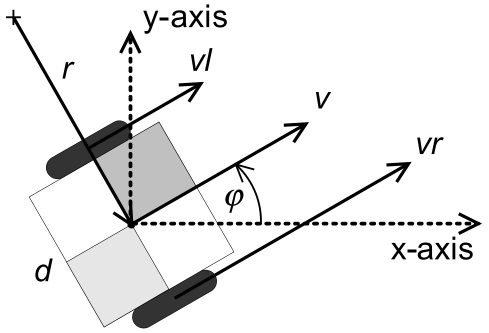


Fig 1. Basic motion parameters of the robot.

This work is being solved at the VSB-Technical University of Ostrava, Czech Republic. The project is supported by GAČR project 102/08/1429 and also by the internal grant SP/2010165 „Development and using new methods for design and realization of embedded systems".

Fig 2. Gamepad controller.

conversion of desired velocity v_d and angular velocity ω_d of robot into the movement of stick

$$jx=\frac{\omega_d}{\omega_{max}}\,,\ \ jy=\frac{v_d}{v_{max}} \qquad (2)$$

Than, the velocities of robot's left and right wheel are defined

$$vl=jy\cdot v_{max}-jx\cdot(\omega_{max}\cdot d)/2$$
$$vl=jy\cdot v_{max}+jx\cdot(\omega_{max}\cdot d)/2 \qquad (3)$$

This principle is shown in the Fig. 3. The computer-assisted control (CAC) block in this case is very simple and the control loop is opened. It leads to very difficult manual control of the robot to move it into desired position or orientation. Human is unable to both accurately and quickly control the rotation of the robot and to adjust its relative position. That is why this steering is unsatisfactory.

B. Semi-automatic steering

Current solution of the computer-assisted manual control is based on automatic control of the robot according to the desired direction and speed. This requires closer cooperation with the vision system. See the Fig. 4. The vision system supplies a position of the robot *rx*, *ry* in axis x, y of the playground and angular position *rφ*.

The objective of this control is to allow moving with the robot as it was a 2D point

$$jx=\frac{vx_d}{v_{max}}\,,\ \ jy=\frac{vy_d}{v_{max}} \qquad (4)$$

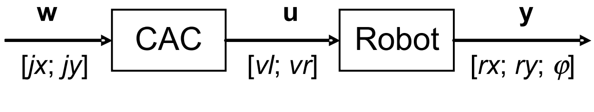


Fig 3. Schema of differential steering.

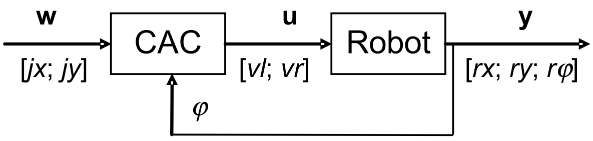


Fig 4. Schema of semi-automatic steering without filter.

where vx_d, vy_d are desired speeds along the axis x and y. There are two possible configurations, without a filter (Fig. 4) and with a filter (Fig. 5). The filter reconstructs the precise position and the speed of the robot. It also allows the CAC to compensate the error from the trajectory due to rotation of the robot. Currently, there is used an extented Kalman filter [1][3].

The schema of the CAC block is in the Fig. 6. Suppose that a player wants to move the robot to given direction with certain speed.

If the robot does not move it should first rotate around its vertical axis to a desired direction and then it should start moving with a desired speed. This will ensure following of a desired trajectory. This function is implemented in the Stationary rotation block. If actual velocity *v* of the robot is smaller than predefined minimum velocity v_0 and absolute value of angle error $\Delta\varphi$ is greater than a predefined maximum angle error $\Delta\varphi_0$ then desired speed v_d is zeroed. Simulated trajectories are shown in the Fig. 7.

If the robot is already moving, it is not possible to change the desired direction and to maintain the planned trajectory.

First, the robot has to rotate to the desired direction while it is still moving. Then it can gradually approach the trajectory that it would move as a mass point from the time of direction change in an ideal case. The tracking of ideal

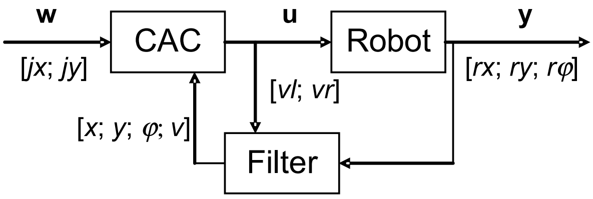


Fig 5. Schema of semi-automatic steering with filter.

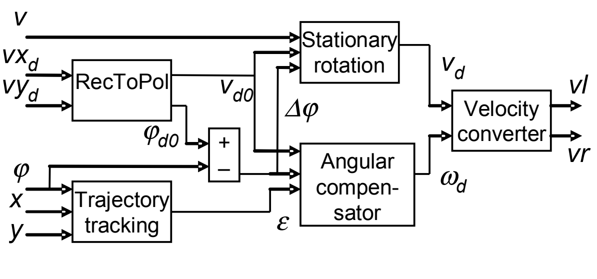


Fig 6. Schema of CAC for semi-automatic steering.

mass point is encapsulated in the Trajectory tracking block. The control requirements – direction (angle error $\Delta\varphi$) and position control (parameter ε) act against each other and this must be reflected by compensators design including system non-linearity and bounds of centripetal acceleration. The compensators are placed in the Angular compensator block. The Velocity converter block converts the pair of desired velocity v_d and angular velocity ω_d into the desired velocities *vl*, *vr* of left and right wheel. Simulated trajectories are shown in the Fig. 8.

If the robot is standing or moving, it is possible to invert the direction of robot. This reduces the required angle of rotation. This functionality is not shown in Fig. 6. If the robot is moving, the range of desired directions suitable for the inversion of robot direction is significantly lower. This is because gradual approach to the desired trajectory is more time efficient than stopping the robot and turning the direction of motion by 180 degrees.

C. *Fully-assisted steering*

This steering is next logical step in evolution of manual control module. See the Fig. 9. It requires cooperation with the Higher strategy and the Motion control module. The higher strategy controls all robots except the manually controlled ones and instructs them the desired positions and velocities [4]. The CAC block overrides the position and velocity for manually controlled robots. The motion control plans trajectory in real-time and controls position of the robots by changing the desired wheel velocity [5].

There are three modes of operation

- Free movement
- Tracking the ball
- Shot.

The free movement mode allows similar movement of the robot as semi-automatic steering. The difference is in this mode the robot will cooperate with other robots and avoid collision with them.

The tracking ball mode causes the robot to start catching the ball. It is activated by pressing the button b_0. If the stick is also applied, it controls the direction in which to meet the ball.

The shot mode makes the robot to shoot to the direction specified by stick.

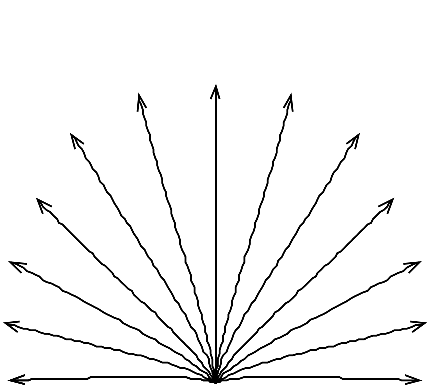


Fig 7. Robot trajectories for particular desired directions with step of 15 degrees and zero speed in the beginning.

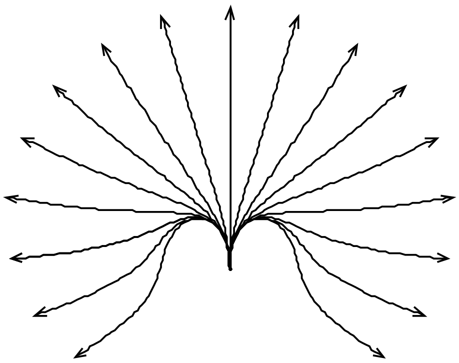


Fig 8. Robot trajectories for particular desired directions with step of 15 degrees and constant velocity.

D. *Group-assisted steering*

This steering will be the final extension of fully-assisted steering. The basic principle is same as previous except a fact that one player is controlling all of the robots.

Only one robot is controlled at the moment, so the capability of robot switching has to be developed. There are several possibilities of making such capability

- The robot nearest to the ball – the robot is selected automatically or by a button.
- Switching to a next robot – selection of the next robot in sequence.
- Select robot by POV-hat or stick – the robot is selected by analysis of current positions of robots on the playground.

It may be difficult for player to quickly recognize the selected robot on the playground. This can be solved by

- External lighting of the robot by LEDs.

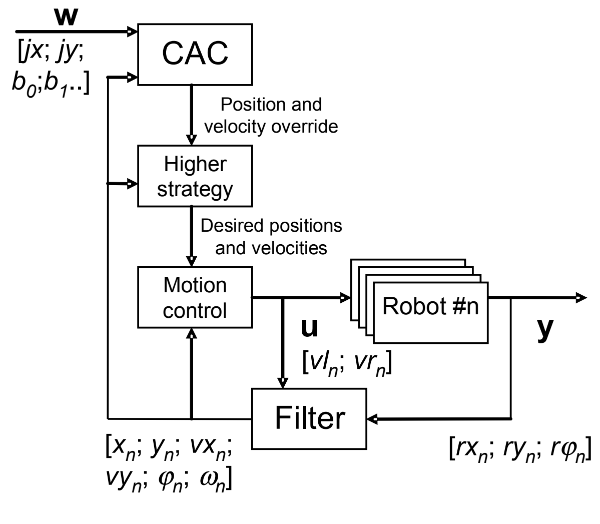

w
$[jx; jy; b_0;b_1..]$
CAC
Position and velocity override
Higher strategy
Desired positions and velocities
Motion control
Robot #n
u
$[vl_n; vr_n]$
y
Filter
$[x_n; y_n; vx_n; vy_n; \varphi_n; \omega_n]$
$[rx_n; ry_n; r\varphi_n]$

Fig 9. Schema of fully-assisted steering.

- Only virtual selection of the robot on soccer monitor.

IV. Additional functions

There are several additional requirements on the computer-assisted manual control. Some of them are general extensions, others are important for concrete type of steering.

A. Collision Detection

In the robot motion control, a collision detection is necessary. When a collision occurs, it leads not only to stagnation of the robot but it also causes that the underlying mathematical model of the robot gives false output. The model is part of the filter block in the motion control module and provides the predicted position and the speed of the robot based on control requirements. The information about collision can be used to the resetting of the model.

Another problem occurs if there is a permanent obstacle in the way. Then the described motion control is not able to react properly and rotate the robot. The solution is to stop the robot until it rotates and gets out of the collision.

Collision detection is performed in two ways

- A priori – detection is made on the basis of the position of the robot to known bounds (of the playground) and the direction of the velocity vector,
- A posteriori – detection is made on the basis of significant difference between the predicted robot position and its actual position.

B. Game controls

In addition to the control of the direction of the robot in two axes, there are more features important for the strategy testing. Some of these functions apply only on semi-automatic steering

- Acceleration – the possibility of a short-time increase of robot speed, this function is useful for shooting the ball.
- Brake – allows rotation of the robot in place.
- Tracking the ball – robot automatically targets towards the ball, allowing its easy hitting.

C. Force feedback

When the robot gets stuck, typically due to an impact of the robot or an attempt to pass through the barrier, the game controller vibrates strongly for a brief moment.

V. Conclusion

In this paper we described the Manual control module of the robot-soccer system. This module provides interface between the game controller and robot. Currently there is used the principle of semi-automatic steering, which is the step between the simple differential steering and the fully-assisted control method. The manual control module also uses additional functions as the collision detection and force feedback. Next research will focus on design of fully-assisted control. The requirements for the upcoming CAC methods were also described.

References

[1] P. Wagner, *et al.*, “Control of Group of Robots”, In *9th RoEduNet International Conference*, to be published, 2010.

[2] J. Kotzian, Z. Machacek, *et al.*, “Embedded control system for the mobile robot”, *WSEAS Transactions on Systems*, pp. 2261-2268, 2005.

[3] L. Jetto, S. Longhi, “Development and Experimental Validation of an Adaptive Extended Kalman Filter for the Localization of Mobile Robots”, *IEEE Transactions on Robotics and Automation*, Vol.15, pp. 219-229, 1999.

[4] V. Srovnal, B. Horak, *et al.*, “Strategy Description for Mobile Embedded Control Systems Exploiting the Multi-agent Technology”, *Lecture Notes in Computer Science*, pp. 936-943, 2007.

[5] G. Novak, M. Seyr, “Simple Path Planning Algorithm for Two-Wheeled Differentially Driven (2WDD) Soccer Robots”, *WISES 2004*, 2004.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Software and hardware in the loop component for an IEC 61850 Co-Simulation platform

Mohamad Haffar
Euro System, GIPSA-lab
(UMR 5216)
Grenoble-France
Email: mohamad.haffar@euro-system.fr

Jean Marc Thiriet
GIPSA-lab (UMR 5216), UJF
Grenoble-France
Email: jean-marc.thiriet@ujf-grenoble.fr

Mohamad El-Nachar
Université Libanaise Faculté Génie
Branche1, GIPSA-lab
Tripoli-Lebanon, Grenoble-France
Email: mohammad.el-nachar@gipsa-lab.grenoble-inp.fr

Abstract— **The deployment of IEC61850 standard in the world of substation automation system brings to the use of specific strategies for architecture testing. To validate IEC61850 architecture, the first step consists in validating the conformity of the object modeling and services implementation inside devices. The second step consists in validating IEC61850 applications compliance according to the project specifications. A part of the architecture can of course be tested "physically"; however in the design phase or when the actual architecture cannot be checked directly, modeling is helpful. In our research study we propose a co-simulation approach based on several components allowing the realization of advanced tests. This paper describes the need and the design implementation of software and hardware in the loop components as well as the object modeling concept of IED models**.

I. List of Abbreviation

The following table 1 contains the definition of acronyms used in this paper.

Table I.
Abbreviation List

Acronym	Definition
IED	Intelligent Electronic Device
SAS	Substation Automation System
FAT	Factory Acceptance Test
HITL	Hardware In The Loop
SITL	Software In The Loop
UCAIUG	Utility Communications Architecture International Users Group
GOOSE	Generic Object Oriented Substation Event
MICS	Model Implementation Conformance Statement
PICS	Protocol Implementation Conformance Statement
ICD	IED Configuration Description
MX	Measurand analogue value X
SCL	Substation Configuration language
RTG	Run Time Generation

II. Introduction

In the early 1990s communication technologies and protocols began to appear in SAS application. In 2004, a new worldwide standard of communication IEC61850 was introduced in the majority of substation automation systems carrying out new innovation prospects. The interoperability between devices from different manufacturers as well as the insurance of the overall security of SAS architecture through its communication network becomes now attainable with this new communication standard [1]. In addition, object data modeling is used inside IEC61850 equipments to replace the aspect of non significant addresses. The standard provides a stack of services in order to obtain a self description of the internal objects and communication capabilities of the device; this leads to reduce the time of developing supervision control and data acquisition applications.

IEC61850 authorizes the distribution of functionalities between devices. This aspect permits the exchange of real time inter-equipment messages over the communication network to provide advanced services for electrical substation process which was done previously via hardwiring (e.g. interlocking, load shedding, auto transfer switch, etc...). The real advantage behind consists in reducing the amount of cables inside a substation which can be very long due to the distance between the substations (i.e. installation price). However this new aspect of configuration relies on a peer to peer device messages sent over the communication network. Thus it must be well verified before any implementation [2].

Being the unique standard used in the SAS network, devices from different manufacturers are be able to dialog in order to exchange information and services. This feature brings to the engineering staff more liberty in choosing their IED (e.g. protection relay, power meter …). However, the heterogeneity of IEDs inside SAS architecture yields complexities to the FAT which becomes difficult to realize due to the dispersed location of the switchboards suppliers. An exhaustive FAT must be provided with each project development in order to minimize time of site commissioning. This point is an essential key demanded by the end users.

End Users must have a realistic view about the demanded SAS architecture tests according to their desired requirements. Thus they should comprehend the limitation scope of the conformance tests provided by the international certification organizations such as Kema [3] versus a complete successful implementation of IEC 61850 SAS architecture which should be well coordinated by all involving parties.

This work was supported by Euro System Company and GIPSA Laboratory

III. Limitations Scope of Standard Certification Tests

A generic testing plan for an IEC61850 SAS architecture shall include four testing categories: the conformance testing of a device, the distributed functionalities testing, the interoperability testing and the global system performance evaluation [4]. Witness and hold points shall be specified in the testing plan for carrying out inspections that verify the quality of tests.

UCAIUG has established a conformance testing program for IEDs which permits the certification of the internal object data modeling, communication oriented services and the global configuration online testing.

These conformance tests could be considered as parts of the device conformance testing. Therefore distributed functionalities, interoperability and global system performance evaluation tests are still left unanswered. Guidelines are given by the international certification organization to accomplish these advanced tests.

New tools and advanced models should be provided to fill the gap of tests. These tools should also permit the cohabitation of real devices of a switchboard supplier with virtual device of the missing switchboard to ensure an advanced FAT in a heterogeneous SAS architecture.

Based on the guidelines of advanced tests proposed by international certification organization, our work deals with a Co-Simulation Platform to accomplish a complete IEC 61850 advanced test. This platform is constituted of a simulation software and other additional components. Virtual IEDs will be modeled inside the simulation platform by implementing real communication IEC61850 services and data objects. The additional components are used in order to ensure the overall utility of the simulation platform. As shown in Fig. 1 additional component are divided into three categories: the standard simulation configuration component, external process interface component and Hardware in the loop component.

The major purpose of the Standard simulation configuration is to generate and configure any IED model by importing its device configuration file. Using this component our simulation platform will be able to simulate an heterogeneous architecture containing IEDs from different suppliers.

In this paper we show the purpose and the implementation detail of the external process interface and hardware in the loop component.

IV. OPNET Network Simulator

Modeling and simulating a networked system requires a network simulator that takes into account all major contributions including protocols, medium access control technology, network load, communication imperfections, timeouts and packet losses etc. Popular network simulators include NS2, OMNET, Qualnet, OpNet etc. OpNet Modeler is chosen in our research because it implements a large number of libraries of standard equipment (e.g. switches and link models) as well as proprietary models, which is useful for simulation and reduces the development time [5]. Object oriented modeling approach fitted into many programming editors is used in OpNet in order to facilitate the construction of models (Fig. 2).

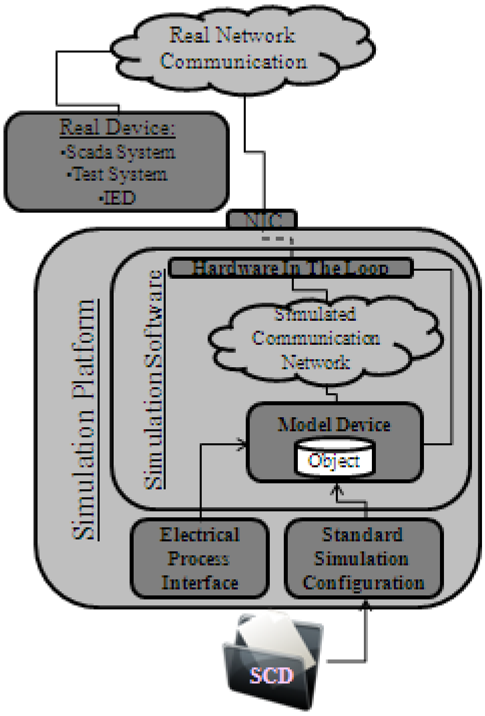


Fig. 1 Co-Simulation platform architecture

A network device is modeled as a node which is composed of many modules connected by packet streams or static wires. Each module aims to represent specific aspect of the node's behavior such as data creation, data storage, data processing or routing (IP layer), data transmission (Transport layer), etc. Thus the modeling of each layer of the TCP IP stack is made in OpNet modules.

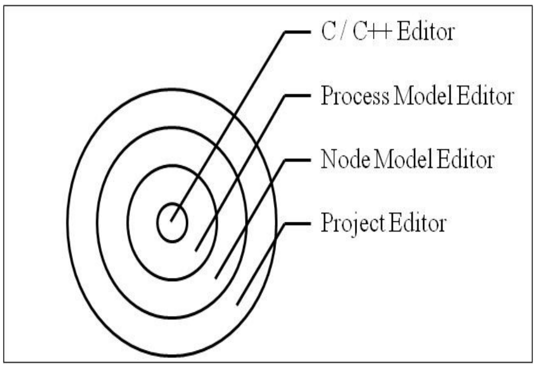


Fig. 2 Simulation software editors

The process is programmed using the finite state machine (FSM – Technology). This approach facilitates the supporting of the implementation of applications, algorithms and queuing policies. States and transitions graphically define the progression of a process in response to events. Each state in process model contains embedded C/C++ code, supported by an extensive library of functions designed for network programming.

Two kinds of states are used in OpNet Modeler: red states and green states.

A red state in the process model is an unforced state. Unforced states allow a pause between the enter executives and exit executives, and thus can model true states of a system. After a process has completed the enter executives of an unforced state, it blocks and returns control to the previous context that invoked it. When an interrupt occurs and invokes the process, the exit executive will be treated as well as the test of the transition to the next state.

A green state in the process model is a forced state. Forced states are so called because they do not allow the process to wait. In other words, the exit executives of a forced state are executed by a process immediately upon completion of the enter executives.

V. Co-simulation Components

This paragraph shows the importance of the presence of the external process interface and the hardware in the loop components inside our Co-Simulation platform in different phase of IEC61850 project.

A. *Typical architecture of an IEC 61850 switchboard under test*

Each IEC61850 server (i.e. device or model) is delivered with its Model Implementation Conformance Statement 'MICS' and Protocol Implementation Conformance Statement 'PICS'. MICS documentation describes the capabilities of the device in term of functionalities described in object format. However PICS is another documentation which describes the IEC61850 services supported by the device. The conformance tests of a device [6] is based on tests that cover the conformity of object data modeling as well as the conformity to IEC 61850 Abstract Communication Services Interfaces 'ACSI' according to PICS and MICS documentations.

Three types of services can be found inside each IEC61850 server:

- Client/Server pooling services which are initiated by the IEC61850 client.
- Peer To Peer services (via IEC61850 GOOSE message). These services are generated by the server on event. It provides the sharing of functionalities between devices.
- Reporting services which are heavy load generated periodically or on event by the server. This kind of services permits to decrease the network load by reducing the number of pooling services.

It is shown in Fig. 3 an image of a typical IEC61850 future substation. The substation is divided into three switchboards including each a group of IEDs coming from different device supplier. The switchboard S1 is the one under test inside S1 factory placement. The switchboard S2 shares functionalities with the switchboard S1. This switchboard is considered absent in our architecture for the fact that it comes from another device supplier. The last switchboard S3 doesn't share any functionality with the other switchboards but is connected to the same communication network. IEDs of the switchboard S3 are also missed during FAT of S1 thus it can be considered that all physical IEDs that belong to S2 and S3 will be replaced by their models inside the simulation platform.

In order to validate the functionalities of the switchboard S1 two different kinds of tests should be provided:

- Test of the functional part of the switchboard's IED,
- Test of the communication part of the switchboard's IED.

Functional part aims to test all the protection and control parameters of IED inside S1. This is done by connecting an electrical process board to S1 and injecting the process values on the inputs of its IEDs.

The most critical point in the communication test consists to verify the exchanging of peer to peer messages between switchboards that shares functionalities. Therefore three different tests should be provided:

- Testing the publisher part of the peer to peer messages from the switchboard S1 to S2 according to the configuration of S1 IEDs.
- Testing the subscriber part of peer to peer messages from S2 to S1 which consists to verify that the IEDs inside S1 accept the peer to peer messages coming from the S2.
- Verifying that the end to end time of the peer to peer message doesn't exceed the time specified in the specification of the project.

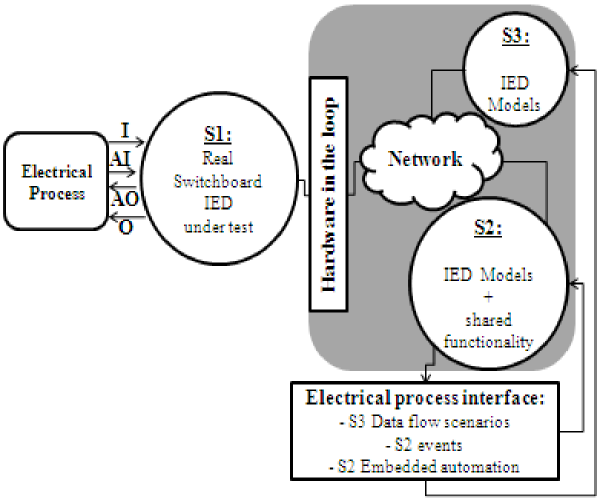


Fig. 3 Typical IEC61850 architecture under test

B. *Hardware and software in the loop during FAT*

To test the publisher part of S1, an electrical process board must inject real inputs to the IEDs inside S1 in order to publish the peer to peer messages. Hardware in the loop component is developed in our study in order to route the real network message from the real world into the simulation platform. Once arrived to the simulation world, the peer to peer message enters the simulation network and reaches at the end to the virtual IED subscribing to this message.

In the factory acceptance test stage, a punctual study of the end to end time parameter of the peer to peer messages must be given for the reason that these messages are used to solve time critical electrical applications. The dynamicity of this parameter comes from the network delays due to the data flows sharing the same network. Real IEC61850 data flows scenarios must be generated as configured inside the IEDs of S1, S2, and S3. Therefore it is crucial to have an electrical process interface inside the simulation platform which corresponds to the electrical process board inside the real world. This component is connected to the simulation platform in order to produce real IEC61850 data flow scenarios for S3 and generate planning events to test the peer to peer message sent from S2 to the switchboard under test S1 (Fig. 3).

C. Hardware and software in the loop during platform development

Our Co-Simulation platform is based on the modeling of device communications. The increase of confidence in the platform is an important point in our research study. To accomplish this point, compliance tests should be provided to IEDs models according to the IEC 61850 flowchart tests. Thus, in order to test the compliance of IED services according to PICS documentation, the software in the loop and hardware in the loop component must be used with the simulation platform. The software in the loop will be used in this case in order to generate Inputs/ Outputs simulated process value (e.g. voltages and currents exceeding values, status trip and default) to the IED model and check its behavior (i.e. reports and peer to peer message transmission). However the hardware in the loop component will be used in order to connect the model under test to a test external system to validate the services declared in its PICS documentation.

VI. Hardware in the Loop Module Implementation

A HITL node is based on an edge Router interface which is used inside the simulation platform in order to provide the workflow for exchanging packets between simulated and real devices. The detailed of this workflow is shown in Fig. 4.

The first important job of the edge interface router is the mapping of external network interface of the simulation machine to the network interface of the appropriate device model. HITL allows multiple interfaces to map different network addresses in the simulated network.

The HITL module has many associated attributes that permit to describe the HITL behavior. One of these attribute is used for filtering packets received by HITL in order to reject unexpected packets and allocate memory for expected real world packets. If the packet is accepted, the edge router enter the creation phase where it constructs the skeleton of the simulated or real packet weather if the packet received is real or simulated.

To succeed in the phase of copying the contents of packet fields between the two formats, HITL requires a predefined well-known protocol format included in the Standard Model Library. Otherwise, if the packet received is not a part of the library, a custom code must be written for providing the conversion of the unsupported protocol format packets else it will be dropped by the edge router and will not enter the simulation world. The protocols format packets supported or partially supported by HITL are the ones associated to standard TCP/IP layers as shown in Fig. 5.

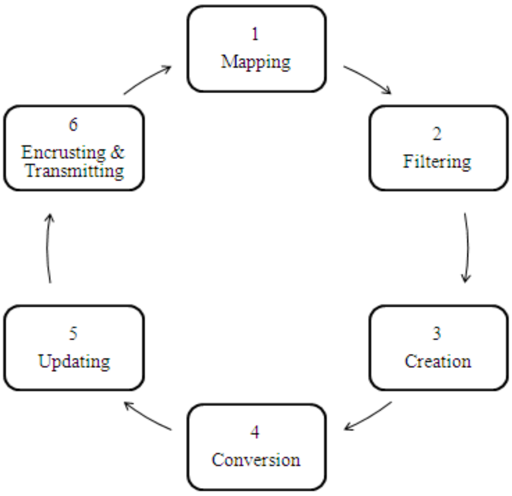


Fig. 4 Hardware in the loop workflow

As described in the introduction, the Simulation Platform is used for solving Power Management System ‘PMS’ issues, thus HITL must provide conversion of protocols associated to PMS more particularly the IEC 61850 standard. IEC 61850 packets format and other industrial protocols are not included inside the Standard Model Library Packets thus the conversion of application layer packets are not provided with a nominal HITL.

Our customized user code is developed to provide the conversion of protocol application packets. In addition, the code is also used to fill in uncompleted conversion points of transport packets by updating (i.e. phase 5 of the HITL workflow shown in figure 6) TCP missing fields which are associated to the application (e.g. data length field).

A surplus point of our custom code is that it has been developed in a way to make use of HITL with any industrial hardware device taking into consideration that it is possible to have heterogeneous power management architecture combining the IEC 61850 and other industrial protocols such as Modbus or others.

After a successful conversion and updating of partially supported packets format, the HITL encrusts real or simulated packet into a SITL conversion block and routes it finally to the appropriate model if the original packet is real or to the external world if the original packet is simulated.

VII. Software in the loop module

A. OPNET Problematic

In OpNet modeler, C functions can be placed in three different placements:

- Process model states,

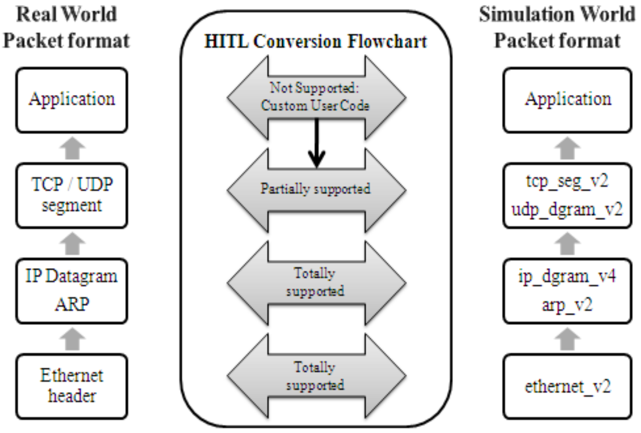


Fig. 5 Hardware in the loop conversion supported packets

- Function blocks attached to a process model,
- External OpNet Library.

When a function is defined inside a function block, it can only be used by the process associated to the function block. External OpNet libraries differ from the function block because when a C function is defined in an external library it can be shared with different process models. Finally if the C function is defined inside a state, only the state can have access to this function.

Three types of variables can be found for particular use inside the OpNet process model: the state, the temporary and the global variable. The table 2 shows which kind of functions can have access to these variables.

Table II.
Variables visibility inside simulation

		Visibility			
		Opnet State executive	Opnet Function block	Opnet External Library	External application
Type of variables	Temporary	X			
	State	X	X		
	Global	X	X	X	

What should be concluded is that only OpNet functions can have access to the simulation variable environment. Electrical process interface is considered as an external application to the simulation software. Therefore as shown in the table 1 this application will not be able to exchange variable values with the simulation on run time. For this reason a software in the loop package [7] is developed in this study in order to make this point feasible.

B. *Software in the loop package constitution*

Software in the loop consists on a package associating three different modules: C predefined research library, Data base, electrical process software. The purpose of this package is to provide an access method which authorizes the electrical process interface to exchange information with the simulation variables.

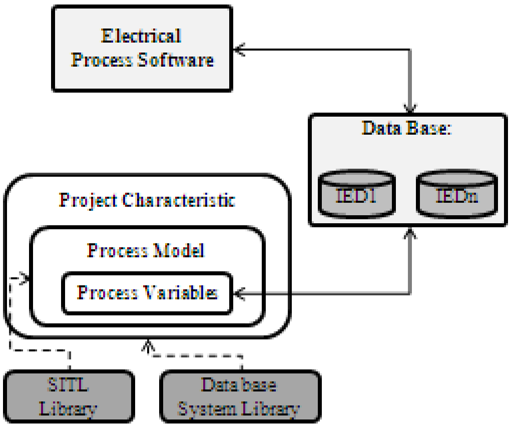


Fig. 6 Software in the loop package

The data base shown in Fig. 6 is used as an intermediate component between the OpNet process and the electrical process interface. Each IED electrical process values is included as a table inside this database in order to have at the end all the electrical process values of all the IEDs integrated inside the modeled switchboards.

In order to ensure a real time exchange between the OpNet process variables and the data inside the database, two libraries are integrated inside the simulation software. The first one is a system library which is implemented in the project properties to provide a communication language between simulation software acting as a data base client and the data base server. The SITL library is developed in our research in order to authorize the simulation to read and write values inside the data base. The read procedure is programmed in a periodical mode in order to collect the changes of the process values. The period of generation will depend on the cycle time of the real IED which will be modeled. On each process value changes, this library generates the appropriate OpNet process interrupt to invoke the state that corresponds to the changes. The collected values will be then placed inside the simulation environment variable. The writing procedure is programmed in event mode. This function is invoked by the library when an external client sends a writing request to an IED model inside the simulation.

VIII. Hardware and Software in the Loop Test

The simulation test described in this paragraph shows how HITL module is used in order to test the compliance of data modeling inside an IED model.

A. *IEC61850 modeling concept*

The IEC61850 includes a comprehensive set of data models for substation (and beyond) functionality. The models are, in general, constructed from a set of common data classes (defined in part 7-3 of the standard). The common data classes are used to build data objects, which are then clustered in logical node classes. Each logical node class represents some basic application function. Instances of the logical node classes are then combined to construct logical devices. A physical network device may contain an arbitrary number of logical devices. The set of classes (common data classes, data objects, and logical node classes) define the network visible data types (network interface), along with some standardized behavior, definitions, and configuration parameters [8]. One of the most important points in testing a physical or model device is to verify the compliance of its data object modeling. The paragraph below describes how HITL module is used for accomplishing this point.

B. Hardware in the loop for testing object modeling

When a physical or a model IED is developed, it is delivered with its PICS and MICS documentation which describe the capability of the IED in term of functionalities and services. Before delivering the IED, it is crucial to verify the conformity of these two documentations according to the IEC61850 specification to increase confidence in implementation (i.e. interoperability). Verifying the MICS of the IED consists to check:

- The existence of the functionalities defined in the MICS
- The compliance of the object modeling according to IEC 61850 standard dictionary (i.e. part 7.3 and 7.4)

The testing of IEC 61850 implementations is potentially a large task, given the large scope of the standard services and data models. IEDScout is a universal IEC 61850 test systems [9] that can be connected to any IEC 61850 device (server) and provides many useful functions for verifying data model, reading and writing data, reporting, along with GOOSE publishing and subscribing.

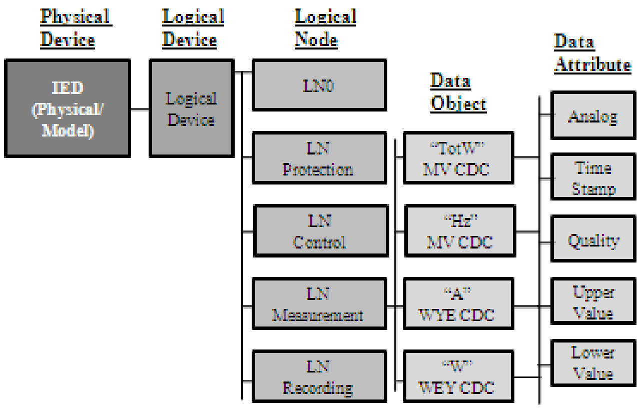


Fig. 7 Simple server MICS representation

A simple IEC 61850 measurement unit functionality has been implemented in our IED model to test the data object implementation. The model IED does not implement the functionality, but it is important to note that even unmapped model elements are implemented in the IEC61850 side of the model. The MX logical node contains four standard measurement unit data objects: TotW, A, W, Hz which permits to calculate the energy, current, power and frequency. Fig. 7 shows the MICS representation of the IED model.

Fig. it is explained that the generation of IEC61850 dictionary relies on the implementation of essential points:

- An ICD configuration file which describes the content of the server (i.e. simple server in our example). This file is generated by the SCL standard editor that produces IEC61850 configuration files.
- SCL RTG Library which includes predefined IEC 61850 code that aims to generate on run time the IEC61850 object dictionary according to the ICD configuration file and to implement the auto discover functionality.

For testing the conformity of implementing the server data model, HITL is used to connect the real IEDScout test system to the IED model using real to simulated scenario. The aim idea consists on showing the content of the IED model and identifying the object included in the model that are conformed or not to the standard dictionary. The test system sends an auto discover query to the configured IED model in order to discover the content of its standardized data object. This query passes through the HITL component as shown in Fig. 8. At its turn, the IED model sends back a file transfer response containing its entire data object content, the response reaches the real world also through the HITL component. In our example, the test system shows clearly the content of the simple server included in Fig. 8. This allows us to certify the implementation of simple server data object.

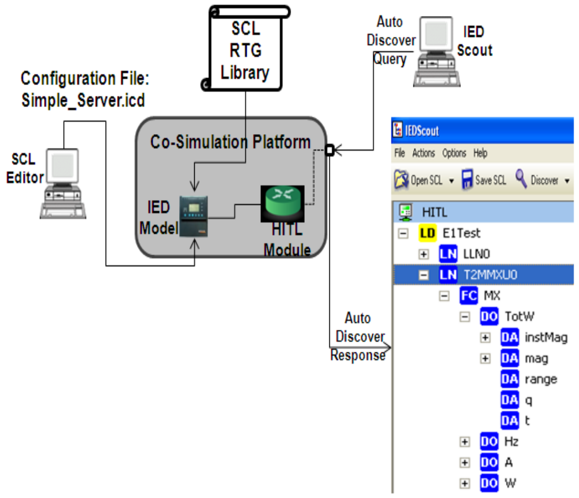


Fig. 8 Object modeling testing platform

C. Software in the loop Test

In order to validate the functionality of the SITL package, a small application is developed in our study. Inside this application, the electrical process software is developed in order to support 1 IED containing 4 categories of process values: Current, voltage, frequency measurements and finally the status position.

In order to validate the SITL package, a simple OpNet modeler node is developed. This node contains a simple module that implements one process model. This process model will be further implemented inside the application OpNet module of the final IED IEC61850 model. The first state 'red' is an unforced state which waits for an interruption coming from the process in order to execute the output transitions. This interrupt will be generated periodically from the SITL library. On the exit executive of the initial state, many states can be invoked depending on the condition calculated by the SITL Library. The activation of a condition depends weather the process value has changed or not. Each process measurement (i.e. current, voltage, frequency, and status) is associated to a forced state in order to run the control procedure associated to the measured value. In order to give a demonstration of the SITL package, the measurements are printed in the OpNet debugger console 'ODB' inside each forced state. As shown in the OpNet Debugger Interface 'White Window' in Fig. 9, when the voltage value changes to 220kV, this value is inserted inside the data base and printed on the OpNet debugger 'Output Consol of OpNet'.

IX. Conclusion

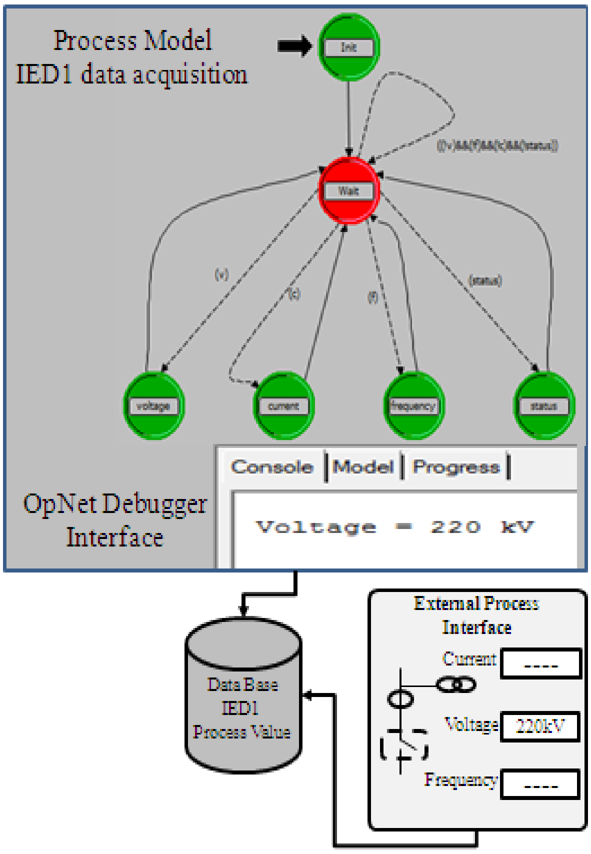
The paper shows the development of a co-simulation platform, for the evaluation of IEC61850-based architectures. The use of OpNet Modeler, which is an event-based simulation tool for networks, is interesting in such a context. A HITL/SITL gateway has been developed, allowing interfacing physical actual components, and virtual components, altogether to constitute a whole "hybrid" architecture.

The whole set of tests, required by the IEC 61850 standards, can be executed on this hybrid architecture, which allows the evaluation of architectures, in the design phase, as well as existing architectures but which require off-line validations.

The preliminary tests of the approach are promising; the next stage will be the validation of the approach on an actual existing architecture.

Acknowledgment

We are thankful to OpNet Technologies, Inc. for their active support under OpNet University Program (OUP).



Voltage = 220 kV

Fig. 9 Software in the loop package validation

References

[1] M. Haffar, J.M. Thiriet, E. Savary - Modeling of substation architecture implementing IEC 61850 protocol and solving interlocking problems - 7th IFAC International Conference on Fieldbuses & Networks in Industrial & Embedded systems, Toulouse, France (November 7-9, 2007), pp. 291-294.

[2] IEEE C37.115-2003, "IEEE Standard Test Method for Use in the Evaluation of Message Communications Between Intelligent Electronic Devices in an Integrated Substation Protection, Control, and Data Acquisition System," IEEE Power Engineering Society, New York, June 2004

[3] E. A. Udren, D. Dolezilek, "IEC61850: Role of Conformance Testing in Successful Integration"; KEMMA T&D and Schweitzer Engineering Laboratories 2006

[4] IEC; IEC61850 Communication Networks and Systems in substations, part 10: Conformance Testing (IEC61850-10), 2005; http://www.iec.ch

[5] J.Guo, W.Xiang, S.Wang ; "Reinforce Networking Theory with OPNET Simulation," Journal of Information Technology Education, MI, USA, 2007.

[6] C. Jiongcong, R.Yanming, G. Xinhua, J. Yangjun, "The research on Conformance Testing Platform of Numerical Substation," Elec. Dist., CICED 2008, Decembre 2008

[7] R. Martinez, W. Wu, K. Mcneill, "hardware and software-in-the-loop techniques using the opnet modeling tool for jtrs developmental testing," MILCOM, Monterey, October 2003.

[8] A.P. Apostolov, "Modeling Systems with Distributed Generators in IEC 61850," Power Sys. Conf, PSC '09. Clemson, SC, March 2009.

[9] A. Apostolov, B. Vandiver, "Functional Testing of IEC 61850 Based IEDs and Systems," Power System Conference and Exposition, Los Angeles, PES2004, October 2004

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 825–830

ISBN 978-83-60810-27-9
ISSN 1896-7094

Real-time controller design based on NI Compact-RIO

Maciej Rosol, Adam Pilat, Andrzej Turnau

***Abstract*—The paper is focused on NI Compact-RIO configured as a controller for the active magnetic levitation used here as a benchmark for time-critical systems. Three real-time configurations: soft, soft with IRQ and hard FPGA are considered. The quality of the real-time control has been tested for each configuration.**

***Index Terms*—real-time control, magnetic levitation, CompactRio, scheduling.**

I. Introduction

THREE different NI Compact-RIO (cRIO) configurations are designed and verified in control experiments performed in the real-time. The main attention is focused on real-time deterministic behavior of the PID controller constructed in a different way for a particular configuration. cRIO is recommended and promoted by National Instruments company as a rugged industrial control and acquisition system that incorporates a real-time processor and reconfigurable FPGA for reliable stand-alone embedded applications. The active magnetic levitation used here as a benchmark is a time-critical system. Therefore, a punctual and fully determined control algorithm is required. The quality of the control algorithm execution depends strongly on a used platform: Power PC or FPGA and the execution mode: software timing loop, interrupt event, hardware timing loop. A desirable design goal is usually the construction of the so-called ”hard real-time” system. We do not always manage to meet this goal. In principle, to develop the hard real-time controller a certain level of familiarity with cRIO and skill is required. Control experiments that have been performed illustrate several timing aspects: jitter, execution time of control algorithm, determinism of data exchange, ect. It is important to answer to the following question. How far one can dimnish the sampling period not disturbing the system performance?

II. Real-Time Control Application Design

The reconfigurable control system may contain the following components:

1) cRIO FPGA core application for input, output, communication, and control,
2) time-critical loop for floating-point control, signal processing, and point-by-point decision making
3) normal-priority loop for embedded data logging, remote Web interface, and Ethernet communication
4) networked host PC for remote graphical user interface, historical data logging, and postprocessing

Depending on requirements of an application, one can implement particular components. The RIO FPGA chip is connected to the I/O modules in a star topology, for direct access to each module for precise control and unlimited flexibility in timing, triggering, and synchronization. A local PCI bus connection provides a high-performance interface between the RIO FPGA and the real-time processor. The magnetic levitation control system structure is shown In Fig. 1.

A PC is dedicated to data acquisition and monitoring. The cRIO-9014 controller is equipped with two hardware platform: the Power PC microcontroller operating under the real-time the VxWorks operating system and FPGA (a Spartan-3 XILINX chip containing 3 milions gates). Power PC is triggered by a 400 MHz clock. FPGA is triggered by a 40 MHz clock. The PWM generator of the control signal is implemented in FPGA. There are also two modules: NI 9401 digital I/O and NI 9215 analog I/O. The PWM signal is transfered from FPGA through the NI 9401 output digital module and the PWM power interface to the electromagnet that operates as the actuator. The signal proportional to the sphere position is transfered from the sensor to the sensor conditioning circuit and farther to the NI 9215 input analog module.

III. Time-critical experiments of two cRIO configurations

Three cRIO configurations shown in Fig. 2 have been built to be tested:

1) Power PC soft real-time,
2) Power PC Interrupt ReQuest (IRQ),
3) FPGA hard real-time.

The PID controller operating in a hard real-time loop is built inside the FPGA chip (Fig. 3). The sampling period can be changed.

However, its minimal value has to be set to 10 ms due to the conversion time of the analog signal (the maximal value is equal to 100 kHz per chanel). The real-time platform is responsible for communication with FPGA and the application that runs on PC and is responsible for changing the parameters of the generator PWM and PID controller. If we use Power PC and VxWorks as a real-time control platform then FPGA is only used to receive the measurment signal and transfer the control signal and measure the so-called jitter signal. The timing loop operating at the 40 MHz frequency is responsible for generating the PWM resolution signal. It generates also the 32-bits counter (in ticks) for the jitter measuring. The measurement accuracy is the clock triggering frequency dependent and is equal to 25 ns. The time-critical while loop executes the PID control algorithm. The position signal related to the sphere position is scaled in meters due to the one-dimensional

978-83-60810-27-9/09/$25.00 © 2010 IEEE

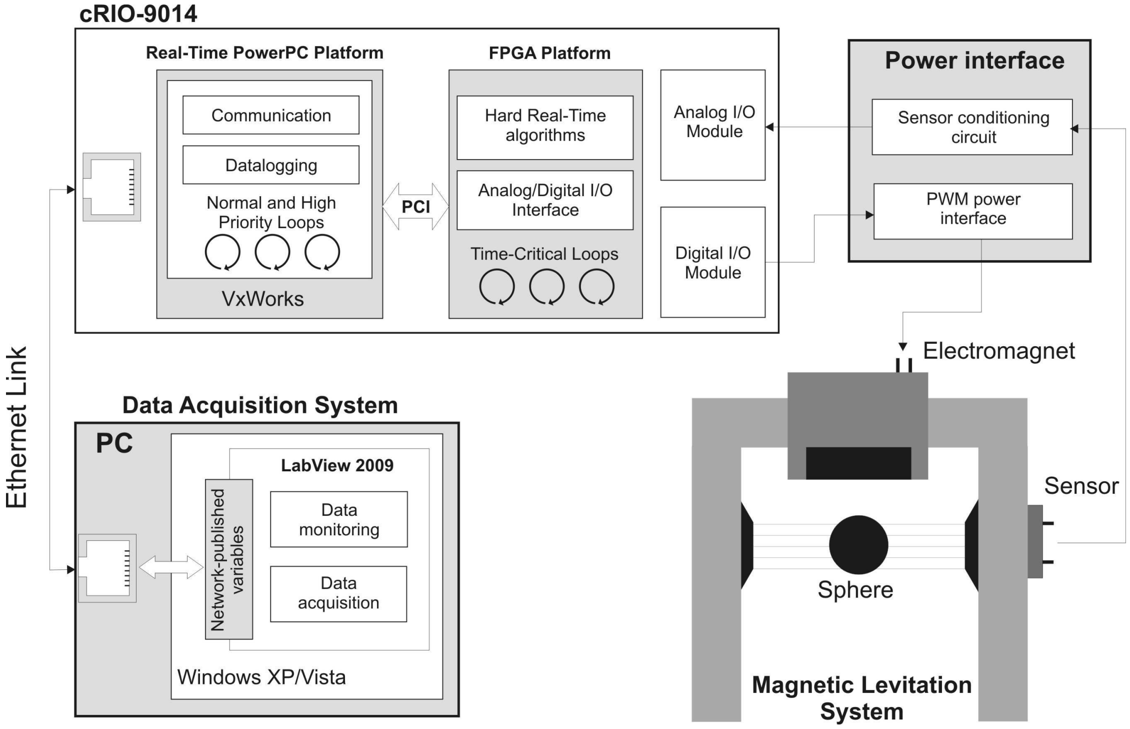


Fig. 1. Structure of magnetic levitation control system

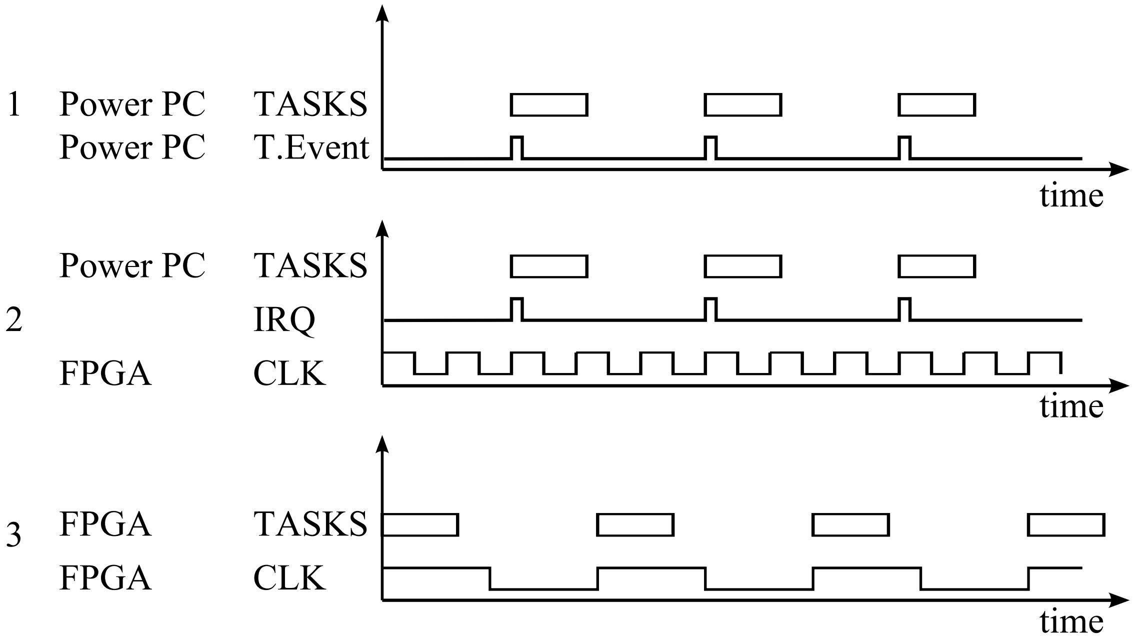


Fig. 2. Three experimental cRIO configurations

look up table. Calculations of the controller are expressed as fixed-point numbers wit the accuracy equal to $1.5259 \cdot 10^{-5}$. The output signal of the controller is a number from the range $0 \div 1$. It is scaled and becomes a duty cycle in the range $0 \div 4095$ as far as the 12-bits resolution is concerned. The sampling period for the FPGA hard real-time configuration has been verified and the 25 ns value has been confirmed. This is obvious due to the hardware paralel implementation of the PID algorithm. In Fig. 4 the logging sessions of primary work parameters corresponding to the operational VxWorks system (the upper frame) and to the realization of the user application (the lower frame) are shown. During the Power PC

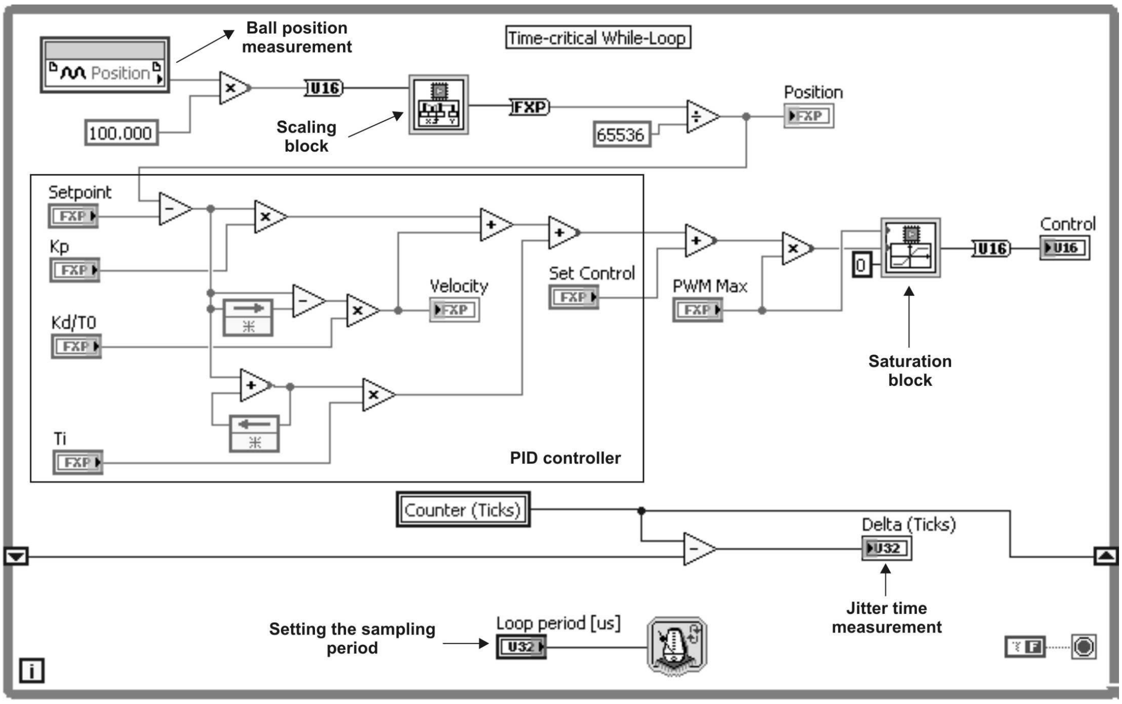


Fig. 3. The PID controller operating in a hard real-time loop running on FPGA chip

session calculation procedures have been executed with data exchange with the FPGA system. This was done to check the performance of this platform with the overhead time devoted to communication. The grey area in Fig. 4 corresponds to one complete execution of the loop which lasts 1 ms. Four interesting signals can be noted in the lower frame, namely: 0-SquareGener, 1-PositionScale, 2-PD_Controller, 4-Saturation. The first one provides square wave signal generator. The second one is devoted to scaling of the sensor signal. Next one implements the PID controller and the last one limits the control signal to the physical bounds. The execution times of these procedures are 28.1334 ms, 22.8882 ms, 30.0407 ms and 16.2125 ms respectively. It means that the time of calculations together with the context switching time between the procedures is equal to 355.84 ms. Hence, the maximal theoretical sampling frequency, without taking into account VxWorks time overhead, is equal to 2.81 kHz. In practice one should not go beyond 1 kHz. Similarly to the Power PC soft real-time configuration the performance of the Power PC Interrupt ReQuest control structure was check by running the Real-Time Execution Trace Toolkit session.

In Fig. 5 the logging data are presented. As can be seen the real-time task is IRQ triggered. The PowerPC platform waits for interrupt generating by FPGA. The execution times of the calculation procedures are comparable to Power PC soft real-time configuration. However the total time of calculations is shorter (equal to 253.27 ms). Hence, the maximal theoretical sampling frequency, without taking into account VxWorks time overhead, is equal to 3.95 kHz. In practice one should not go beyond 2 kHz.

IV. Experimental jitter of two configurations

The real-time controllers have been tested in three cases. However, the jitter has not been illustrated in the third case. This is not a mistake. There is nothing to show. Only the third case is characterized by high accuracy and repeatability of execution determined by 25ns resolution. The FPGA chip is triggered by the hardware timer. In contrast to this case, the first two have different triggering mechanisms. Usually, they introduce uncertainty into the real-time execution. The time stamps in the number of control task events 650 were registered for jitter analysis. The histograms are scaled in the full range of the available data. The results corresponding to two experimental cRIO configurations (see Fig. 2) are shown in Fig. 6 and Fig. 7. Let us define the following factors for further results analysis:

- min - is the minimal time instant registered for an event,
- max - is the maximal time instant registered for an event,
- N - is the maximal number of requests at the same time interval.
- xs [μs] - is the time instant related to the maximal number of events, alas different to the nominal (desired) time instant,
- DN [%] - is the negative deviation from xs,
- DP [%] - is the positive deviation from xs,
- DNn [%] - is the negative deviation from the nominal (desired) time instant,
- DPn [%] - is the negative deviation from the nominal (desired) time instant.

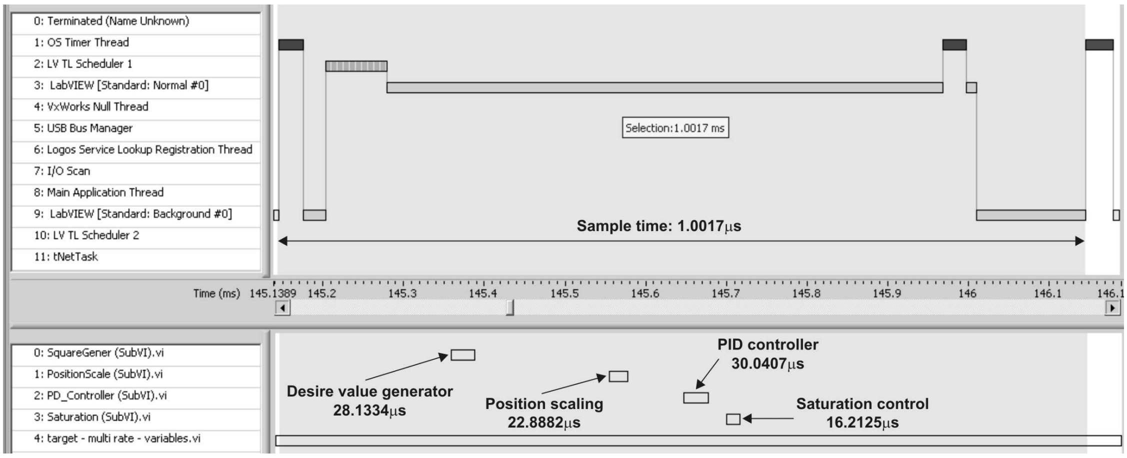

0: Terminated (Name Unknown)
1: OS Timer Thread
2: LV TL Scheduler 1
3: LabVIEW [Standard: Normal #0]
4: VxWorks Null Thread
5: USB Bus Manager
6: Logos Service Lookup Registration Thread
7: I/O Scan
8: Main Application Thread
9: LabVIEW [Standard: Background #0]
10: LV TL Scheduler 2
11: tNetTask
Selection:1.0017 ms
Sample time: 1.0017µs
Time (ms) 145.1389 145.2 145.3 145.4 145.5 145.6 145.7 145.8 145.9 146 146.1 146.1
0: SquareGener (SubVI).vi
1: PositionScale (SubVI).vi
2: PD_Controller (SubVI).vi
3: Saturation (SubVI).vi
4: target - multi rate - variables.vi
Desire value generator 28.1334µs
Position scaling 22.8882µs
PID controller 30.0407µs
Saturation control 16.2125µs

Fig. 4. Real-Time Execution Trace Toolkit session for the Power PC soft real-time configuration with 1 ms sample period

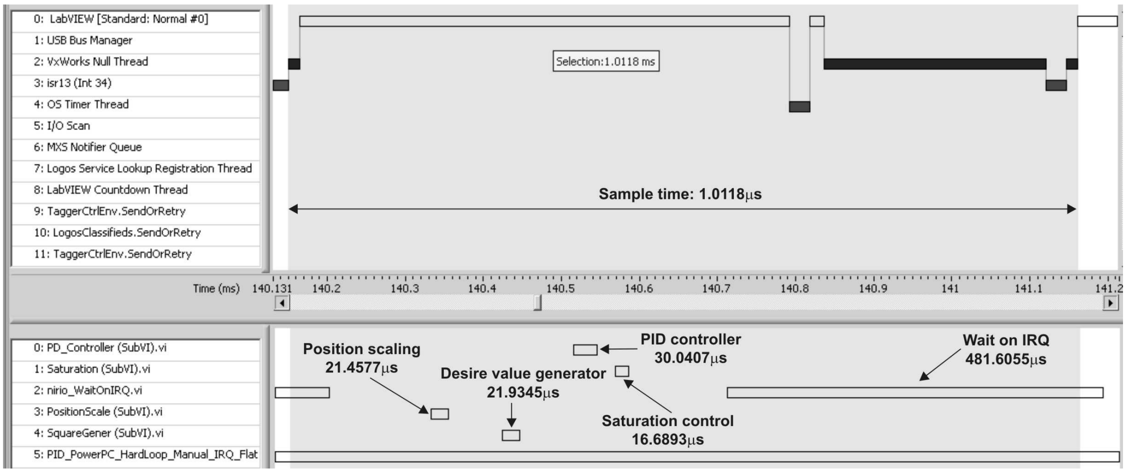

0: LabVIEW [Standard: Normal #0]
1: USB Bus Manager
2: VxWorks Null Thread
3: isr13 (Int 34)
4: OS Timer Thread
5: I/O Scan
6: MXS Notifier Queue
7: Logos Service Lookup Registration Thread
8: LabVIEW Countdown Thread
9: TaggerCtrlEnv.SendOrRetry
10: LogosClassifieds.SendOrRetry
11: TaggerCtrlEnv.SendOrRetry
Selection:1.0118 ms
Sample time: 1.0118µs
Time (ms) 140.131 140.2 140.3 140.4 140.5 140.6 140.7 140.8 140.9 141 141.1 141.2
0: PD_Controller (SubVI).vi
1: Saturation (SubVI).vi
2: nirio_WaitOnIRQ.vi
3: PositionScale (SubVI).vi
4: SquareGener (SubVI).vi
5: PID_PowerPC_HardLoop_Manual_IRQ_Flat
Position scaling 21.4577µs
Desire value generator 21.9345µs
PID controller 30.0407µs
Saturation control 16.6893µs
Wait on IRQ 481.6055µs

Fig. 5. Real-Time Execution Trace Toolkit session for the Power PC Interrupt ReQuest configuration with 1 ms sample period

A. *Configuration 1*

One can notice that the sampling frequency equal to 1 kHz is a critical value for the soft real-time. Decreasing the sampling period (from 1 ms to 0.5 ms) is not feasible. The system operates at 995 ms (see Fig. 6c). This demonstrates that the clock mechanisms used by LabView and VxWorks do not cooperate correctly in the Power PC timing loop configuration. Moreover this environment shows also a lack of punctuality for 1 and 2 ms sampling periods (see the histograms in Fig. 6a and Fig. 6b).

TABLE I
STATISTICS OF CASE 1

Case	min [μs]	max [μs]	N
a	1976.847	2065.452	99
b	1083.259	1116.691	36
c	963.333	1026.092	80

In the case of timing events triggered by the FPGA clock. when tasks are executed at Power PC (illustrated in Fig. 7) we are also surprised by the fact that the system triggered by the FPGA interrupt responds in a strange way. There is a large distribution of single events fortunately quantitatively

TABLE II
CONFIGURATION 1 CASE A

xs	2005.788
DN %	-1.442
DP %	2.974
DNn %	-1.157
DNp %	3.272

TABLE III
CONFIGURATION 1 CASE B

xs	1098.379
DN %	1.376
DP %	1.667
DNn %	8.325
DNp %	11.669

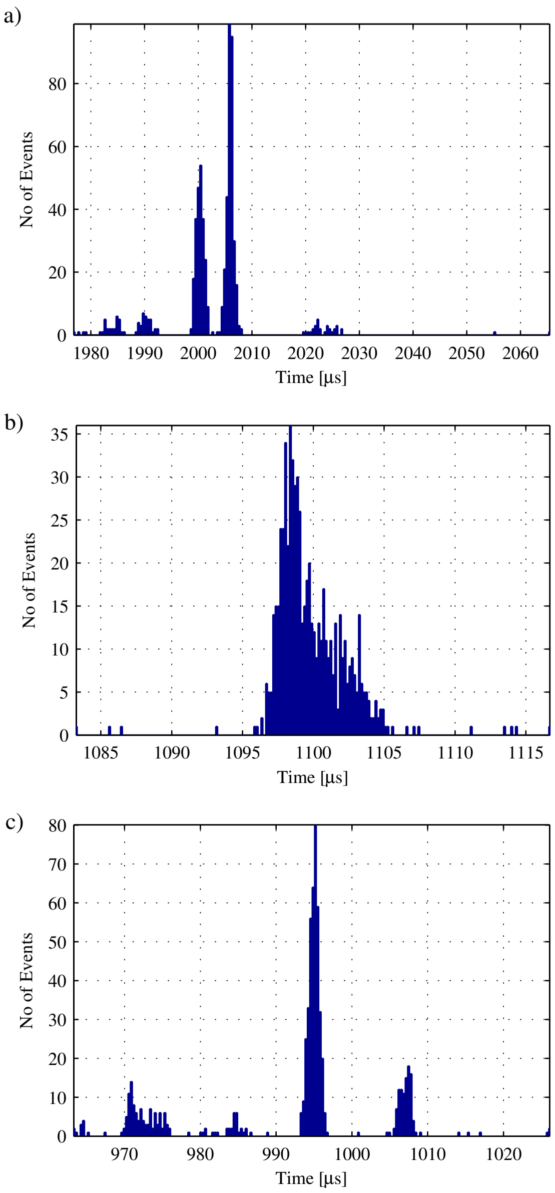


Fig. 6. . Histograms a) 2ms, b) 1ms, c) 0.5 ms for the timing loop with the time critical priority.

TABLE IV
CONFIGURATION 1 CASE C

xs	995.185
DN %	3.200
DP %	3.105
DNn %	92.666
DNp %	105.218

insignificant. It is surprising, as if the interrupt handler poorly managed IRQ despite the existence of the acknowledgment mechanisms. Of course, one could neglect the events that occurred fewer times than 10% of the 650 recorded events. Then statistics would be beneficial for the IRQ. As far as Configuration 2 is concerned we can notice a significant improvement in control quality, much of the tasks is executed in the vicinity of the desired time. However, even at such a small number of events as 650 there was a significant time variation (jitter) resulting from the interrupt handler (interrupt routines).

B. *Configuration 2*

Analysing the collected data it should be noted that in both cases (the Configurations 1 and 2), the magnetic levitation system steered by PID controller, where the derivative of the error is determined, would pay a high risk of loss of stability in the worst case, at best, the occurrence of oscillations. Uneven course of events has influence on the rate of change of the error signal fed to the input of the differentiating controller part. Unfortunately, a value calculated in such a way is incorrect and inconsistent with reality.

TABLE V
STATISTICS OF CASE 2

Case	min [μs]	max [μs]	N
a	1791.058	2490.667	353
b	723.133	1214.041	410
c	373.309	645.915	105

TABLE VI
CONFIGURATION 2 CASE A

xs	2005.510
DN %	10.693
DP %	24.191
DNn %	10.447
DNp %	24.533

TABLE VII
CONFIGURATION 2 CASE B

xs	999.423
DN %	27.644
DP %	21.474
DNn %	27.686
DNp %	21.404

TABLE VIII
CONFIGURATION 2 CASE C

xs	499.338
DN %	25.239
DP %	29.354
DNn %	25.338
DNp %	29.183

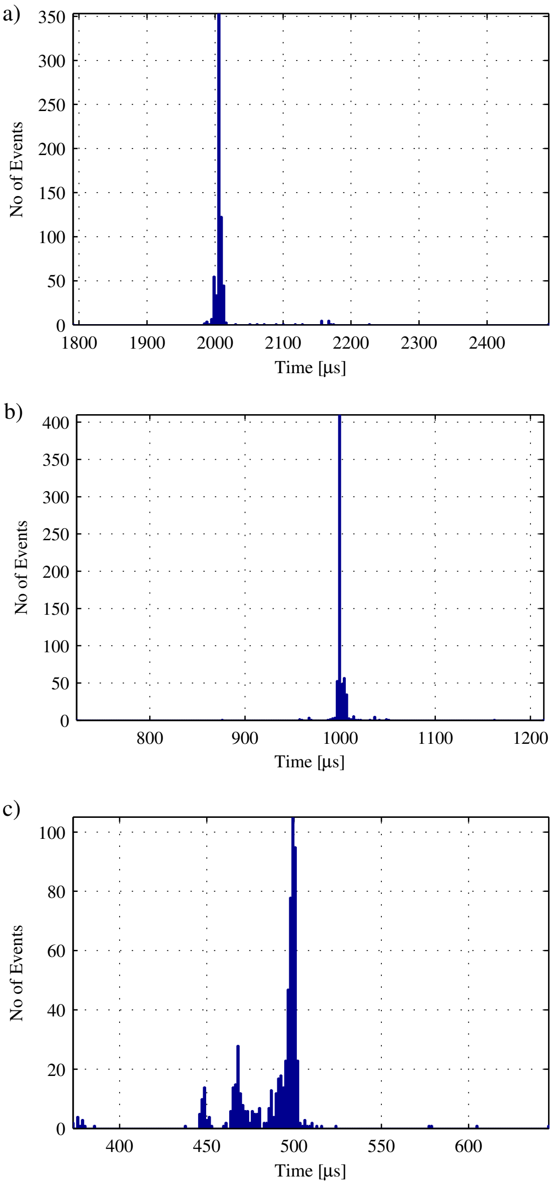


Fig. 7. Histograms a) 2ms, b) 1ms, c) 0.5ms for IQR timing events triggered by the FPGA clock. Tasks executed at Power PC

V. Conclusion

With this research it has been shown that the real-time control development must be considered precisely and with specific attention to the used hardware. The designed controller must be checked and verified how operates. Especially the punctuality of the control task call must be satisfied. The timing loop mechanism with time-critical priority is a fine mechanism for triggered tasks, but is limited to 1kHz of the sampling frequency. The timing events are condensed around the requested sample period. In the case of IRQ driven task, generated on the base of FPGA clock, the events are handled at the desired sample time, but sometimes a few events are called with a distance of about 25% far away from the nominal trigger time. The observed jitter gives a number of inequalities in the controller calculation. Note, that the error derivative calculation is very sensitive for incorrect timing. The tested National Instruments CompactRIO hardware gives a number of possibilities to develop a wide range of control configurations. The PowerPC is dedicated to data management and exchange between hard time-critical layer realised in the FPGA. Summarising, the control task should be implemented in the FPGA and the Power PC used for data acquisition and controller adjustment task only.

References

[1] P. Bilik, L. Koval and J. Hajduk, *CompactRIO Embedded System in Power Quality Analysis*, Proceedings of the International Multiconference Computer Science and Information Technology, 2008, pp. 577 580

[2] W. Blokland and G. Armstrong, *A CompactRIO-based Beam Loss Monitor for the SNS RF Test Cave*, Proceedings of EPAC08, Genoa, Italy, pp. 1050-1052

[3] C. Dase, J. S. Falcon, and B. MacCleery, *Motorcycle Control Prototyping Using an FPGA-Based Embedded Control System*, IEEE CONTROL SYSTEMS MAGAZINE October 2006, pp.17-21

[4] InTeCo Ltd., *Magnetic Levitation System, Users Guide*, Inteco 2003

[5] National Instruments LabVIEW Toolkit Web portal, "LabVIEW Toolkits," [Online]. Available: http://www.ni.com/toolkits

[6] National Instruments LabVIEW Real-Time Web portal, "LabVIEW Real-Time for measurement and control," [Online]. Available: http://www.ni.com/realtime

[7] A. Pilat, *Control of magnetically levitated systems*, Ph.D. Dissertation (in Polish), AGH University of Science and Technology, 2002, Kraków, Poland

ISBN 978-83-60810-27-9
ISSN 1896-7094

Intelligent Car Control and Recognition Embedded System

Vilem Srovnal Jr., Zdenek Machacek, Radim Hercik, Roman Slaby,
Vilem Srovnal

VSB – Technical University of Ostrava, Measurement and Control, FEECS
Ostrava – Poruba, Czech Republic
Email: {vilem.srovnal, zdenek.machacek,
radim.hercik, roman.slaby, vilem srovnal }@vsb.cz

***Abstract*—There is presented control system design with autonomous control elements focused on field of automotive industry in this paper. The main objective of this document is description of the control and monitoring system with integrated image processing from the camera. The images obtained from the camera are used for recognizing routing and traffic situation. During system proposal we focused our attention on integration of components for the car localization using GPS and navigation system as well. The implemented embedded system communicates with other car control units using CAN bus and industrial Ethernet. The communication interface between driver and car integrated system is carried out by process visualization on the LCD touch panel.**

I. Introduction

Results for automotive industry development has been the main aim of the presentation in this paper. Our research has focused on process control and monitoring of a process states and values and the specific solution for intelligent car control with autonomous elements has been presented in this contribution. The solution was implemented in a prototype of an electric car, where video-cameras for image processing and objects recognition in a real time were used. Stream data of the recognized objects allow identify route lane, people, other cars, traffic signs and situations. In addition to objects recognition, the accurate position must be obtained from GPS module and then transferred to control system. Information of exact position is useful for the car localization process and system navigation. The position data can be helpful during estimation of a distance between car and surrounding objects where supersonics sensors cannot be used.

The actuators, sensors, control units and video-cameras communicate with each other using industrial interfaces. Implemented industrial interfaces are CAN bus, wire or wireless industrial Ethernet and Bluetooth for extended features. The communication system is able to transfer secured data from one car to another by using open wireless industrial network. Car view information, captured on the camera, is transferred from the car to intelligent highway system. There is also space and possibility to use wireless communication connection for remote diagnostic test. [6]

Information of the system is displayed on the touch screen panel that is connected into central development board based on 32.bits processor i.MX35. This type of processor is suitable for communication, graphic and multimedia applications. In virtue of using a lot of hardware interfaces and due to complexity of realization, we made a decision to use two types of operating system. Both of them are based on Unix platform. The first one is RTAI Linux which supports a lot of drivers for devices like cameras or modules for wireless connection or Bluetooth. On the other hand there is a lack of preferable memory protection. The kernel runs in the same memory space like drivers and the bug in driver can consequently cause the failure of the whole operating system. The new Linux reboot lasts rather long and it is annoying for a car driver or other passengers in the car. The second solution offers applying the system with great memory protection and memory adaptive partitioning management. The micro-kernel and drivers or other processes run in separate memory space. The failure of one driver or process does not cause operating system crash but only automatic process restart in milliseconds. Subject of concern is the operating system QNX Neutrino RTOS. Disadvantage of QNX Neutrino RTOS is weak support for wireless network drivers.[4]

II. Car System Architecture Design

The system architecture design is composed of embedded modules with special or universal function for the car control process. The central control embedded unit has many functions such as the system for battery management, motion control system, image recognizing system, global position system and others. In this paper we have focused our attention to the system for image processing and recognition of objects. The system embedded design is proposed for mobile devices like robots, car or industrial devices for product testing or measurement. Actual development design is composed of low cost products.

I. Embedded Car System

We propose embedded car system with 8.bits, 16.bits and 32.bits processors according to its functionality. The central unit is based on ARM architecture with build-in 32.bits processor i.MX35. This embedded central unit is reserved for graphic applications, image processing with objects recognition and communication processes. We prefer low power

This work is supported by grant of the Grant Agency of the Czech Republic GA102/08/1429 - Safety and security of networked embedded system applications.

consumption for all our embedded units and that is why we have chosen ARM architecture. [5]

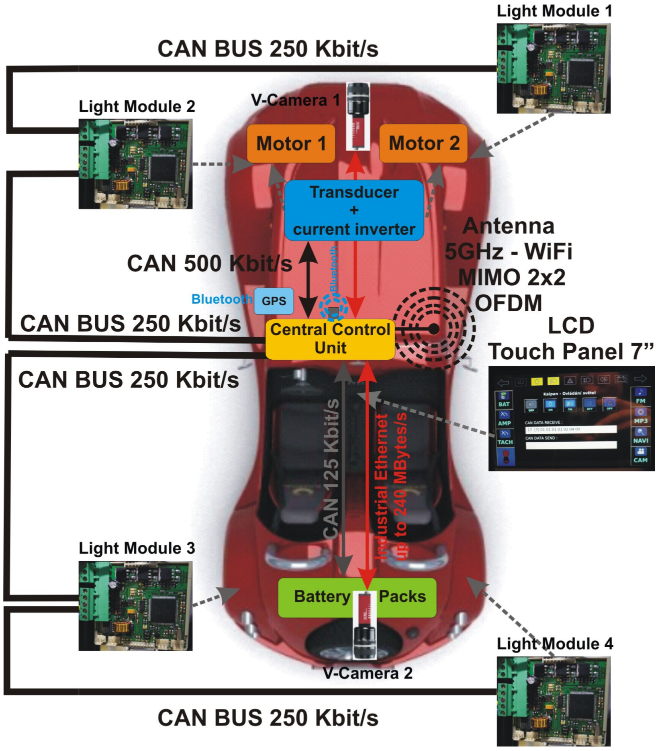


Fig. 1 Embedded car system architecture

Figure 1 presents some of the basic embedded modules implemented in the car control and monitoring system.

II. Software Implementation for Vision Car System

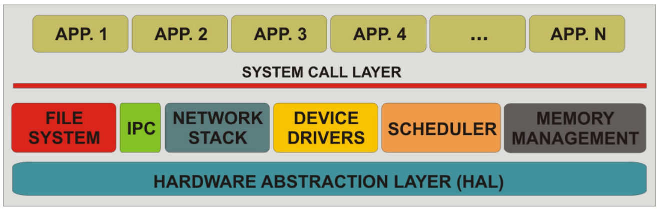
The vision system capability is to react to various types of traffic situations. Together with processing image data there has to be artificial intelligence implemented with ability to predict a critical traffic situation. Our development has been divided into 3 phases. The first one is testing our objects recognition algorithms design in Matlab with cheap webcam on platform x86. Standard USB interface was used for webcam connection. The next development phase was a test on mobile devices like PDA. The last phase was a transfer of the recognition system in embedded device that is part of car system. A new communication interface was used. We decided to choose a new camera with GigaEthernet interface on industrial Ethernet protocol. There is high data stream of data for a new 4 Megapixel camera with resolution up to 2336 x 1752l therefore we need very fast communication interface.

III. Real Time Operating System Implementation

Central control unit carries many functions which present a lot of processes running at the same time. Therefore we decided to use the real time operating system. There are 2 types of operating system mentioned in this paper – RTAI Linux and QNX Neutrino RTOS. [1]

RTAI Linux was modified for our purpose and for our hardware architecture. For this operating system, communication drivers have been adjusted for wire and wireless connection, based on industrial protocols - FlexCAN, EtherCAT and Bluetooth.

QNX Neutrino RTOS was rather easier to implement due to BSP supplied by QNX Company, but there was a problem with wireless communication driver. We still continue working on it with QNX community cooperation. Following figures 2 and 3 show the difference between Embedded Linux and QNX.

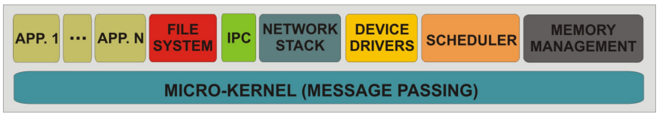

APP. 1 | APP. 2 | APP. 3 | APP. 4 | ... | APP. N

SYSTEM CALL LAYER

FILE SYSTEM | IPC | NETWORK STACK | DEVICE DRIVERS | SCHEDULER | MEMORY MANAGEMENT

HARDWARE ABSTRACTION LAYER (HAL)

Fig. 2 Architecture of embedded Linux operating system

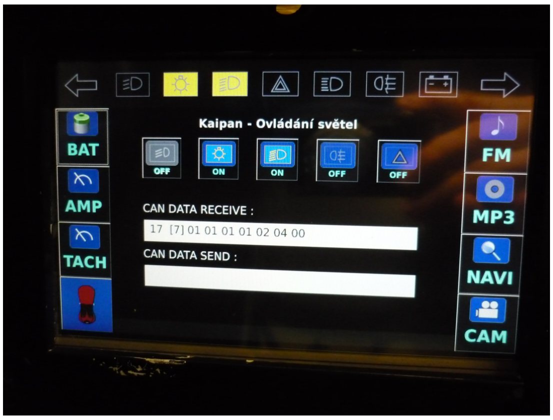

APP. 1 | ... | APP. N | FILE SYSTEM | IPC | NETWORK STACK | DEVICE DRIVERS | SCHEDULER | MEMORY MANAGEMENT

MICRO-KERNEL (MESSAGE PASSING)

Fig. 3 Architecture of QNX Neutrino RTOS

IV. QT cross-platform graphic framework

The software Qt is the cross-platform application and UI framework used for embedded applications. Graphics interface was chosen with regards to both operating systems. Our selection was influenced by our experiences with graphical QT system from Nokia. QT embedded graphical interface is applicable for Linux and QNX as well. Figure 4 shows the example of car graphic user interface.


Kaipan - Ovládání světel

CAN DATA RECEIVE :

17 [7] 01 01 01 01 02 04 00

CAN DATA SEND :

Fig. 4 LCD touch panel with QT embedded car system

III. Recognition Driver System

The recognition driver system was developed for traffic signs and traffic lanes detection. The system processes real-time image captured by a camera. Implemented algorithms are based on the idea of standardized form and appearance of traffic signs.

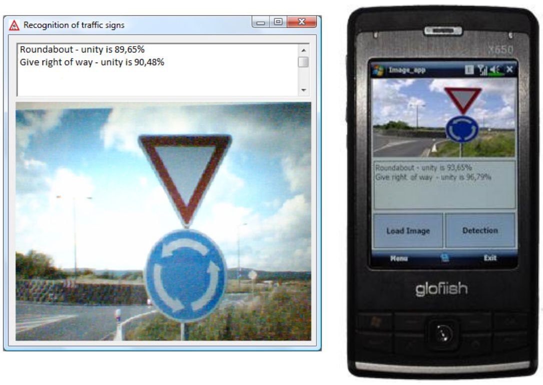
Nowadays there are two production technologies of image scanners available, namely scanners based on CMOS and CCD technologies. Better quality cameras use scanners based primarily on CCD technology. The advantage of this technology is a high luminous sensitivity ensuring better image quality with low brightness. In comparison, the CMOS technology is much cheaper, because it is based on standard technology, which is used in mass production of memory chips. By virtue of this technology, the scanning element can be placed on one chip together with other electronic circuit elements. Moreover, the advantage of CMOS technology is lower energy consumption in comparison with CCD technology. Generally, there is a reasonable argument that when the scanner is heated up, the undesirable noise increases and reduces the quality of final record.

For the presented application determined for image recognition, the VGA camera has satisfactory resolution. Better resolution would extend the process time, as each point of image matrix prolongs recognition process. In this case, a web camera with VGA resolution based on CMOS technology is used for image recording. These cameras communicate with in-built systems with the help of multi-purpose standard interface. The advantage of these cameras is a low price and very good availability, while undesirable disturbance, which can be fixed by developed software, is a disadvantage. Methods for reducing the undesirable noise disturbance are based on using filters, such as linear and median filters, which are very simple, reliable and quickly implemented.

The developed system retrieves images in real time directly from a webcam and they are evaluated immediately. Webcams Logitech with resolution 352 * 288 pixels were used for the test. Image quality reflects necessary quality of traffic signs identification. Low image resolution of traffic signs causes that recognition success will decrease to 90%. It is about 5% less than if the original high resolution image is evaluated. Application speed is convenient, as it is able to process more than 10 frames per second with presented camera resolution.

An application is created for Linux, Windows CE, Windows Mobile QNX. Testing was performed on a development kit i.MX35. Use of 32-bit multimedia applications processor based on iMX357 ARM11 ™ core is run on 532MHz frequency and size of RAM is 128 megabytes.

Implemented algorithms are suitable for wide range of applications in existing PDA devices and mobile phones. These devices are limited by low frequency computing power and memory space compared to being commonly used in desktop computers. The application was tested in devices E-TEN Glofiish X650. This device contains a Samsung S3C2442 processor 500 of MH, 128 MB Flash ROM, 64 MB RAM and a VGA TFT display 2.8-inch. The device has also a camera 2Mpix. Installed Operating System is Windows Mobile 6 Professional. Testing has shown that reduced performance of the device has largely affected algorithms time consumption. Acceleration of algorithms is likely by more efficient use of resource devices and optimizing memory management. The developed application shows examples of detection system for traffic signs in Fig.5.



Roundabout - unity is 89,65%
Give right of way - unity is 90,48%

Roundabout - unity is 93,65%
Give right of way - unity is 96,79%

Fig. 5 Application for devices with operating system Windows CE and Windows Mobile

Execution time for implemented algorithms depends on several parameters. The first parameter is size of the input image, size and number of objects that are inside image. Number of inside objects is input criteria which affects the number of cycles that must be done during patterns comparison. The execution time, which consumes the algorithm, depends on the performance of computer on which it has been executed. Processing time code implemented in C on a desktop PC in test events, did not exceed the period of 1 second.

The verification of the developed software was accomplished approximately on 50 traffic signs. The majority of test results are correct, i. e. in majority of cases traffic signs were detected correctly, but rarely there was not detected any traffic sign. This error was caused by damage to traffic signs, the excessive pollution or poor light conditions. Traffic signs were successfully recognized with patterns in range from 90 to 95%.

During the recognition process, the distance of traffic signs from the camera plays a big role. Due to a small number of pixels of traffic sign that is located too far, it cannot be guaranteed that the system will find the required consensus. Threshold for identifying traffic signs is approximately equal to 50 meters, if the camera zoom value is set to 1:1.

IV. Algorithms for Traffic Signs Recognition

The algorithms are based on ideas of standardized appearance and shape of traffic signs. Parameters are defined by traffic signs in the Czech Republic, which are stated in the regulation no.30/2001, Ministry of Transport The algorithms consist of two main parts. The first part implements correction and segmentation of the input image. The second part implements object searching, analysis and user information system.

V. Image segmentation and conversion

The basic aim of image segmentation is to search continuous parts in whole figure from a camera. From these analysed parts the objects are created, which are explored by parameters and the similarity of the patterns. The method chosen for continuous parts searching is based on conversion of the input colour image into the binary structure, taking into account the colour matrix with limited number of colours. Converted traffic signs image is composed of 5 basic colours, where 4 colours (red, blue, black and yellow) determine the motive of the label and white colour is chosen as the background of the label. The analysis of algorithms also solves the segmentation problem of traffic signs, which are composed just of red and blue colours, so it is converted to only one object, which has to be divided to separated objects. There are not green colour analyses, because traffic signs do not contain this color. For converting the image to a binary algorithm defined by next presented function is used. The example of image segmentation is presented in Fig.6.

$$f\left(p_{(x,y)}\right)=\begin{cases}0 & for \quad R_{(x,y)},G_{(x,y)},B_{(x,y)}>=h\\ 1 & for \quad R_{(x,y)},G_{(x,y)},B_{(x,y)}<h\end{cases} \tag{1}$$

where: p is pixel
h is colour limit
R,G,B are colour components

1	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	1	1
1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0
1	1	0	1	0	1	0	0	0
1	1	0	1	0	1	0	1	1
1	0	0	1	0	1	0	1	1

2	0	0	0	4	4	4	4	4
2	2	0	0	0	0	0	4	4
2	2	0	0	0	0	0	0	0
2	2	0	3	3	3	0	0	0
2	2	0	3	0	3	0	0	0
2	2	0	3	0	3	0	5	5
2	0	0	3	0	3	0	5	5

Fig. 6 Continuous parts searching (left) and Individual part segmentation with different number of each segment (right)

VI. Image rotation and angle correction

The camera images, which are obtained from real camera output, can be rotated to incorrect angles. Rotation is related to one point by an angle α. The simplest cases are: $\alpha=90^{\circ}$, $\alpha=180^{\circ}$, $\alpha=-90^{\circ}$. In these cases it is practical and easy to change the representation of x and y axis. In reality, the angle α does not take exactly predictable values, thus we need to use trigonometric functions conversion to new pixel location. Special situation occurs when the traffic signs not only need to correct their orientation, but also to determine the angle of which are deflected from the vertical position. Traffic signs are not always installed completely vertically. This fact leads to lower conformity of recognition. The solution image has to be rotated to known angle. This angle can be calculated by algorithms based on symmetry of traffic signs as shown in Fig.7.

The figure shows the geometric layout of the problem where the presented trigonometric functions is used for determination the angle α. The presented algorithm for rotation calculation cannot serve for traffic signs of circular shape. Angle α is calculated by next equation. [3]

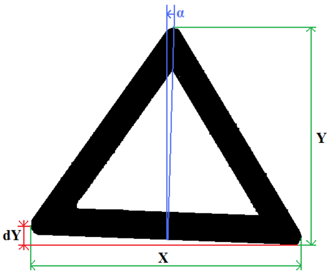


Fig. 7 Geometric expression of rotation traffic signs

$$tg\,\alpha=\frac{dY}{X} \tag{2}$$

The algorithm for rotation chooses reference point, which is located in the lower left corner of the editing image. The calculation of new coordinates is based on the next presented equations, where x‘,y‘ are new coordinates and x, y are current coordinates.

$$x'=\cos\left(\alpha+tg^{-1}\frac{y}{x}\right)\cdot\sqrt{x^2+y^2} \tag{3}$$

$$y'=\sin\left(\alpha+tg^{-1}\frac{y}{x}\right)\cdot\sqrt{x^2+y^2} \tag{4}$$

The rotation method gets images of traffic signs in a vertical position. New calculated pixel coordinates appear in the original matrix, but not always. Therefore there is necessary to consider boundary data fields so as not to exceed their limits.

VII. Image pattern analysis

After image segmentation and image rotation, there is possibility to analyze patterns with the actual edited image from camera. Traffic signs patterns are stored in a binary matrix. Each of the patterns is size-standardized at 100 x 100 pixels. The resolution is chosen on the basis of compromise between size of the matrix patterns and quality of the patterns.

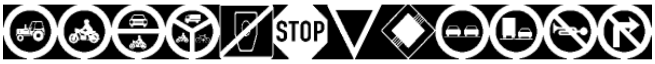


Fig. 8 Example of binary matrix with patterns of traffic signs

Each object has location in the original matrix and is given its size. In order to apply the correlation function, size unification securing of objects and patterns is needed. Function re-calculates the object's size on defined dimension. The resize algorithm can be used after object continues parts recognition. The captured image can contain more traffic labels with other recognized noise, which is not desirable. Therefore, the method for object segmentation is supplemented with object centres identification. In case the centre is outside the other object area, it is solved separately. This recognition method eliminates problems with noise parts of objects, which disturb object comparison with patterns by correlation. This is processed by analyzing the geometric centre of coordinates and objects. The example of described method object separation is shown in Fig.9.

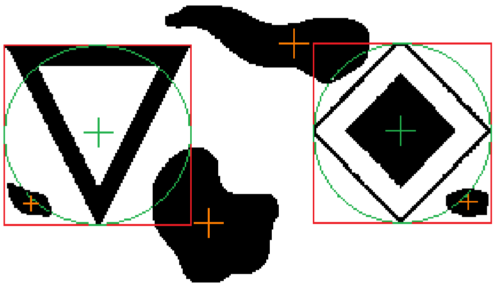


Fig. 9 Example of recognition basic points of image objects

Obtained object is then compared with correlation function used to all patterns. In this process the biggest matching of the object with patterns in percentage is expressed. The correct result is evaluated from the chosen pattern, which is the most identical to traffic sign in actual image. Practical tests have found that matching in more than 87% is sufficient to search traffic signs.

Learning the patterns number and rate of compliance is an adequate criterion for the correct formulation of the result recognition. Each number corresponds with specific patterns traffic signs.

Correlation function is used to determine the relationship between two signals (in this case signal represents the analysed image), the similarity of their histories, depending on their mutual displacement. Correlation can be expressed for linear and discrete signals. These signals cannot just be one-dimensional (vector) signal, but can be expressed by multidimensional signals. Correlation function of two signals is called a peer or cross-correlation functions. The result of correlation is a new signal, which has been displaced amplitude proportions have similar signals. This fact can be expressed as correlation coefficient R, which takes values between zero and one and reflects the similarity of two signals. For discreet binary two-dimensional signals can be expressed by the correlation coefficient R values range from zero to one.

$$R=\frac{\sum_{x=1}^{x}\sum_{y=1}^{y} f(x,y)\cdot g(x,y)}{x\cdot y} \qquad (5)$$

Function f(x,y) represented examined image and g(x,y) represented image pattern. Multiplying the correlation coefficient value by 100 represents the unity signal in percentage. Correlation is often used for the detection of known signals. [2]

VIII. Number recognition and analysis from image

Optical character recognition OCR is a method that enables the digitization of texts from retrieved images. The developed program converts the image either automatically or must learn to recognize characters. The converted text is almost always dependent on the quality of the draft should undergo thorough proofreading, because OCR program does not recognize all the letters correctly.

For the detection of traffic signs that inform about the maximum speed limit a simplified version of OCR algorithms are applied. OCR algorithm is applied to each traffic sign that is probably found with the help of the correlation function in comparison to the patterns. Segments of the traffic signs are systematically compared with known patterns of numbers. [3]

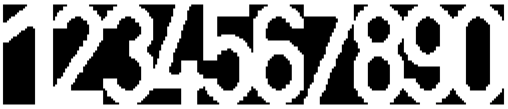


Fig. 10 Patterns for implemented OCR algorithm

Neural networks are useful for solving problems in image and signal recognition or diagnosis. The neural network is generally designed for a structure of spread parallel processing information, which consists of certain (usually very high) number of simple computing elements. Each element is named as the neuron. Neuron receives a finite number of inputs and their input information and passes its output to a finite number of outputs information. Formalized algorithm model of neuron is shown in Fig. 11.

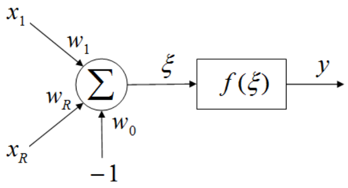


Fig. 11 Formalized algorithm model

Activity of this neuron can be expressed by:

$$y=f(\xi)\,, \qquad (6)$$

where y is input of neuron and ξ is so-called potential, which may be expressed by equation

$$\xi=\sum_{j=1}^{R} w_j x_j - w_0 \qquad (7)$$

In this relationship x_j symbol indicates the value of j-th entry of the neuron, w_j denotes the value of j-th entry, and w_0 denotes the threshold of neurons and the function $f(\xi)$ shows generally nonlinear signal transfer function of neurons.

Neurons are arranged in networks, this arrangement is known as network topology. Neural network does not contain any information about finding the object.

At the moment, neural methods and algorithms for OCR analyses are prepared, but implemented and successfully tested OCR recognition by correlation function is available. The problem of correlation function, compared to other method, is insufficient speed of algorithm execution, which increases with number of patterns.

V. Algorithms for Traffic Lanes Recognition

For the detection of lanes on a road, Hough transform is implemented. At the moment the implementation is solved in MATLAB environment and it is prepared for implementation to embedded system. This transformation is analytical method used to find a parametric description of the objects in the picture. This method is used for the detection of simple objects in the picture, which are example lines, circles and ellipses. The main advantage of this method for lane detection is robust against irregularities and infringements looking curve, which is particularly suitable for the detection of dashed centre line. To find a mathematical model of the object in a picture we used Hough transform as an input pixel input image. For example, detection of lines in the image edited by the Hough transform was used in the equation:

$$x \cdot \cos \theta + y \cdot \sin \theta = r \qquad (8)$$

Where:

r – Length of normal from the origin of coordinates to the line.

θ - Angle between the normal and the *x*.

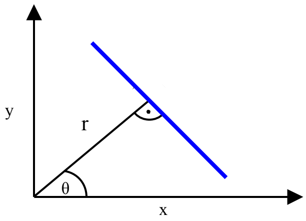


Fig. 12 Parametric description of detected traffic lane

Hough transform is applied here, by an adjustment of the input image. The adjustment consists of the image transfer to binary image, and finding local maxima using edge detector. One of the most important edge detection is Canny edge detector, which is implemented by algorithm for edge detection in two-dimensional discrete image.

VI. Conclusion

The main goal of this paper is to show development of car system for object recognition. The basic car system architecture and algorithms for traffic signs and lanes recognition are presented here. Neural networks were used in order to solve problems in image and signal recognition or diagnosis. The problem of image processing and object recognition was discussed in this contribution. We presented development stages of recognition system.

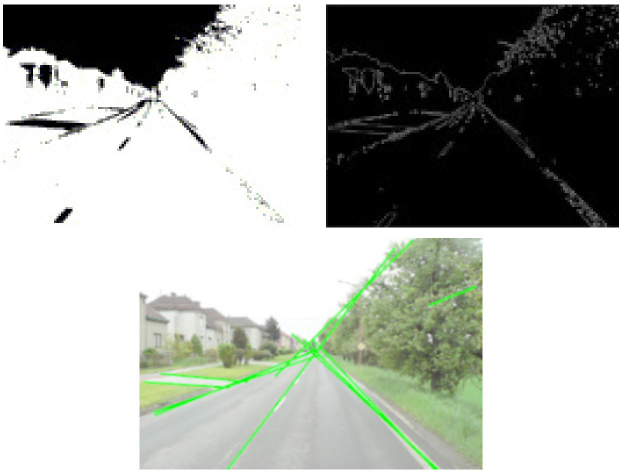


Fig. 13 Examples of road lanes recognition system implementation

Acknowledgment

This work was supported by projects GA102/08/1429 - Safety and security of networked embedded system applications.

References

[1] K. Arnold, Embedded Controller Hardware Design, San Diego, USA, 2001. 320 p. ISBN 978-1878707529

[2] J. D. Gibson, A. Bovik, Image and video processing, Academic Press, Orlando, USA, 2000. 891 p. ISBN 0121197905

[3] V. Hlavac, M. Sedlacek, Zpracování signálu a obrazu, BEN, Praha, CZ, 2007. 255 p. ISBN 978-80-01-03110-0.

[4] J. Kotzian, V. Srovnal Jr,: Distributed embedded system for ultralight airplane monitoring, ICINCO 2007, Intelligent Control Systems and Optimization, Anger, France, 2007 p.448-451 ISBN 978-972-8865-82-5

[5] T. Sridhar, Design Embedded Communications Software. CMP Books, San Francisco, USA, 2003, ISBN 1-57820-125-X.

[6] V. Srovnal Jr., Z. Machacek, V. Srovnal, Wireless Communication for Mobile Robotics and Industrial Embedded Devices, In proceedings ICN 2009 8th International Conference on Networks, Cancun, Mexico, 2009 p.253-258 ISBN 978-0-7695-3552-4

4rd International Workshop on Secure Information Systems

The SIS workshop is envisioned as a forum to promote the exchange of ideas and results addressing complex security issues that arise in modern information systems. We aim at bringing together a community of security researchers and practitioners working in such divers areas as networking security, antivirus protection, intrusion detection, cryptography, security protocols, and others. We would like to promote an integrated view at the security of information systems.

As information systems evolve, becoming more complex and ubiquitous, issues relating to security, privacy and dependability become more critical. At the same time, the global and distributed character of modern computing – typically involving interconnected heterogeneous execution environments – introduces many new and challenging engineering and scientific problems. Providing protection against increasingly sophisticated attacks requires strengthening the interaction between different security communities, e.g. antivirus and networking. New technologies lead to the emergence of new threats and attack strategies, involving smart mobile devices, peer-to-peer networks, instant messaging, VoIP, mesh networks or even networked consumer devices, such as house appliances or cars. Furthermore, the increasing openness of the communications infrastructure results in novel threats and can jeopardize critical enterprise and public infrastructure, such as industrial automation and process control systems. Not only it is estimated that half of all Web applications and Internet storefronts still contain some security vulnerabilities, but secure commerce applications are also exposed to increasingly elaborate attacks, including spyware, phishing and other social engineering methods.

In order to develop a secure system, security has to be considered in all phases of the lifecycle and adequately addressed in all layers of the system. At the same time, good engineering has to take into account both scientific and economic aspects of every solution: the cost of security has to be carefully measured against its benefits – in particular the expected cost of mitigated risks. Most companies and individuals treat security measures in information system as a necessary, but often uncomfortable, overhead. The increasing penetration of computing in all domains of everyday life means that security of critical business systems is often managed and maintained by personnel who are not knowledgeable in the field. This highlights the importance of usability and ease of configuration of security mechanism and protocols.

Covered topics include (but are not limited to):

- Access control
- Adaptive security
- Cryptography
- Copyright protection
- Cyberforensics
- Honeypots
- Intrusion detection
- Network security
- Privacy
- Secure commerce
- Security exploits
- Security policies
- Security protocols
- Security services
- Security evaluation and prediction
- Software protection
- Trusted computing
- Threat modeling
- Usability and security
- Viruses and worms
- Zero-configuration security mechanisms

Program Committee

Krzysztof Cabaj, Warsaw University of Technology, Poland, Poland

Costas Constantinou, University of Birmingham, UK, United Kingdom

Nicolas Courtois, University College London, UK, United Kingdom

Lech Janczewski, University of Auckland, New Zealand, New Zealand

Paul Kiddie, University of Birmingham, UK, United Kingdom

Jerzy Konorski, Gdansk University of Technology, Poland

Igor Kotenko, SPIIRAS, Russian Federation

Zbigniew Kotulski, Warsaw University of Technology, Poland and IPPT PAN, Poland, Poland

Bogdan Ksiezopolski, Maria Curie-Sklodowska University, Poland

Pascal Lafourcade, University Joseph Fourier, France

Shiguo Lian, France Telecom R&D Beijing, China

Ke Liao, Oklahoma university, USA

Guangjie Liu, Nanjing University of Science and Technology, China, China

Jozef Lubacz, Warsaw University of Technology, Poland, Poland

Norka Lucena, Syracuse University, USA, USA

Wojciech Mazurczyk, Warsaw University of Technology, Poland

Symeon Papavassiliou, National Technical University of Athens, Greece, Greece

Josef Pieprzyk, Macquarie University, Australia, Australia

Zbigniew Piotrowski, Military University of Technology, Poland and University College London, UK, Poland

Janusz Stoklosa, Poznan University of Technology, Poland, Poland

Krzysztof Szczypiorski (Chairman), Warsaw University of Technology, Poland

Konrad Wrona, NATO C3 Agency, Netherlands

Xiaoyi Yu, Peking University, China, China

Organizing Committee

Bogdan Ksiezopolski, Maria Curie-Sklodowska University, Poland

Wojciech Mazurczyk, Warsaw University of Technology, Poland

Krzysztof Szczypiorski (Chairman), Warsaw University of Technology, Poland

International Management and Scientific Committee

Konrad Wrona, NATO C3 Agency , Netherlands

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 839–845

ISBN 978-83-60810-27-9
ISSN 1896-7094

A Security Model for Personal Information Security Management Based on Partial Approximative Set Theory

Zoltán Csajbók
Department of Health Informatics
Faculty of Health, University of Debrecen
Sóstói út 2-4, H-4400 Nyíregyháza, Hungary
Email: csajzo@de-efk.hu

***Abstract*—Nowadays, computer users especially run their applications in a complex open computing environment which permanently changes in the *running* time. To describe the behavior of such systems, we focus *solely* on externally observable execution traces generated by the observed computing system. In these extreme circumstances the pattern of sequences of primitive actions (execution traces) which is observed by an external observer cannot be designed and/or forecast in advance. We have also taken into account in our framework that security policies are partial-natured.**

To manage the outlined problem we need tools which are approximately able to discover secure or insecure patterns in execution traces based on *presupposes* of computer users. Rough set theory may be such a tool. According to it, the vagueness of a subset of a finite universe U is defined by the difference of its lower and upper approximations with respect to a partition of the universe U. Using partitions, however, is a very strict requirement. In this paper, our starting point will be an *arbitrary* family of subsets of U. Neither that this family of sets covers the universe nor that the universe is finite will be assumed. This new approach is called the *partial approximative set theory*. We will apply it to build up a new security model for distributed software systems solely focusing on their externally observable executions and to find out whether the observed system is secure or not.

I. Introduction

NOWADAYS, computer end users especially run their applications in a complex open computing environment which as a rule consists of software components of finite numbers. Each component has an individual behavior, and the global behavior of the whole system is a collection of individual ones.

Personal users watch their applications, work with one of them, and, in general, also follow details of other applications with attention. Moreover, the open computing environment permanently changes in the *running* time. Consequently, the pattern of the sequence of primitive actions, which is observed by an external observer, cannot be designed and/or forecast in advance. Nevertheless, from an execution trace which is actually observed during a short observation time interval someone ought to decide whether the system works safely or not, of course, with respect to a certain security specification.[1]

Under these peculiar circumstances, organizations first of all have to understand what has to be protected, and why. The answers determine the choice of the organization's *security strategy* which is, in turn, expressed in *security policies* [6]. Security policies *prescribe* and *proscribe* behaviors of software systems specifying acceptable and unacceptable execution traces of applications.

In corporate information security management there have been many approaches for security policy specification. Traditionally, security policies are formulated along the so-called CIA taxonomy [18] which sees security as the combination of three attributes—confidentiality, integrity, and availability.

However, there are different challenges between corporate information security management and personal information security management. Under the personal information security we do not exclusively mean the privacy. Under the information security from personal point of view we also mean, among others, the protection of personal desktop PCs or notebooks from the intrusion, applications and data from the damage, etc.

Users in personal computing environment are inundated by recommendations how they should use and operate their system. In headwords only: strong password creation tips and maintenance, virus protection, software downloading and installation, removable media risks, encryption and cryptographic means, system backups, incident handling, e-mail and internet use best practices, etc. Non-professionals, of course, cannot convert these pieces of good advice into security policies, particularly into a formal one.

Arising from the human thinking [26], all computer users have *anticipated hypotheses* how an application or the whole computer system should or should not work. This may range from informal expected behaviors of the system, their elements might be called expected 'milestones', to more formal ones described in user manuals. Without any knowledge about

[1] This strange situation is smartly described by Schneier: "You have to imagine an intelligent and malicious adversary inside your system (the 'Satan' of Satan's computer), constantly trying new ways to subvert it. You have to think like an alien." ([3], from the Foreword by B. Schneier)

running application or having either some informal behaviors and/or more formal descriptions about it, we have to make an attempt to build up an information security management model which is approximately able to discover secure or insecure patterns in execution traces of the running system. Thus, in contrast to the traditional approach mentioned above, in order to describe the behavior of distributed software systems in a personal computing environment, we focus *solely* on externally observable executions generated by the observed computing system.

As usual, the software components can operate with each other. Their interconnections may be intended or *ad hoc*. Notice, however, that in both cases, the mechanisms of these interconnections mostly remain concealed from the external observers. In particular, based on only external observations, we cannot model these synchronization mechanisms.

II. Related Work

Our approach partly relates to the theory of a very special class of security policies called *properties* proposed by Lamport [20]. Informally, a security property is a set of execution traces which is defined exclusively in terms of individual execution traces, or, in other words, a property may not specify a relationship between possible execution traces of an application [21]. Access control or availability are properties, while information flow policy is typically not, the latter is a more general one. According to another terminology, properties are *point-wise* in nature, while the information flow policy is *point-free* in nature. Methods for specifying and reasoning about properties are well understood. Alpern and Schneider have shown [1] that every property is the intersection of a so-called *safety* and a *liveness* property. For more details about properties see, e.g., [20] [30], [21], [22].

We also partly relate to the theory of *hyperproperties* proposed by Clarkson and Schneider in [7]. In this approach *every* set of execution traces is a property. Thereupon, the hyperproperty is a set of properties, or simply, a set of sets of execution traces. In other words, both properties and hyperproperties are defined in a *point-wise* manner—properties based on their elements, the execution traces, and hyperproperties based on their elements, the properties.

Analogous to safety and liveness property, Clarkson and Schneider defined the notions of *hypersafety* and *hyperliveness*. They have proved that every hyperproperty is the intersection of a hypersafety and a hyperliveness [7]. This generalizes Alpern and Schneider's result about properties.

Both theories of properties and hyperproperties have a characterization in terms of topology. In the Plotkin topology on properties, safety and liveness correspond to *closed* and *dense* sets, respectively [1]. In [7], this topological characterization is generalized to hyperproperties, showing that hypersafety and hyperliveness also correspond to closed and dense sets in the lower Vietoris topology which is a construction on Plotkin topology [31], [32].

It is an interesting question of the relation between the traditional approaching to the security policies based on CIA taxonomy and the formulation of them based on properties or hyperproperties. This is an open question. Clarkson and Schneider [7] think that the language of confidentiality, integrity, and availability is orthogonal to hypersafety and hyperliveness. Whereas Benenson et al. worked out a framework which, in their opinion, offers a new perspective onto the CIA taxonomy [5]. Namely, to some extent, confidentiality has a large overlap with information flow, integrity has parallels with safety, and availability has some resemblance to liveness.

However, our approach to hyperproperties as sets of sets of execution traces is different from the Clarkson and Schneider's one [7]. Namely, in contrast to their notion, we do not consider *every* set of execution traces as a property. Our proposal is characteristically a *point-free* approach. We deal with the set of execution traces *per se*, and the same is true for hyperproperties. Of course, in some special cases, a point-free feature can coincide with a feature defined in a point-wise manner. Avoiding the ambiguities, in our framework we will not use the terms 'property' and 'hyperproperty'.

We model distributed software systems as semantic system model, so-called *traced-based model*. A traced-based model describes the behavior of a system as a set of execution traces where a trace models a possible execution sequence of the whole system or its some components. To our notion the Mazurkiewicz' trace theory stands the nearest [23], [13]. We, however, at least temporarily, use it in a very simple form.

III. Problem Statement

We briefly sum up our framework informally as follows.

We model security policies as sets of sets of execution traces. We also have taken into account in our framework that security policies are partial-natured. Typically some policies may apply only to a specific application or type of information. For example, the average response time policy would be practical to be applied to the whole observed system. But, in an open computing environment, we *cannot* influence the response time of unknown applications, such as a database's query via the internet. As another example, probably it is enough to enforce the information flow policy on such software processes which handle confidential information.

To manage the outlined problem we need tools which are approximately able to discover secure or insecure patterns in execution traces. By applying some sort of knowledge discovery method it is possible to some extent to reveal the secure or insecure nature of the observed system. However, the result, obtained from the execution traces in this way, cannot be precise (exact) due to the permanent changes of the open computing environment and insufficient knowledge about it.

Rough set theory is a relatively new data-mining technique used in the discovery of patterns within data. This theory provides a powerful foundation to reveal and discover important structures in data and to classify complex objects. One of the main advantages of rough set theory is that it does not need any preliminary or additional information about data.

The rough set theory was introduced by the Polish mathematician, Z. Pawlak in the early 1980s [27], [28]. It was a new mathematical approach to *vagueness* [29]. According to Pawlak's idea, the vagueness of a subset of a finite universe U is defined by the difference of its upper and lower approximations with respect to a partition of the universe U.

Using partitions, however, is a very strict requirement. In this paper, our starting point will be an *arbitrary* family of subsets of an *arbitrary* set U. Neither that this family of sets covers the universe nor that the universe is finite will be assumed. Within this new framework, our concepts of lower and upper approximations are straightforward *point-free* generalizations of Pawlak's ones [9]. This is called the *partial approximative set theory*. This new approach suits the partial nature of security policies, as well. We will apply this theory to build up a new security model for distributed software systems *solely* focusing on their externally observable executions and to find out whether the observed system is secure or not.

The rest of the paper is organized as follows. In Section IV we summarize the basic notations. Section V will outline a general approximation framework. Sections VI and VII briefly present the basic principles of the rough set theory and the partial approximative set theory, respectively. In Section VIII we will describe the behavior of distributed software systems by means of partial approximative set theory.

IV. Basic Notations

Let U be any set. Let $\mathfrak{A} \subseteq 2^U$ be a family of sets of which elements are subsets of U. The union of $\mathfrak{A}$ is $\bigcup\mathfrak{A} = \{x \mid \exists A \in \mathfrak{A}(x \in A)\}$, and the intersection of $\mathfrak{A}$ is $\bigcap\mathfrak{A} = \{x \mid \forall A \in \mathfrak{A}(x \in A)\}$. If $\mathfrak{A}$ is an empty family of sets we define $\bigcup\emptyset = \emptyset$ and $\bigcap\emptyset = U$.

The number of elements of the set U is denoted by $\#U$.

If ϵ is an arbitrary binary relation on U, let $[x]_\epsilon$ denote the ϵ-related elements to x, i.e., $[x]_\epsilon = \{y \in U \mid (x, y) \in \epsilon\}$.

Let $\mathcal{A}$ be a finite set of *symbols* called the *alphabet*. A *string* is a finite or infinite sequence of symbols chosen from $\mathcal{A}$. We denote strings by small Greek letters $\sigma, \varrho, \tau, \ldots$, with possible sub- or superscripts. String containing no symbols is called the *empty string* and is denoted by λ.

If $\sigma = a_1a_2\ldots a_n$ is a finite string, $n = |\sigma| = |a_1a_2\ldots a_n|$ is called the *length* of σ. In particular, $|\lambda| = 0$.

Let $\mathcal{A}^*$ and $\mathcal{A}^\omega$ denote the set of all finite and infinite strings made up of symbols chosen from $\mathcal{A}$, respectively. We also use the following notations: $\mathcal{A}^+ = \mathcal{A}^* \setminus \{\lambda\}$, $\mathcal{A}^\infty = \mathcal{A}^* \cup \mathcal{A}^\omega$.

For any finite strings $\sigma_1 = a_1a_2\ldots a_n$, $\sigma_2 = b_1b_2\ldots b_m$, the *concatenation* $\sigma_1 \circ \sigma_2 = a_1a_2\ldots a_n \circ b_1b_2\ldots b_m$ of σ_1 and σ_2 is the string $\sigma_1\sigma_2 = a_1a_2\ldots a_nb_1b_2\ldots b_m$. If $\sigma_1 \in \mathcal{A}^*$, $\sigma_2 \in \mathcal{A}^\omega$ then $\sigma_1 \circ \sigma_2 = \sigma_1\sigma_2$ is also defined. The empty string is the identity for concatenation, that is for any string $\sigma_1 \in \mathcal{A}^*, \sigma_2 \in \mathcal{A}^\omega$, we have $\lambda\sigma_1 = \sigma_1\lambda = \sigma_1$ and $\lambda\sigma_2 = \sigma_2$.

A symbol $a \in \mathcal{A}$ *occurs* in string $\sigma \in \mathcal{A}^\infty$, if $\sigma = \sigma_1 a \sigma_2$ for some strings $\sigma_1 \in \mathcal{A}^*, \sigma_2 \in \mathcal{A}^\infty$.

Let $\sigma \in \mathcal{A}^*$, $\tau \in \mathcal{A}^\infty$ be two strings. σ is a *prefix* (initial part) of τ if there exists $\rho \in \mathcal{A}^\infty$ such that $\tau = \sigma\rho$. The prefix is *proper* if $\rho \neq \lambda$ and *nontrivial* if $\sigma \neq \lambda$.

Given $L \subseteq \mathcal{A}^\infty$, let $\mathtt{pref}(L)$ denote the set of all prefixes of all strings in L. Clearly, $L \subseteq \mathtt{pref}(L)$. In particular, $\mathtt{pref}(\{\sigma\}) = \mathtt{pref}(\sigma)$ is the set of all prefixes of the string $\sigma \in \mathcal{A}^\infty$. Note that $\mathtt{pref}(L) = \bigcup\{\mathtt{pref}(\sigma) \mid \sigma \in L\} \subseteq \mathcal{A}^*$.

Let $\mathcal{A}$ be an alphabet and σ be a finite string over an arbitrary alphabet. Then $\pi_{\mathcal{A}}(\sigma)$ denotes the (string) *projection of σ onto $\mathcal{A}$* defined as follows:

$$\pi_{\mathcal{A}}(\sigma) = \begin{cases} \lambda, & \text{if } \sigma = \lambda; \\ \pi_{\mathcal{A}}(\sigma'), & \text{if } \sigma = \sigma' a \wedge a \notin \mathcal{A}; \\ \pi_{\mathcal{A}}(\sigma')a, & \text{if } \sigma = \sigma' a \wedge a \in \mathcal{A}. \end{cases}$$

Informally, a projection onto $\mathcal{A}$ cleans strings of all symbols not in $\mathcal{A}$. $\pi_{\mathcal{A}}(\sigma) \neq \lambda$, if at least one symbol $a \in \mathcal{A}$ occurs in string σ.

Similarly, if $\Sigma \subseteq \mathcal{A}^*$, $\pi_{\mathcal{A}}(\Sigma) = \{\pi_{\mathcal{A}}(\sigma) \mid \sigma \in \Sigma\}$. $\pi_{\mathcal{A}}(\Sigma) \neq \{\lambda\}$, if in at least one string in $\sigma \in \Sigma$, at least one symbol $a \in \mathcal{A}$ occurs in σ.

V. A General Approximation Framework

In order that the vagueness can be treated in a general approximate framework, let our initial concept be the following. A pair of maps $f, g : 2^U \to 2^U$ is a *weak approximation pair on U* if

$$\forall X \in 2^U \ (f(X) \subseteq g(X)).$$

As Düntsch and Gediga noticed in [14], this constraint seems to be the weakest condition for a sensible concept of approximations of subsets in U.

The maps f, g is a *strong approximation pair on U* if each subset $X \in 2^U$ is bounded by $f(X)$ and $g(X)$ [14], i.e.,

$$\forall X \in 2^U \ (f(X) \subseteq X \subseteq g(X)).$$

In [26], a new hypothesis about approximation was drawn up recently. According to this assumption, the notion of "approximation" may be mathematically modelled by the notion of Galois connections. From now on, we call it the *approximation hypothesis*.

Let $(P, \leq_P)$ and $(Q, \leq_Q)$ be two posets.

Definition 1. A pair (f, g) of maps $f : P \to Q$, $g : Q \to P$ is a *(regular) Galois connection* or an *adjunction* between P and Q if

$$\forall p \in P \forall q \in Q \ (f(p) \leq_Q q \Leftrightarrow p \leq_P g(q)).$$

f is called the *lower adjoint* and g the *upper adjoint* of the Galois connection.

We also write (P, f, g, Q) for a whole Galois connection. If $P = Q$ it is said (P, f, g, P) is a Galois connection on P.

Remark 2. Here we adopted the definition of Galois connection in which the maps are monotone. It is also called monotone or covariant form. For more details, see, e.g., [11], [12], [15], [17].

VI. Fundamentals of Rough Set Theory

The basic concepts and properties of rough set theory can be found, e.g, in [28], [19]. Here we cite only a few of them which will be important in what follows. We partly restate these well-known facts on the language of approximations.

Definition 3. A pair (U, ε), where U is a finite universe of discourse and ε is an equivalence relation on U, is called *Pawlak's approximation space*.

A subset $X \subseteq U$ is *ε-definable*, if it is a union of ε-elementary sets, otherwise X is *ε-undefinable*. By definition, the empty set is considered to be an ε-definable set.

Let $\mathfrak{D}_{U/\varepsilon}$ denote the family of ε-definable subsets of U.

In Pawlak's approximation spaces, the lower and upper approximations of X can be defined in two equivalent forms, namely, in a *point-free* manner—based on the ε-elementary sets, and in a *point-wise* manner—based on the elements.

Definition 4. Let a Pawlak's approximation space (U, ε), a subset $X \in 2^U$ be given. The *lower ε-approximation* of X is

$$\begin{aligned} \underline{\varepsilon}(X) &= \bigcup\{Y \mid Y \in U/\varepsilon, Y \subseteq X\} \\ &= \{x \in U \mid [x]_\varepsilon \subseteq X\}, \end{aligned}$$

and the *upper ε-approximation* of X is

$$\begin{aligned} \overline{\varepsilon}(X) &= \bigcup\{Y \mid Y \in U/\varepsilon, Y \cap X \neq \emptyset\} \\ &= \{x \in U \mid [x]_\varepsilon \cap X \neq \emptyset\}. \end{aligned}$$

The set $B_\varepsilon(X) = \overline{\varepsilon}(X) \setminus \underline{\varepsilon}(X)$ is the *ε-boundary* of X. X is *ε-crisp*, if $B_\varepsilon(X) = \emptyset$, otherwise X is *ε-rough*.

Based on binary relations on U, lower and upper ε-approximations can be generalized via their *point-wise* definitions [19].

Definition 5. Let ϵ be an arbitrary binary relation on U and $X \in 2^U$. The *lower ϵ-approximation* of X is

$$\underline{\epsilon}(X) = \{x \in U \mid [x]_\epsilon \subseteq X\},$$

and the *upper ϵ-approximation* of X is

$$\overline{\epsilon}(X) = \{x \in U \mid [x]_\epsilon \cap X \neq \emptyset\}.$$

If ϵ^{-1} denotes the inverse relation of ϵ, in the same manner one can also define lower and upper ϵ^{-1}-approximations.

Theorem 6 ([19], Proposition 134)**.** *Let ϵ be an arbitrary binary relation on U. Then the pairs $(\overline{\epsilon}, \underline{\epsilon^{-1}})$ and $(\overline{\epsilon^{-1}}, \underline{\epsilon})$ are Galois connections on $(2^U, \subseteq)$.*

Some other properties of lower and upper ϵ-approximations are expressed by some properties of binary relations, and vice versa.

Theorem 7. *Let ϵ be an arbitrary binary relation on U.*

1) *The pair $(\underline{\epsilon}, \overline{\epsilon})$ is a weak approximation pair if and only if ϵ is connected.*
2) *The pair $(\underline{\epsilon}, \overline{\epsilon})$ is a strong approximation pair if and only if ϵ is reflexive.*
3) *The pair $(\overline{\epsilon}, \underline{\epsilon})$ is a Galois connection on $(2^U, \subseteq)$ if and only if ϵ is symmetric.*

Proof: In [19]. *1.* Proposition 136., *2.* Proposition 137., *3.* Proposition 138. ■

It can be shown that even if the relation ϵ is symmetric, it is not sufficient that the lower and upper ϵ-approximations defined in a *point-free* manner form a Galois connection [10].

VII. Fundamentals of Partial Approximative Set Theory

In practice there are attributes which do not characterize all members of an observed collection of objects. A very simple example, when one investigates an infinite set via a finite family of its finite subsets. For instance, a number theorist studies regularities of natural numbers using computers.

Throughout this section let U be any non-empty set.

Definition 8. Let $\mathfrak{B} \subseteq 2^U$ be a non-empty family of non-empty subsets of U called the *base system*. Its elements are the *$\mathfrak{B}$-sets*.

A family of sets $\mathfrak{D} \subseteq 2^U$ is *$\mathfrak{B}$-definable* if its elements are $\mathfrak{B}$-sets, otherwise $\mathfrak{D}$ is *$\mathfrak{B}$-undefinable*. A non-empty subset $X \in 2^U$ is *$\mathfrak{B}$-definable* if there exists a $\mathfrak{B}$-definable family of sets $\mathfrak{D}$ such that $X = \bigcup \mathfrak{D}$, otherwise X is *$\mathfrak{B}$-undefinable*. The empty set is considered to be a $\mathfrak{B}$-definable set.

Let $\mathfrak{D}_{\mathfrak{B}}$ denote the family of $\mathfrak{B}$-definable sets of U.

Definition 9. Let $\mathfrak{B} \subseteq 2^U$ be a base system and X be any subset of U. The *weak lower $\mathfrak{B}$-approximation* of X is

$$\mathfrak{C}^\flat_{\mathfrak{B}}(X) = \bigcup\{Y \mid Y \in \mathfrak{B}, Y \subseteq X\},$$

and the *weak upper $\mathfrak{B}$-approximation* of X is

$$\mathfrak{C}^\sharp_{\mathfrak{B}}(X) = \bigcup\{Y \mid Y \in \mathfrak{B}, Y \cap X \neq \emptyset\}.$$

Notice that $\mathfrak{C}^\flat_{\mathfrak{B}}$ and $\mathfrak{C}^\sharp_{\mathfrak{B}}$ are straightforward *point-free* generalizations of lower and upper ε-approximations.

Theorem 10 ([10], Theorem 4.5)**.** *Let the fixed base system $\mathfrak{B} \subseteq 2^U$ and maps $\mathfrak{C}^\flat_{\mathfrak{B}}$ and $\mathfrak{C}^\sharp_{\mathfrak{B}}$ be given.*

1) $\forall X \in 2^U (\mathfrak{C}^\flat_{\mathfrak{B}}(X) \subseteq \mathfrak{C}^\sharp_{\mathfrak{B}}(X))$.
2) $\forall X \in 2^U (\mathfrak{C}^\flat_{\mathfrak{B}}(X) \subseteq X)$*—that is, $\mathfrak{C}^\flat_{\mathfrak{B}}$ is contractive.*
3) $\forall X \in 2^U (X \subseteq \mathfrak{C}^\sharp_{\mathfrak{B}}(X))$ *if and only if $\bigcup \mathfrak{B} = U$—that is, $\mathfrak{C}^\sharp_{\mathfrak{B}}$ is extensive if and only if $\mathfrak{B}$ covers the universe.*

In other words, the pair of maps $\mathfrak{C}^\flat_{\mathfrak{B}}, \mathfrak{C}^\sharp_{\mathfrak{B}} : 2^U \to 2^U$ is a weak approximation pair on U, and it is a strong one if and only if the base system $\mathfrak{B}$ covers the universe.

Remark 11. If $\bigcup \mathfrak{B} \neq U$, then $\forall X \subseteq U \setminus \bigcup \mathfrak{B}\ \forall B \in \mathfrak{B}(X \cap B = \emptyset)$. Consequently, for all these subsets $\mathfrak{C}^\sharp_{\mathfrak{B}}(X) = \bigcup \emptyset = \emptyset$, i.e., the empty set is the weak upper $\mathfrak{B}$-approximation of certain non-empty subsets of U. This uncommon case may be interpreted so that our knowledge about the universe encoded in the base system $\mathfrak{B}$ is incomplete. This case may be excluded by a partial map called strong upper $\mathfrak{B}$-approximation. For more details, see [9].

Definition 12. Let the fixed base system $\mathfrak{B} \subseteq 2^U$ and maps $\mathfrak{C}^\flat_{\mathfrak{B}}$ and $\mathfrak{C}^\sharp_{\mathfrak{B}}$ be given. The quadruple $(U, \mathfrak{B}, \mathfrak{C}^\flat_{\mathfrak{B}}, \mathfrak{C}^\sharp_{\mathfrak{B}})$ is called a *weak $\mathfrak{B}$-approximation space*.

Theorem 13 ([10], Corollary 4.10)**.** *Let the weak $\mathfrak{B}$-approximation space $(U, \mathfrak{B}, \mathfrak{C}^\flat_{\mathfrak{B}}, \mathfrak{C}^\sharp_{\mathfrak{B}})$ be given.*

The pair of maps $(\mathfrak{C}^\sharp_{\mathfrak{B}}, \mathfrak{C}^\flat_{\mathfrak{B}})$ forms a Galois connection on $(2^U, \subseteq)$ if and only if the base system $\mathfrak{B}$ is a partition of the universe U.

VIII. The Security Model

An execution sequence consists of linearly ordered *observable* atomic actions. The manner in which execution sequences are represented is irrelevant, they may be finite sequences of primitive events, higher-level system steps, program states, state/action pairs, etc., or a mixture of all these [4], [30].

A. Types of Atomic Actions

Let $A_i = \{a_1, a_2, \ldots, a_{i_m}\}$ be a finite set of externally observable atomic actions of the i^{th} components of the distributed software system called the i^{th} *required component action set* (alphabet) ($i = 1 \ldots, n$). The required component action sets are not necessary pairwise disjoint, and the common atomic actions in different alphabets are undistinguishable. An *execution trace* (string) $\sigma \in A_i^\infty$ is a finite or infinite sequence of not necessarily different atomic actions.

Let A_{unsafe} be a finite set of insecure actions, called the *unsafe action set*, which may happen during the running time of the observed system.

$\bigcup A_i = \bigcup_{i=1}^n A_i = A_1 \cup A_2 \cup \ldots \cup A_n$ has to be augmented by a number of additional atomic actions which may or may not influence the safety of the system. Their set is called the *doubtful action set* and denoted by $A_{doubtful}$. For instance, usual interactions between end users and peripheral devices probably do not influence the safety of the system. But, too frequent interactions between them may indicate an unsafe behavior.

Note that the required component actions sets A_i are component specific, while the unsafe and doubtful action sets are not.

Let $\mathcal{A} = \bigcup A_i \cup A_{doubtful} \cup A_{unsafe}$, called the *(system) action set.* Let $\mathcal{A}_i = A_i \cup A_{doubtful} \cup A_{unsafe}$ denote the finite action set correspond to the i^{th} component called the i^{th} *component action set* ($i = 1, 2, \ldots, n$).

B. Anticipated Behaviors of Applications

We model the users' anticipated behaviors of a software component by a set of acceptable (feasible) finite linearly ordered sequences of the expected 'milestones' from A_i, in informal cases, or finite linearly ordered sequences of the required component actions corresponding to a given precedence dependency on A_i, in formal cases.

In latter case, let us suppose that the actions in a required component action set A_i have to be performed with respect to the ordering constraints called 'precedence' dependencies. Two actions are precedentially ordered if one has to precede the other specifying the precedential ordering such that starting a certain action depends on other actions being completed beforehand. For a simple example, one must enter into an application before quitting it. In symbols, $A_i = \{\text{'}enter\_into\_application\text{'}, \ldots, \text{'}quit\_application\text{'}\}$ and '$enter\_into\_application$' $\leq$ '$quit\_application$'. Of course, there may be many other precedences that must be obeyed.

We can represent these precedences by a set consisting of the pairs of type ('$before\_action$', '$after\_action$'). This situation is usually modelled as a *scheduling problem* which is a common source of partial orders. That is, the precedence dependency relation is reflexive, transitive, and antisymmetric, and renders the required action set A_i to a poset $(A_i, \leq)$. It is assumed that the precedence dependencies is configurable in this way and it is captured by the system architect in a precedential model.

In formal cases, let $\overrightarrow{A_i}$ denote the set of all topological sorts of the poset $(A_i, \leq)$. By the Szpilrajn's Theorem [33], $\#\overrightarrow{A_i} \geq 1$. An element of $\overrightarrow{A_i}$ can be represented as a finite execution $a_{i_1} a_{i_2} \ldots a_{i_m}$ of all elements of A_i in such a way that for all $i_k < i_l$, either $a_{i_k} \leq a_{i_l}$ or a_{i_k} and a_{i_l} are incomparable in $\leq$. This means that a topological sort in $\overrightarrow{A_i}$ is consistent with the precedence dependencies defined on A_i.

Similarly, in informal cases, let $\overrightarrow{A_i}$ denote simply the set of all acceptable (feasible) finite linearly ordered sequences of the expected 'milestones' from A_i.

Note that, $\overrightarrow{A_i} \subseteq A_i^*$, in both cases.

C. Modelling Software Systems and their Components

A distributed software system is modelled by a non-empty set of *infinite* execution traces over the system action set $\mathcal{A}$.

Definition 14. Let $\Sigma \subseteq \mathcal{A}^\omega$ be a non-empty set of infinite executions over the action set $\mathcal{A}$. Then, by a *distributed software system* we mean an $(\mathcal{A}, \Sigma)$ pair.

Similarly, a software component or application is a non-empty set of *infinite* execution traces over the i^{th} component action set $\mathcal{A}_i$.

Definition 15. Let $\Sigma_i \subseteq \mathcal{A}_i^\omega$ be a non-empty set of infinite executions over the action set $\mathcal{A}_i$. Then, by a *software component* or an *application* we mean an $(\mathcal{A}_i, \Sigma_i)$ pair ($i = 1, 2, \ldots, n$).

If a software system or its a component execution terminates, i.e., it is representable by finite execution trace, we represent it as an infinite execution trace by infinitely stuttering the empty action λ. The empty action λ represents the fact that the software system or its any application has not started yet, or their running has already finished.

D. Modelling Observations

An observation of a distributed software system is modelled by a non-empty set of *finite* execution traces over the system actions set $\mathcal{A}$.

Definition 16. Let $\Sigma^{obs} \subseteq \mathcal{A}^*$ be a non-empty finite set of finite execution traces over the system action set $\mathcal{A}$.

Then, by a *system observation* we mean an $(\mathcal{A}, \Sigma^{obs})$ pair.

Observations on components can be analogously defined over $\mathcal{A}_i$ ($i = 1, \ldots, n$).

Definition 17. Let $\Sigma_i^{obs} \subseteq \mathcal{A}_i^*$ be a non-empty finite set of finite executions traces over the component action set $\mathcal{A}_i$.

Then, by a *component observation* or an *application observation* we mean an $(\mathcal{A}_i, \Sigma_i^{obs})$ pair ($i = 1, 2, \ldots, n$).

In particular, Σ^{obs} and Σ_i^{obs} may consists of exactly one execution trace.

E. Modelling Security

According to the expected behavior of the system, an application is *acceptable* if it fully meets the requirements of the precedential model (prescription).

The component which contains at least one unsafe action is *unacceptable* (proscription). However, the transition from the ‘acceptable’ decision to the ‘unacceptable’ decision is a *vagueness* problem.

For the i^{th} component ($i = 1, \ldots, n$) we define

$$\begin{aligned}\mathfrak{S}_i &= \mathtt{pref}(\overrightarrow{A_i}) \\ &= \{\mathtt{pref}(\sigma_{i_j}) \mid \sigma_{i_j} \in \overrightarrow{A_i},\, i_j = 1, \ldots, \#\overrightarrow{A_i}\},\end{aligned}$$

where $\mathtt{pref}(\sigma_{i_j})$ is the set of all prefixes of either a topological sort σ_{i_j} of the poset $(A_i, \leq)$ in formal cases or a linearly ordered sequence of the expected ‘milestones’ in informal cases.

Let $\mathfrak{S} = \bigcup_{i=1}^{n} \mathfrak{S}_i$. Clearly, $\mathfrak{S} \subseteq 2^{A^*} \subseteq 2^{\mathcal{A}^*}$.

The sets in $\mathfrak{S} \subseteq 2^{\mathcal{A}^*}$ are not necessarily pairwise disjoint, and, in general, $\bigcup \mathfrak{S}$ does not cover $\mathcal{A}^*$. In other words, $\mathfrak{S}$ *is a base system over the universe* $\mathcal{A}^*$ *by the terminology of the partial approximative set theory.*

F. The Vagueness of Acceptability

Let an observation set of the i^{th} application Σ_i^{obs} be given.

If Σ_i^{obs} includes at least one finite execution trace over $\mathcal{A}_i$ in which at least one unsafe action occurs, i.e.,

$$\pi_{A_{unsafe}}(\Sigma_i^{obs}) \neq \{\lambda\},$$

the observed action set Σ_i^{obs} is *unacceptable*.

Let us assume, that the observed action set does not include unsafe action, i.e., $\pi_{A_{unsafe}}(\Sigma_i^{obs}) = \{\lambda\}$. By based on only this observation set Σ_i^{obs}, can the observed application be considered acceptable or not? This problem, as we also mentioned above, is a vagueness problem.

In order to answer this question, let us form the lower and upper approximations of $\mathtt{pref}(\Sigma_i^{obs})$ with respect to the base system $\mathfrak{S}$.

The lower $\mathfrak{S}$-approximation of $\mathtt{pref}(\Sigma_i^{obs})$ is

$$\mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs})) = \bigcup\{S \in \mathfrak{S} \mid S \subseteq \mathtt{pref}(\Sigma_i^{obs})\},$$

and the upper $\mathfrak{S}$-approximation of $\mathtt{pref}(\Sigma_i^{obs})$ is

$$\mathfrak{C}^{\sharp}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs})) = \bigcup\{S \in \mathfrak{S} \mid S \cap \mathtt{pref}(\Sigma_i^{obs}) \neq \emptyset\}.$$

By means of lower and upper $\mathfrak{S}$-approximation, the set $\mathtt{pref}(\Sigma_i^{obs})$ of all prefixes of observed execution traces may be approximated, consequently, *the safety of the observed action set* Σ_i^{obs} *can be estimated to a certain degree, by the information encoded in* $\mathfrak{S}$.

$\mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ and $\mathfrak{C}^{\sharp}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ can be interpreted as a security evaluation of the system working at the moment of the end of an observation time interval.

The execution traces in $\mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ can be classified with certainty as members of $\mathtt{pref}(\Sigma_i^{obs})$. Formally, all elements of $\mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ suit an element in $\mathtt{pref}(\overrightarrow{A_i})$, or, in other words, a prefix of an execution trace in $\overrightarrow{A_i}$. Its elements can be interpreted as certainty safe execution traces of the observation set Σ_i^{obs} with respect to $\mathfrak{S}$, i.e., they fully correspond to the users’ expectations.

$\mathtt{pref}(\Sigma_i^{obs}) \setminus \mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ may include, e.g., action sequences that miss some required actions or contain required actions in wrong order. It may also include right ordered action sequences which, however, contain doubtful actions between two required actions.

The execution traces in $\mathfrak{C}^{\sharp}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ are not guaranteed that all of them are members of $\mathtt{pref}(\Sigma_i^{obs})$. It may happen, e.g., that an execution sequence for a while suits a prefix of an execution trace in $\overrightarrow{A_i}$, but the next action is not a required one, it may be missing or it is a doubtful one.

By the partial feature of the base system $\mathfrak{S}$, it may take place that $\mathtt{pref}(\Sigma_i^{obs}) \nsubseteq \mathfrak{C}^{\sharp}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$. It may emphasize that our knowledge about the system encoded in the base system $\mathfrak{S}$ is not enough to approximate $\mathtt{pref}(\Sigma_i^{obs})$.

The situation $\mathtt{pref}(\Sigma_i^{obs}) \subseteq \mathfrak{C}^{\sharp}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs}))$ can be interpreted so that all observed execution traces for a while suit a prefix of an execution trace in $\overrightarrow{A_i}$, but some of them has a wrong continuation or the next action is a doubtful one.

Of course, $\mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs})) \subseteq \mathtt{pref}(\Sigma_i^{obs})$. If $\mathfrak{C}^{\flat}_{\mathfrak{S}}(\mathtt{pref}(\Sigma_i^{obs})) = \mathtt{pref}(\Sigma_i^{obs})$, it means that all elements of the observation set Σ_i^{obs} can be viewed as safe execution sequences with respect to the users’ expectations.

IX. Conclusions and Future Work

We have shown a framework in which many questions corresponding to security features of distributed software systems can be represented uniformly.

The presented model is a theoretical outline. The next most important task is to work out a partial approximative information system model analogous to Pawlak’s one [28]. This will enable to set up models which can be applied to solving practical problems. Such a model could be a base on which expert systems for enforcement of security policies can be built up.

Theorem 13 shows that the pair of weak lower and upper $\mathfrak{B}$-approximations forms a Galois connection only in a very special case. This restriction can be partly exceeded by using the notion of partial Galois connection [24].

Acknowledgment

The author would like to thank Tamás Mihálydeák for valuable discussions and suggestions.

References

[1] Alpern, B.,” Schneider, F., *Defining liveness*. Inf. Process. Lett. 21, 4 (Oct.), 181–185, 1985.

[2] Alpern, B.,” Schneider, F., *Recognizing safety and liveness*. Distributed Computing, 2:117–126, 1987.

[3] Anderson, R., *Security Engineering: A Guide to Building Dependable, Distributed Systems*. First edition. Wiley Computer Publishing, 2001.

[4] Bauer, L.,” Ligatti, J.,” Walker, D., *More enforceable security policies*. In Cervesato, I. (Ed.), Foundations of Computer Security: Proceedings of the FLoC’02 workshop on Foundations of Computer Security (2002), pp. 95–104.

[5] Benenson, Z.," Freiling, F.C.," Holz, Th.," Kesdogan, D.," Penso, L.D.: *Safety, Liveness, and Information Flow: Dependability Revisited*. In: Karl, W., Becker, J., Gropietsch, K.E., Hochberger, Ch., Maehle, E. (eds.) ARCS 2006—19th International Conference on Architecture of Computing Systems, Workshops Proceedings, pp. 56–65. Frankfurt am Main, Germany (2006)

[6] Caelli, W.," Longley, D.," Shain, M., *Information security handbook*, Stockton Press, New York, 1991.

[7] Clarkson, M. R.," Schneider, F.B., *Hyperproperties*, Proceedings 21st IEEE Computer Security Foundations Symposium (Pittsburgh, PA, June 2008), IEEE Computer Society (2008), pp. 51–65.

[8] Cousot, P.," Cousot, R., *Abstract interpretation and application to logic programs*, Journal of Logic Programming, 13(2–3):103–179, 1992. http://www.di.ens.fr /∼cousot/COUSOTpapers/JLP92.shtml

[9] Csajbók, Z., *Partial Approximative Set Theory*, In Programs, Proofs, Processes, Sixth Conference on Computability in Europe, CiE 2010, Ponta Delgada (Azores), Portugal, June 30 - July 4, 2010, Abstract and Handout Booklet, 2010. To appear.

[10] Csajbók, Z., *Partial Approximative Set Theory: A View from Galois Connections*, In Kovács, Emöd (ed.) et al., Proceedings of the 8th international conference on applied informatics (ICAI 2010), January 27–30, 2010, Eger, Hungary. Eger: Eszterházy Károly College. To appear.

[11] Davey, B. A.," Priestley, H. A.: *Introduction to Lattices and Order*, Second edition, Cambridge University Press, Cambridge, 2002.

[12] Denecke, K.," Erné, M.," Wismath, S.L., *Galois Connections and Applications*, Kluwer Academic Publishers, Dordrecht, London, Boston, 2004.

[13] Diekert, V.," Rozenberg, G., *The Book of Traces*, World Scientific, Singapore, 1995.

[14] Düntsch, I.," Gediga, G., *Approximation Operators in Qualitative Data Analysis*, In: H. de Swart et al. (Eds.): TARSKI, Lecture Notes in Computer Science Vol. 2929, Springer-Verlag, Berlin, Heidelberg, 214–230, 2003.

[15] Erné, M.," Koslowski, J.," Melton, A.," Strecker, G. E., *A primer on Galois connections*. In: S. Andima et al. (eds.), Papers on General Topology and its Applications. 7th Summer Conf. Wisconsin. Annals New York Acad. Sci. **704**, New York (1994), pp. 103-125.

[16] Focardi, R.," Gorrieri, R., *Classification of security properties (Part I: Information flow)*. In Foundations of Security Analysis and Design 2000, volume 2171 of Lecture Notes in Computer Science, pages 331-396. Springer, 2001.

[17] Gierz, G.," Hofmann, K. H.," Keimel, K.," Lawson, J. D.," Mislove, M.," Scott, D.S., *Continuous Lattices and Domains*, Encyclopedia of Mathematics and its Applications 93, Cambridge University Press, 2003.

[18] *Information Technology Security Evaluation Criteria (ITSEC)*. Version 1.2, Juni 1991.

[19] Järvinen, J., *Lattice theory for rough sets*, In: *Transactions on Rough Sets VI.* LNCS, vol. 4374, Springer, Heidelberg, 2007, pp. 400–498.

[20] Lamport, L., *Proving the correctness of multiprocess programs*. IEEE Transactions of Software Engineering, 3(2):125-143, Mar. 1977.

[21] Ligatti, J.," Bauer, L.," Walker, D., *Edit Automata: Enforcement mechanisms for run-time security policies*. International Journal of Information Security, 4(1-2):2-16, February 2005

[22] Ligatti, J. A., *Policy Enforcement via Program Monitoring*. PhD thesis, Princeton University, June 2006.

[23] Mazurkiewicz, A., *Trace Theory*, In W. Brauer et al., editors, Petri Nets, Applications and Relationship to other Models of Concurrency, number 255 in Lecture Notes in Computer Science, pages 279-324, Berlin-Heidelberg-New York, 1987, Springer

[24] Miné, A., *Weakly relational numerical abstract domains*, Ph. D. thesis, École Polythechnique, 2004.

[25] Naldurg, P.," Campbell, R.H.," Mickunas, M.D., *Developing Dynamic Security Policies*, Proceedings of the 2002 DARPA Active Networks Conference and Exposition (DANCE 2002), San Francisco, CA, USA, IEEE Computer Society Press, May 29-31, 2002

[26] Pagliani, P. and Chakraborty, M., *A Geometry of Approximation. Rough Set Theory: Logic, Algebra and Topology of Conceptual Patterns*, Trends in Logic, Vol. 27, Springer, 2008.

[27] Pawlak, Z., *Rough Sets*, International Journal of Information and Computer Science, Vol. 11(5) (1982), pp. 341–356.

[28] Pawlak, Z., *Rough Sets: Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Dordrecht, 1991.

[29] Read, S., *Thinking about Logic: An Introduction to the Philosophy of Logic*, Oxford University Press, Oxford, 1995.

[30] Schneider, F.B.," Morrisett, G.," Harper, R., *A Language-Based Approach to Security.* In: Informatics: 10 Years Back, 10 Years Ahead. Lecture Notes in Computer Science, Vol. 2000. Springer-Verlag, 2001. pp. 86-101.

[31] Smyth, M.B., *Powerdomains and predicate transformers: a topological view*, In J. Diaz, editor, Automata, Languages and Programming, pages 662-675, Berlin, 1983. Springer-Verlag. Lecture Notes in Computer Science Vol. 154.

[32] Smyth, M.B., *Topology*, In Abramsky, S.," Gabbay, D.M.," Maibaum, T.S.E. (eds.): Handbook of Logic in Computer Science. Volume I. Background: Mathematical Stuctures, pp. 641-761, Clarendon Press, Oxford, 1992.

[33] Szpilrajn, E., *Sur l'extension de l'ordre partiel*, Fund. Math. 16 (1930), pp. 386-389.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Social Engineering-Based Attacks: Model and New Zealand Perspective

Lech J. Janczewski
The University of Auckland, New Zealand
Email: lech@auckland.ac.nz

Lingyan (René) Fu
The University of Auckland, New Zealand
Email: reneweiss14@gmail.com

Abstract— **The objective of this research was to present and demonstrate the major aspects and underlying constructs of social engineering. An in-depth literature review was carried out resulting in the construction of a conceptual model of social engineering attacks. A case study was undertaken to understand the phenomenon with New Zealand-based IT practitioners to contribute insightful opinions. On this basis an improved model of social engineering-based attacks was formulated.**

I. Introduction

IN THE context of information security, social engineering describes a method of launching attacks against information and information systems. Both organizations and individuals have suffered enormous loss from these attacks. However, social engineering as a security threat is constantly overlooked because awareness of this phenomenon is currently low and there is a lack of a conceptual model to represent this form of attack.

The objective of this research was to present and demonstrate the major aspects and underlying constructs of social engineering and identify the relations between them. An in--depth literature review was carried out resulting in the construction of a conceptual model of the social engineering attacks.

A case study was undertaken to understand the phenomenon and a total of twenty-five New Zealand-based IT practitioners participated in this research to contribute insightful opinions.

Analysis of the collected opinions allowed us to present an improved model of social engineering-based attacks.

The paper starts with the formulation of the research objective and hypotheses followed by a presentation of a literature-based model of these attacks. Analysis of the New Zealand-based researchers' opinions is then presented. The paper terminates at the formulation of an improved model of social engineering based attacks.

II. Research objective, working hypotheses and methodology

The overall research objective of this study was defined as:

Exploring the significant entities and relations within social engineering-based attacks

To properly address this research objective, we formulated the following specific research questions:

RQ1*:What are existing security vulnerabilities which can be exploited by social engineering based-attacks?*

RQ2*:What are the methods of social engineering-based attacks?*

RQ3*:What are the consequences of a successful social engineering attack?*

RQ4*:What can be done to mitigate Social Engineering-based attacks?*

RQ5*:What is New Zealand's perspective of social engineering-based attacks?*

An exploratory case study approach was chosen to conduct the research investigation.

III. Proposed Model

Based on analysis and discussion of the conducted literature review, several key entities of social engineering attacks have been identified: People, Security Awareness, Psychological Weaknesses, Technology, Defenses, and Attack Methods.

As shown in Table 1, hacking and social engineering are the most common attack methods adopted across most fields. Traditional hacking and malicious code attacks are technical-based, targeting system or application vulnerabilities. However, the effectiveness of technical-based attacks has decreased as technological security solutions are gradually being adopted by more and more people and organizations [1]. Hence, this has encouraged technical hackers to choose the alternative – social engineering, targeting the existing vulnerabilities of both people and technology [2]. As a result, it is considered as the biggest security threat faced by both organizations and individuals today.

Social engineering is a diverse and complex technique for gaining unauthorized access to confidential proprietary and personal information [3]. It is primarily known as non-technical (human-based) attack, including impersonation, dumpster diving, shoulder surfing, and reverse social engineering; however, it could also be technology-based, including pop up applications, email attachment, online scams, and vishing. The literature review also indicates that the trend of technology-based attack is increasing with the emerging Internet technologies, such as online services and social networks [4], [5], [6] and [7].

IV. Consequences

The impact of social engineering attacks can be significant to both individuals and organizations [3]. The phase of the attack decides the type of information that may be valuable to social engineers. For instance, at the information gathering stage, information such as miscellaneous trivia within an organization; internal documents such as phone directories and organizational charts can be very useful. From an individual perspective, it will be personal information, such as date of birth, address and so on. Loss of CIA is the lesser damage and seen as the

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Table I.
Summary of Security Threats and Attack Methods

Threat Initiator	Motivation	Attack Action	Method
Hacktivism (hacker, cracker, Phreaker, script kiddies)	• Ego, fame • Destruction of system or information	• DoS / DDoS • System intrusion • Website defacement • DNS attack	• Hacking • **Social engineering**
Computer Criminal	• Theft of data • Monetary gain • Privacy compromise • Illegal information disclosure • Unauthorized data alteration	• Cyber stalking • Fraudulent act (man-in-the-middle, spoofing)	• Hacking • **Social engineering**
Insider (disgruntled, poorly trained, dishonest, malicious, negligent, former employee)	• Revenge • Monetary gain • Intelligence	• Unauthorized access • Computer abuse • Fraud • Theft of proprietary information (both logical and physical) • Bribery • System sabotage	• Malicious code (logic bomb, Trojan, malcode inserted as part of the software development process) • **Social engineering**
Industrial Espionage	• Competitive advantage	• Unauthorized access • Information theft • System Penetration	• Hacking • **Social engineering**

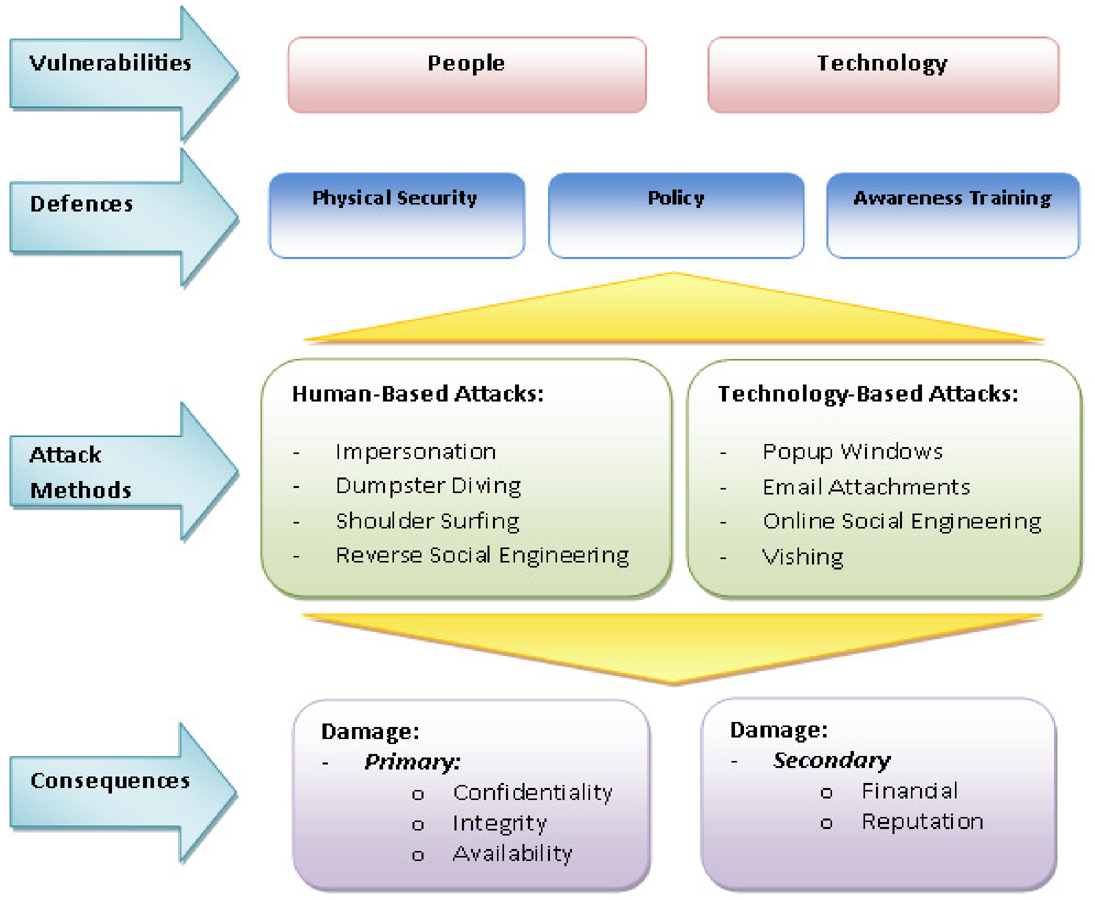


Fig. 1 The Conceptual Model of the Major Aspects of Social Engineering-Based Attacks

TABLE 2
SIX ENTITIES USED FOR DATA ANALYSIS

Entities	Description
People	Vulnerabilities in people that can be exploited by social engineering-based attacks.
Technology	Vulnerabilities in technology that can be exploited by social engineering-based attacks.
Security Strategy	Vulnerabilities in security strategy that can be exploited by social engineering-based attacks.
Defenses	Possible defenses to mitigate the risk of social engineering-based attacks.
Attack Methods	Types of the common social engineering-based attack methods.
Consequences	The damage of social engineering-based attacks to both organizations and individuals.

primary result of a successful social engineering attack. The information gathered is later used in the subsequent stage - attack phase. Again depending of the purpose of the attack, the level of damage varies. This is considered as the secondary damage of successful attacks.

The secondary damage is twofold. From a commercial perspective, firstly, social engineering attacks may cause significant financial losses, for instance, loss of business secrets to the competitor as a result of an insider attack, or loss of important equipments as a result of physical attack. Secondly, public disclosure of sensitive customer information stolen from a financial institution will damage the organization's image, followed by subsequent financial loss. From an individual's perspective, social engineering is a threat to people's personal life and finance, for instance, bogus advertisement and phishing attacks are the typical examples. The direct consequence to these attacks is identity theft which will finally lead to financial damage.

The above major entities form the basis of the social engineering-based attacks model presented in Figure 1.

V. Data Analysis & Finding

To verify the proposed model a total of 25 interviews were conducted, and the participants (from 17 organizations) were of various IT related backgrounds and experiences. The organizations the participants work for are across industries, including financial institution, security advisory services, government department, IT advisory, IT outsourcing, multi discipline computer firms, multi discipline consulting firms, and education. Among these organizations, there were seven local organizations and ten international organizations. The participants' job functions can be largely divided into seven categories: IT advisor, IT architect specialist, IT consultant, security specialist, IT solution designer, database management, and IT educator.

The entities used for the interview were derived from the conceptual model proposed in Fig.1 and summarized in Table 2. Major findings are presented next while the detailed answers are in [8].

VI. The Refined Model

Based on previous discussion and analysis of the information gathered during the interview phase, the revised model was developed shown in Fig 2. In the "Vulnerabilities" section, Security Strategy is added as a new entity which emerged from the analysis of the twenty-five conducted interviews. Security Strategy is in place to eliminate existing risks; however, the result from the interview analysis suggests that immature security strategy can be exploited indirectly by social engineering-based attacks. Therefore, Security Strategy is also considered as a vulnerable entity in information security. In the "Defenses" section, Technical Controls, Security-Enhanced Products, and Education entities were added.

These are the additional defence approaches which were believed to improve the mitigation of social engineering risks. In the "Attack Methods" section, Questionnaires are added as a new information gathering method.

Answering the research questions presented at the beginning of the paper:
RQ1: *What are existing security vulnerabilities which can be exploited by social engineering based-attacks?*

This research question was answered by findings from both the literature review and data analysis. The literature review identified two major components in information security that can be directly exploited by social engineering, including people and technology. In agreement with the literature review, the interview analysis suggested that security process should also be considered as a component which is vulnerable to such attacks.

Both the literature review and data analysis have suggested that people are the weakest link in security control systems. In addition, lack of security awareness and psychological weaknesses of people are the main reasons that social engineering-based attacks succeed. According to the literature review, people lack understanding of the current security issues - especially social engineering. This is supported by **64%** of the interview participants. Secondly, in agreement with the identified psychological weaknesses from the literature review, **40%** of the participants commented that appearance is a significant factor that can influence people's perceived trustworthiness.

The findings have suggested that there are three major issues associated with technology. First, the literature review has revealed that a lot of technology products have flaws in the security design. These flaws are used by an attacker to generate technology based scams which preys on people's fear of security. Data analysis shows that **16%** of the participants agreed to this. Secondly, in line with the literature review, **12%** of the participants pointed out that most security technologies are incapable of detecting and preventing social engineering as social engineering bypasses technical controls via manipulating people who are managing them. Additionally, **12%** of the participants stated that there is an increasing trend of malicious misuse of advancing technology products, such as Google Applications. It was also noted that online social networking services are often used as information gathering tools by the social engineers.

From the data analysis, security strategy emerged to be a significant entity that is vulnerable to social engineering-based attacks. The literature review has suggested that social engineering depends on uncertainty, for example, people are unsure about the requestor's identify, or people are

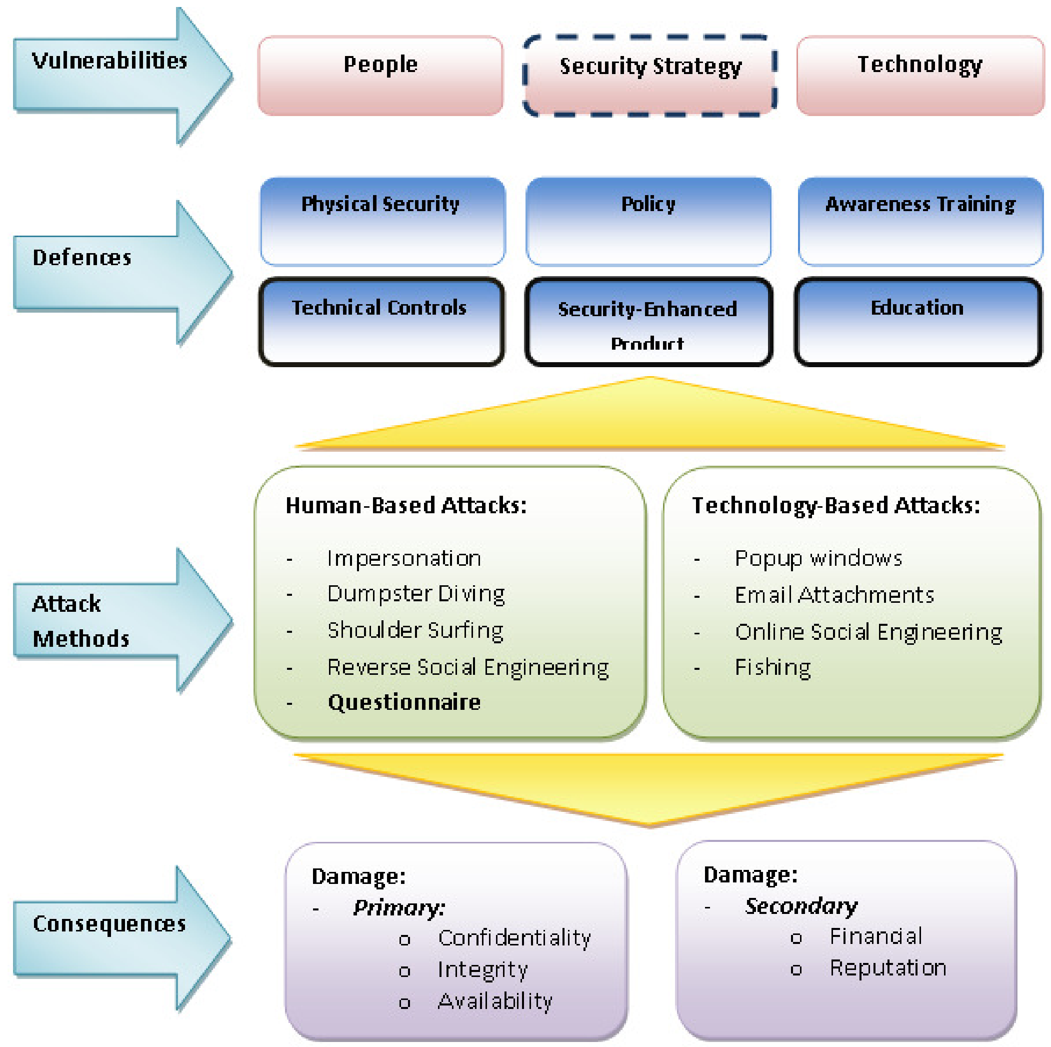


Fig 2. The Revised Conceptual Model of the Major Aspects of Social Engineering-Based Attacks

unsure about if it is the right thing to grant access to the requestor, etc. This is the main reason for putting in place a well-thought-out security strategy. However, **40%** of the interviewed participants emphasized that in general security strategies of organizations in New Zealand had poor security strategies because they overlooked people issues. Four types of threats from ineffective security strategies resulted from the data analysis - human behavior, system types of threats from ineffective security strategies resulted from the data analysis - human behavior, system implementation, physical environment, and governance issues. As the interview participants pointed out, a prospective social engineer may launch attacks against any of these vulnerabilities and conduct successful attacks.

RQ2: *What are the methods of social engineering-based attacks?*

Social engineering is often referred to as the people side of hacking which relies on influencing, deceiving, and psychological manipulation with or without the use of technology. It is diverse and complex and the form of the attack varies based on the attack motivation. Today, social engineering-based attacks can be basically classified into two categories, including human-based and technology-based.

Human-based social engineering is purely based on deception and can be conducted either in person or over the telephone. The methods include impersonation, dumpster diving, shoulder surfing, reverse social engineering, and questionnaire. Also, compared to an in person attack, by telephone is the most widespread mode as voice can be disguised. This mode is also easier for the social engineers to cover their tracks.

Technology-based social engineering is closer to traditional hacking technique. The purpose is to trick users into believing that they are interacting with authentic computer systems through the use of software or application. The attack methods include popup windows, email attachments, online social engineering, fishing, and rogue security software.

RQ3: *What are the damages after the attacks have been launched successfully?*

TABLE 6
ANALYSIS OF PARTICIPANTS' SOCIAL ENGINEERING-BASED ATTACK EXAMPLES

Attack Method	**Case #**	**Description**	**Damage**
Dumpster Diving	**Case 1**	The attacker went through waste to discover useful information such as credit card number simply because the merchant didn't pay attention to their waste management.	The victim lost their confidential information and money.
In Person	**Case2**	The pen-tester pretended to be someone who worked on the management floor and made the cleaner believe him. The cleaner had full access to the building, yet, their security awareness is very low. They are not trained to respond to unusual request.	It is only a penetration testing for the company, therefore nothing was lost.
	Case3	The attacker impersonated a cleaner to plant the key logger to steal sensitive information.	There was no significant damage because that bank has adopted appropriate security controls.
	Case4	People got access to the building and stole company property during office hours simply because the office staffs don't challenge strangers and abnormal behaviours.	Company equipment got stolen; this is considered as a financial loss.
	Case5	Similar to the previous pen-test case, the pen-tester was able to bypass security controls and gain access to company resources because the internal staffs, such as the reception was convinced that the pen-tester was part of the company.	It was a penetration testing; therefore there was no real damage.
Online Social Engineering	**Case6**	The attacker built trust with the victims and sent them photos with an embedded Trojan. The victims accepted the photos because their level of security awareness is very low.	The victim lost their confidential information and money.
	Case7	The attacker set up a fake website with full merchant facilities and security certificate to sell bogus Olympic tickets. It was never a business to begin with, yet, the bank and VeriSign didn't validate at all.	A large number of people paid for the ticket that would never get delivered, over 50 million dollars was scammed.
	Case8	The attacker took advantage of the online dating services and tricked a number of female victims. He first chatted with them through email and gradually built trust with them. He then sent them photos with an embedded key logger to steal their bank account details.	Victims lost money.
Telephone	**Case9**	The attacker tricked people over the phone into revealing biometric authentication (voice) pass phrase to use it later at the authentication point.	People's pass phrase got stolen and used for later attacks.
	Case10	The pen-tester pretended to be a customer who knows about the services. He was able to get sensitive information out of the helpdesk by bypassing the normal identify checks. On the other hand, it also showed that there was a lack of training and security awareness in that company.	If it was a real attack, the company may end up losing business confidentiality.
	Case11	A practical test to see how the bank handled abnormal situations. The initiator pretended to be helpless and prey on the customer services' sympathy.	The tester successfully changed the other party's passwords twice. In a real attack, the likely result is money loss.
	Case12	The attacker called the helpdesk and impersonated one of the contractors trying to gain access to the company network. Fortunately, the helpdesk found out and successfully prevented the potential attack.	Nothing happened because the helpdesk was doing their job properly.

The damages of a successful social engineering attack are twofold, primary and secondary. Similar to traditional hacking, social engineers spend most of their time in preparation via information gathering. To do so, the social engineer needs to gain authorized access to resources which leads to a breach of CIA. This is considered as the primary damage if the attack is successful. The information gathered is then used in the subsequent attack phase - the secondary damage.

Secondary damage can be largely divided into two parts, reputation damage and financial damage. The typical social engineering attack targets organizations with sensitive information, such as customer database. Loss and improper disclosure of information will seriously damage the organizations' reputation as well as finances. Moreover, finances may be rebuilt, but people's confidence and positive reputation can take years to rebuild.

RQ4: *What can be done to mitigate Social Engineering-based attacks?*

Social engineering represents a wide range of threats. To effectively defend against social engineering-based attacks, it is necessary to have a multifaceted approach. Firstly, on the very basic level, physical security must be properly implemented because a lot of the attacks are through gaining physical access. Therefore, access control must be in place to allow the authenticated people to have access while keeping the rest out. Ideally, organizations should apply different access control mechanisms based on the security classification. Secondly, proper technical controls can help to reduce social engineering risks, such as multifactor authentication can effectively prevent unauthorized access, especially for online financial identify theft. Thirdly, security policy is the key and most important element of a good defence against social engineering because it can take out uncertainty which is what social engineering depends on. The policy should include all the business components which are necessary to protect. Ultimately, security policy should be supplemented by education and training. People's vulnerability to social engineering can be described in terms of their awareness to these types of attacks. Therefore, people must be educated and trained constantly to be social engineering resistant.

RQ5: *What is New Zealand's perspective of social engineering-based attacks?*

In general, New Zealand shares a similar trend with other countries in terms of technology adoption and security risks. However, with regards to awareness of security issues and the implementation of countermeasures, New Zealand is a behind others.

First, according to **64%** of the interviewed participants, the majority of businesses and people in New Zealand do not have sufficient understanding of existing security issues. **32%** of them particularly mentioned the social engineering phenomenon is overlooked. The findings show that apart from people's careers, environmental factors (economical scale, legislation, and breach disclosure) also affect people's security awareness. In particular, **28%** of the interview participants mentioned it is mainly because there have not been major security disasters in New Zealand. **8%** of the interviewed participants pointed out that the other reason is the large number of security breaches, especially internal breaches in New Zealand that are underreported due to insufficient law enforcement. In addition, **44%** of the interviewed participants commented that people in New Zealand generally have a higher level of social trust which implies that they are more vulnerable to social engineering-based attacks.

Secondly, with regards to security strategy, **40%** of the interview participants commented that the organizations in New Zealand in general do not have a well-defined security strategy because they overlook people issues. **16%** of them pointed out being a small country; the majority of businesses in New Zealand have less than twenty people which have difficulties implementing segregation of duties. **20%** of the interviewed participants mentioned that the lack of legislations, standards, and compliances in New Zealand also have impact on immature strategies within the organizations. Furthermore, **40%** of the participants commented that immature strategies expose four categories of vulnerabilities which can be exploited by prospective social engineers.

Thirdly, the social engineering-based attack examples given by the participants show the diversity and complexity of this form of attack. As they stressed, there is a need for multifaceted defence approach to mitigate the risks associated with social engineering. In agreement with the approaches identified in the literature, the participants suggested that education, appropriate technical controls, and security-enhanced products should also be adopted.

VII. Research Contribution, Limitations and Future Research

This study makes three contributions:

Firstly, it provides insights regarding the social engineering phenomenon from both theoretical and practical perspective and developing a conceptual model to demonstrate the findings.

Secondly, it enhances understanding of social engineering in terms of the impact of this form of attack to businesses and individuals, through illustration of real New Zealand examples from the research participants.

Thirdly, it provides a multifaceted defence approach towards mitigating the risks associated with social engineering.

The research limitations are twofold. The first limitation is the selection of the interview participants. The participants of this research were the IT practitioners in New Zealand who either had a security background or had an interest in security issues. As a result, there is a lack of understanding from the non-IT perspective as people's perception of the problem might be different. This can be justified as the purpose of this research is to collect insightful opinions about major issues regarding social engineering-based attacks in New Zealand. Therefore, it was not the researchers' interest to investigate the research topic from a non-IT perspective. Secondly, this study constructed a conceptual model for social engineering-based attacks based on the findings from the literature review and interview analysis with twenty-five New Zealand-based IT experts. However, the model and the underlying theory require further validations.

Opportunities for future research on the basis of this research are in three areas. First, as mentioned in the previous section, one area would be to validate the constructed concept of social engineering-based attacks by assessing the underlying theory of each identified entity. In addition to this, a more sophisticated model can be built using this conceptual model. Secondly, this research has provided detailed analysis of how to conduct social engineering attacks. The analysis can be used as the basic guideline to design an educational software application that educates people on how to act in different attack situations. Lastly, as part of this research, a multi-faceted countermeasure against social engineering-based attacks was illustrated. Each of the proposed defense approach can be seen as a future research direction.

References

[1] Twitchell, D. P. "Social engineering in information assurance curricula," in: *Proceedings of the 3rd annual conference on Information security curriculum development*, ACM, Kennesaw, Georgia, 2006.

[2] Granger, S. "Social Engineering Reloaded," 2006.

[3] Hinson, G. "Social Engineering Techniques, Risks, and Controls," *EDPACS: The EDP Audit, Control & Security Newsletter* (37:4/5) 2008, pp 32-46.

[4] DeWalt, D. "Cybercrime: The Next Wave."

[5] Keize, G. "Old worm makes comeback," Computerworld, 2009.

[6] McAfee "McAfee Advert Labs - Top 10 Threat Prediction for 2008," McAfee.

[7] McMillan, R. "Making a PBX 'botnet' out of Skype or Google Voice?," IDG News Services, 2009.

[8] Fu, R. Social Engineering-Based Attacks - Model and New Zealand Perspective, Master Thesis, The University of Auckland, 2009.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Global Mobile Applications for Monitoring Health

Tapsie Giridher
FalconStor Software, Melville,
New York, United States
tapsie.giridher@falconstor.com

Anita Wasilewska,
Jennifer L. Wong
Computer Science Department,
Stony Brook University
{anita, jwong}@cs.sunysb.edu}

Karan Singh Rekhi
Amazon Web Services
rekhi@amazon.com

***Abstract*—The incentive of the mobile applications presented in this paper is the extensive spread of the mobile phone culture during the past decade. The first application is CalorieMeter, a calorie intake monitoring application. Cheer Up, the second application is based on self-help scientific methodologies for diagnosing possibilities of different kinds of depressions. Designs of both applications are based on the ideology of Mobile Phone Template applications. It supports easy transformation of a given application into other application domains and allows us gain natural language and regional languages independence, hence the global nature of our approach. Our applications have been developed for and tested on low- medium range of mobile phones.**

I. Introduction

MOBILE phone applications developed for the improvement of health of individuals capitalize the ubiquitous nature of mobile phones. Our first application, CalorieMeter concentrates on the improvement of the physical well-beingof an individual. The goal of this application is to motivate an individual to monitor his food and calorie intake so that he is able to meet his calorie needs without exceeding his permissible calorie amount for a particular day. The number of calories available for consumption per day is calculated using the Harris-Benedict equation [1]. We have created an intuitive and graphical interface to persuade users to use CalorieMeter regularly. This application as it is presented here targets English-literate users but its food choices can be conveniently adapted to any language and culture. The English texts in the language dependent screens (for example, user's profile) can be readily translated to any language we want to adapt the application to. The English texts can then be easily replaced by translations. All this is possible because the application design follows Template Ideology as defined in [2]. A template application is an application which can easily be modified from one domain to another.

The second application, Cheer Up is designed to help an individual to check the state of his mental health. Its goal is to help the user to become aware of his depressed state, if he is in one and to get a confirmation of his good mental status otherwise. We have created an appealing and enticing interface so the user can understand the flow of the application without any outside explanations. We believe that the intimacy and privacy of mobile phones will make people use the application and consequently help them overcome the stigma attached to depression. Eventually, they would seek a professional help, when needed. The application mainly targets teenagers who are regular cell phone users, might be depressed but otherwise would not seek help. We also show that application design follows Template Ideology.

A. Paper Organization

In section II, we present the physical health aid application, CalorieMeter. In section III, we present the mental health application, Cheer Up. For each application, we have described the application functionality, technical details, challenges and lessons learnt. In Section IV we show the working of the applications followed by the Template Ideology [2]. Section V is a conclusion of the paper.

II. CalorieMeter

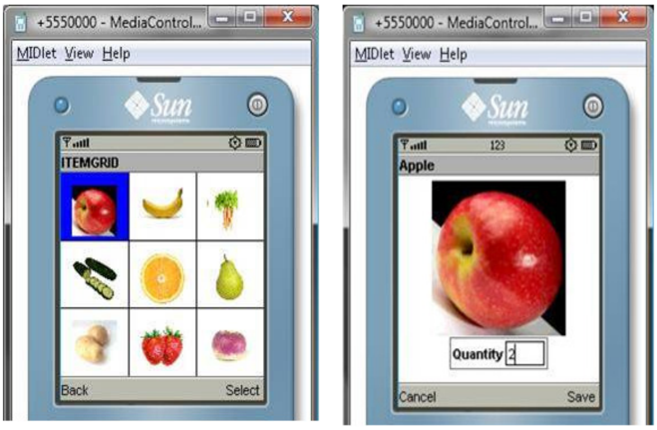
CalorieMeter helps the user monitor his food and calorie intake. In this application, the user sets his profile, enters food consumed, and the application monitors and reports his caloric intake and balance. The application also suggests food items the user could consume while still remaining within the permitted calorie intake.

Four major factors motivated us to develop CalorieMeter. The first factor is the rising obesity throughout the world. WHO [5] projects that by 2015, approximately 2.3 billion adults will be overweight and more than 700 million will be obese. The second factor is the lack of awareness of the calorific value of food items amongst individuals. Another factor is the lack of awareness of the calorie needs of an individual and need of consumption of a balanced diet.

A. Application Functionality

On using the application for the first time, the user is prompted to set a profile. The *Profile* screen consists of a form like interface with fields for name, sex, birth date, height and weight. On saving the profile, the *Home* screen appears. The *Home* screen has four options: *Profile, Food Categories, Calorie Balance* and *What To Eat???*. All the options navigate to their respective screens.

On the *Food Categories* screen, the user selects the food category and a food item grid of the corresponding food category appears. The three food categories are *Fruits and Vegetables*, *Meats and Grains* and *Deserts and Soda*. Each category is represented as a 3X3 grid of food items. The background color of the selected food item in the grid changes dynamically according to the time of the day along with the audio pronunciation (Fig. 1(a)). On selecting the food item the user is navigated to the *Food Item Details*



Quantity 2

Fig 1. (a) Food Items Grid , (b) Food Item Details

screen (Fig. 1(b)). Quantity can be entered on the *Food Item Details* screen [for grains, 1 implies 100 grams]. On choosing *Save*, the calories for the food item are added to the already consumed calories of that particular day.

The *Calorie Balance* screen helps the user approximate his share of daily permitted calories consumed (as calculated for user profile by our algorithm) and left to be consumed through a pie chart. The red color fraction of the pie signifies calories consumed and the green portion of the pie signifies the calories available for the day (Fig. 2(a)).

The What To Eat??? button on the *Home* screen navigates the user to the *What To Eat???* Screen (Fig. 2(b)). This screen suggests intelligent and healthy food item suggestions to the user. The suggestions are displayed in the form of a grid. Any food item consumed from this grid assures that the user does not exceed the permitted calorie intake for that day.

The application also has additional alert messages displayed on the screen on occasions when the user does not eat or exceeds the calorie limit.

B. *Key Features*

CalorieMeter has attractive grid navigation with integrated audio support. The user can use the up, down, left and right buttons to navigate and select food items from the cells of the 3X3 grid. The name of the selected food item is pronounced in English, but that can be changed due to the template design to any other language. The application can also serve as language teaching tool in a case when user language is different than the language used in the application. Another feature is the dynamic resizing of images in the grid. Hence we store only one copy of the image. The application also implements an intelligent algorithm for suggestions of healthy food items that can be consumed to remain within the calorie limit.

C. *Technical Details*

CalorieMeter incorporates 27 .png images with 120X120 size and 27 .wav files in the application and supports low end mobile phones (CLDC, MIDP 1.0). It has been developed using J2SDK, Eclipse IDE, J2ME Wireless Toolkit and Eclipse ME, Eclipse plug-in to develop J2ME code.

We reuse our custom grid implementation code which is populated dynamically based on the food category chosen. Our custom method dynamically re-sizes the images to fit the grid irrespective of the mobile phone screen size. The widths and heights of the images initially 120X120 are divided by an integer factor which is incremented on iteration till the re-sized image fits perfectly in the grid. We also implemented a string tokenizer to parse the .txt files containing data about the food items. We used RecordStores to emulate RDBMS by creating separate RecordStores to store information related to User Profile, Food Categories, Food Items and Food Consumption. Since J2ME does not have a built in date type, we accept the birth date of the user and store it as month, date and year.

What To Eat??? screen considers calories consumed for the day and permissible calorific value for the day and outputs a grid of food items which can be consumed without exceeding the permissible calorie limit. The weight-age assigned to each category is the motivation factor as the output has more healthy options than unhealthy ones.

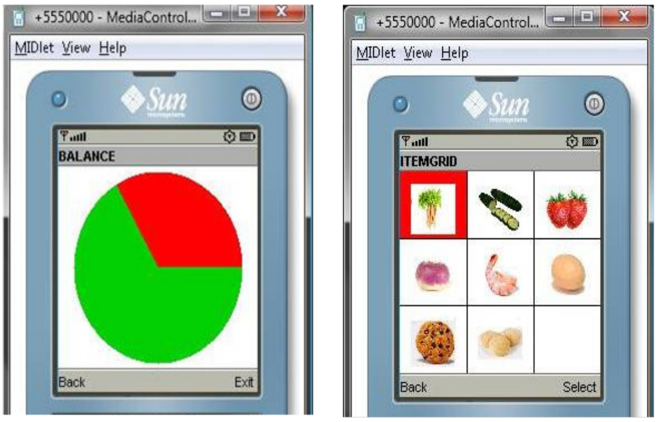


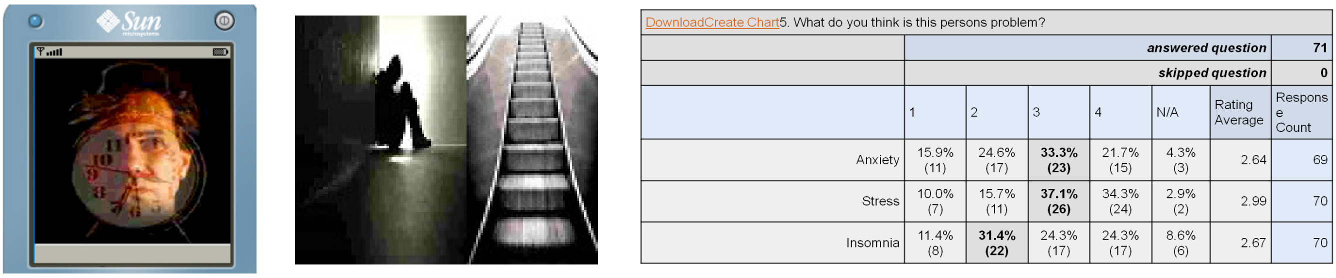
Fig 2. (a) Calorie Balance screen (b) What To Eat??? Screen

D. *Challenges and Lessons Learned*

The success of the application depended on the image size and clarity for easy comprehension by the user. Also, inclusion of packaged food items, platters and combos as food choices could increase the versatility of the application. This can be easily added during the customization. CalorieMeter can be also customized by adding/replacing picture set of food items according to the age groups and the regions, counties of users. Our application does not consider the frequencies of consumption of food items or the physical activities of the individual, but the template design allows and supports extensions.

E. *Related Work and Comparison*

This section describes a short comparison between prototypes of health applications deployed on technology platforms with consideration to type of architecture, motivational strategy employed and target audience. The applications we compare are PmEB [11], BALANCE [9], Heart Angel [10], FotoFit [7], and Wellness Diary [12]. PmEB [11] requires an internet enabled mobile phone whereas HeartAn-



DownloadCreate Chart5. What do you think is this persons problem?

						answered question	**71**
						skipped question	**0**
	1	2	3	4	N/A	Rating Average	Respons e Count
Anxiety	15.9% (11)	24.6% (17)	**33.3%** **(23)**	21.7% (15)	4.3% (3)	2.64	69
Stress	10.0% (7)	15.7% (11)	**37.1%** **(26)**	34.3% (24)	2.9% (2)	2.99	70
Insomnia	11.4% (8)	**31.4%** **(22)**	24.3% (17)	24.3% (17)	8.6% (6)	2.67	70

Fig 3. (a) Sample Survey Picture (b) Sample Diagnose Image (c) Survey Results

gel [10] requires a Bluetooth enabled cell phone. All of them are standalone applications whereas PmEB [11] has client-server architecture. All the applications except PmEB [11] and Wellness Diary [12] require an external hardware device. For example Heart Angel [10] requires a heart rate monitoring hardware device. The motivational strategy in PmEB [11] and Wellness Diary [12] is control, in BALANCE [9] is challenge, in FotoFit [7] is competition and in HeartAngel [10] is encouragement. BALANCE [9], Heart Angel [10], Wellness Diary [12] target fitness enthusiasts, PmEB [11] targets obese mobile phone users and FotoFit [7] was developed for college students.

III. Cheer Up

Cheer Up is designed to evaluate the mental health of individuals by discovering if they are in a depressed state, categorize this state and send a warning to the user. It is an attempt to make Cognitive Behavioral Therapy (referred as CBT hereafter) [14] more accessible and personalized. We hope it would appeal to young people and make them aware that they might need to seek professional help.

A. *Motivation*

Depression is the single biggest cause of suicides in the world [22]. Most cases of depression go unnoticed, unreported and are aggravated due to lack of self-awareness, easily accessible psychiatric help or stigma associated with visiting a psychiatrist. The portable and personalized nature of mobile phones along with its ubiquitous nature might help to breach the gap between people who need professional help and those who eventually reach for it [15].

B. *Requirements Elicitation*

The foundation of Beck's cognitive theory of depression is a stress-diathesis model [13]. Persons may be vulnerable to depression because they have dysfunctional beliefs. These beliefs may remain latent for years, prior to and between depression pangs, but they can become primed by environmental stressors. The Cognitive Behavioral Therapy [13] aims at knowing how we think about ourselves and our surroundings and our emotional reactions to events occurring around us. By changing our thought process, we can change our emotional reactions which in turn can prove helpful in treating depression.

To feed our diagnosis algorithm with quantifiable associations between images & depression types, we conducted a survey of around 100 people in the age group of 20 – 30 (Fig. 3(a)). They were asked to rate the images on the scale of 1 to 5 where 5 indicated maximum association of the image with the type of depression. The entered weights were aggregated and the mean values were used to derive the probability of associations.

Based on the survey results we tabulate the mapping of each image to various types of depressions. This is stored in the configuration file and is used in the diagnosis algorithm.

C. *Application Functionality*

The *Home* screen consists of three options. The first option is *Diagnose* which is the key part of the application where user takes a self-diagnosis by responding, with numbers, to an adaptive display of images (Fig. 3(b)). The users' responses are used to predict their cause and degree of depression. These values are stored as user's profile in the record store until the user takes a new diagnosis. The user's predicted depression type and level is also used to assign a personalized set of therapy tasks like art therapy, cope, mood logs etc. The second option *To-Do* suggests ways to figure the type of depression discovered.

The *Help* button displays instructions on how to use the application. The *Diagnose* button starts a slide-show of images which evaluate the type of depression in the user. For each image, the user selects a number from 0-9 where a higher number indicates that the user can relate to the image to a higher extent (Fig. 2(b-c)). Once the slide-show is over, the results are displayed. Once the user completes the diagnosis, the user's profile is stored in the phone. On clicking on the *To-Do* button, the application launches a therapy task customized for the user based on the depression diagnosis. Scientific research proves the efficacy of art therapy in healing depression [23]. The tasks are usually timed and self-evaluated. Depending on the task image these may try to divert the user's mind, make him busy in a soothing hobby, or try to make him learn to cope with him causes of depression.

D. *Technical Details*

Extensibility was one of the primary goals in our design as we wanted to develop our application as a frame work for a generic family of applications that plays an interactive and adaptive sequence of images to gather up user responses and analyze them. This was made possible through an efficient object oriented design. The Diagnosis Algorithm multiplied the score entered by the user by the weights of various depression types. These weights were calculated as a result of the survey conducted by taking the weighted mean of the percentage of user responses. The sum of all results calculated to get the final result was then mapped to the various types of depressions.

We believe that the algorithm will be more accurate if the data obtained from the survey was actually obtained from real controlled clinical trials of patients suffering from depression. Our application is a proof of concept of a diagnostic tool. Our application can serve as a template for a real application. The diagnostic mobile based tool should be developed in collaboration with medical specialists and we plan to do so.

E. Related Work

There have been some innovative attempts at diagnosing and healing depression using computing technologies. They can be categorized into three major forms Internet Based CBT [19], Computer Based CBT [20], Mobile Based [16], [17], [18], and [21].

Mobile based depression therapy is a very new area and there are only few applications we came across. A disadvantage of these applications is that they require high end mobile phones and rely heavily on the use of the Internet.

In [16] the mobile phone acts as a patient terminal, mobile phone providers connect the patient to the Internet where the IIS web server stores the information. The patient receives an email with a link to interview page. The user selects a number which gives information about the degree of particular symptom.

In [17] the data is stored and can be accessed online through a mobile phone. This existing system contains a real-time advice function. Real time advice function implements an algorithm that is performed over the user responses to analyze the kind of depression he is in and provides encouraging or warning messages in order to boost his morale and take actions to perform better.

In [18] a mobile phone program monitors adolescents' mood, stress and coping behaviors with some specific aims. [21] presents a mobile application developed in Murdoch Children's Research Institute. This application is used to track young users' experiences of moods and stress levels. The authors have developed a real-time, youth-friendly mobile phone program based on momentary sampling (MS) with interactive monitoring programs that can be run on java-enabled MIDP 2.0 mobile phones. MS data may be recorded by calling participants on mobile phones and using automated interactive voice response systems or researcher-lead interviews.

IV. Template Ideology

CalorieMeter can be transformed into an application of another domain with minimal knowledge of programming in J2ME. Three types of changes are possible in the transformation. Firstly the user can change the set of food items displayed in our application. Secondly the weight-age of each category can be changed. Thirdly the categories can be changed. The only constraint is that the number of categories should remain three. However the user would have to clean and build the code and redeploy it as the .jar and .jad file would get modified.

A.CalorieMeter

For example the CalorieMeter might be transformed for use of university students in the USA. In this circumstance, the picture set should consist of fruit and vegetable salads, pizzas, burgers, soda available in the cafeteria. The steps to transform the application are as explained below in the following sentences. To change the set of images, the .png files in the pictures folder of the application source code should be replaced by the picture files of new food items. The size of image should be 120X120. To change the set of audio files, the .wav files should be replaced by the audio files of new food items. The average size of our .wav files was 90kb. Standard notations have been followed in the text files which have the information about the object items to be present in the grid. The following should be followed as standard in the text files. '#' is used as separator between each field on the same line. '?' is used as the end of line at the end of each description. Each text file is one single line. '?$' is used as end of file. For example Categorydefaultentries.txt is in the form of 1#Fruit#Fruit.png#50? Where category id is 1, category name is fruit, icon is fruit.png, weight is 50, weightage is 2, # is a separator between fields and ? is for end of line.

B.Cheer Up

To ensure consistent display of Cheer Up across different mobile phone devices, all images adapt to phone screen size. Data regarding depression types, diagnostic images, and therapy tasks can be added via the configuration files of the application. The application can used as a framework to design and develop applications which evaluate user responses and provide homework tasks in an adaptive manner. All text comes from text files which can be easily changed and translated.

V. Conclusion

In this paper we presented two mobile applications inspired by the globally pervasive nature of mobile phones. Our first application, CalorieMeter helped the user keep track of his calorie intake on a daily basis thereby helping him to maintain a check on the calorie intake in a healthy fashion. Our second application, Cheer Up is an initiative to help depressed individuals become aware of their condition and henceforth curb the stigma attached with seeking external medical help. Mobile phones prove to be an ideal platform for health applications as they are affordable, personalized and interactive. Template Ideology renders our applications independent of regional, language, age, educational and social barriers.

Acknowledgment

Work supported by NCIIA National Collegiate Inventors and Innovators Alliance Sustainable Vision Grant #4891-07, "Sustainable Technology-based Entrepreneurship for the Senegalese Market" and NCIIA National Collegiate Inventors and Innovators Alliance Sustainable Vision Grant #4891-08, "Building a Global Network to Support Sustain-

able Information and Communication Technologies Entrepreneurship in Senegal".

The proof of concept of the Cheer Up application was developed by Karan Singh Rekhi, Ketan Dixit and Nilesh Vijayvargiya, computer science graduate students of Stony Brook University, New York. The proof of concept of the CalorieMeter application was developed by Parag Naik and Aneesha Bulchandani, computer science graduate students of Stony Brook University, New York.

References

[1] Wikipedia. http://en.wikipedia.org/wiki/Harris-Benedict_equation

[2] A. Wasilewska, J. Wong, “Template Mobile Applications for Social and Educational Development” in Proceedings of the International Multiconference on Computer Science and Information Technology,

[3] International Symposium on Intelligent Mobile Technologies for Social Change, Mragowo, Poland, 2009, pp. 391–398.

[4] T. Giridher ET. al, “Mobile Applications for Informal Economies” in Proceedings of the International Multiconference on Computer Science and Information Technology,International Symposium on Intelligent Mobile Technologies for Social Change, Mragowo, Poland, 2009, pp. 345–352.

[5] “RecordStore,” http://java.sun.com/javame/reference/apis/jsr037/.

[6] WHO.http://www.who.int/mediacentre/factsheets/fs311/en/index.html

[7] Z. Frątczak, G. Muntean, and K. Collins, “Electronic Monitoring of Nutritional Components for a Healthy Diet” in Digital Convergence in a Knowledge Society: The 7th Information Technology and Telecommunication Conference IT&T 2007, 2007, pp.91-97.

[8] B. Brown, M. Chetty, A. Grimes, and E. Harmon, "Reflecting on health: a system for students to monitor diet and exercise" in CHI '06: CHI '06 extended abstracts on Human factors in computing systems, New York, NY, USA: ACM, 2006, pp. 1807-1812.

[9] E. Arsand, J. Tufano, J.Ralston, and P. Hjortdahl, “Designing mobile dietary management support technologies for people with diabetes” in Journal of Telemedicine and Telecare2008, Volume 14, Number 7, 2008, pp. 329-332.

[10] T. Denning, A. Andrew, R. Chaudhri, C. Hartung, J. Lester, G. Borriello, and G. Duncan, "Balance: towards a usable pervasive wellness application with accurate activity inference," in Hot Mobile '09: Proceedings of the 10th workshop on Mobile Computing Systems and Applications, New York, NY, USA: ACM, 2009, pp. 1-6.

[11] C. Wylie, and P. Coulton, “Persuasive Mobile Health Applications” in 1st International Conference on Electronic Healthcare for the 21st Century, 2008.

[12] C. Tsai, G. Lee, F. Raab, G. J. Norman, T. Sohn, W. G. Griswold, and K. Patrick, "PmEB: A mobile phone application for monitoring real time caloric balance" in Conference on Human Factors in Computing Systems, 2006,New York, NY, USA: ACM, 2006, pp. 1013-1018.

[13] A. Ahtinen, S. Ramiah, J. Blom, and M. Isomursu, “Design of mobile wellness applications: identifying cross-cultural factors,” in OZCHI ’08: Proceedings of the 20th Australasian Conference on Computer-Human Interaction, (New York, NY, USA), pp. 164–171, ACM, 2008.

[14] Beck, A. T. “Cognitive therapy: A 30-year retrospective” American Psychologist 46, 1991, pp. 368-375.

[15] “Computerized cognitive behavior therapy for depression and anxiety” Review of Technology Appraisal 51,National Institute of Health and Clinical Excellence, 1999.

[16] Hareva, D. H., Okada, H., Kitawaki, T. & Oka, H. “Supportive Intervention Using a Mobile Phone in Behavior Modification”.Acta Med Okayama 63, 2009, pp.113-120.

[17] Okada, H., et al. “Development of an EMA real-time data collection system using a mobile phone.” Journal of Psychosomatic Research 58, 2005, pp. 52-52.

[18] Sophie, C. R., et al. “A mobile phone program to track young people's experiences of mood, stress and coping”. Soc Psychiatry Epidemio, l44, 2009, pp. 501-507.

[19] Pek, V., Nyklíček, I., Smits, N., Cuijpers, P., Riper, H., Keyzer, J. “Internet based cognitive behavior therapy for subthreshhold depression”, 2007, pp. 123-131.

[20] National Institute of Health and Clinical Excellence. “Computerized cognitive behavior therapy for depression and anxiety “.Review of Technology Appraisal 51, 2006.

[21] Reid S. C., Kauer S. D., Dudgeon P., Sanci L. A, Shrier L. A, Patton G. C. “ A mobile phone program to track young people's experiences of mood, stress and coping. Development and testing of the mobile type program”.Centre for Adolescent Health, Murdoch Children’s Research Institute, Royal Children's Hospital, 2 Gatehouse St., Parkville, VIC, 3052, Australia, 2008.

[22] http://www.a1b2c3.com/suilodge/facovr1.htm.

[23] Mcaffrey Ch., Lynn E. “The effect of healing gardens and art therapy on older adults with mild to moderate depression.”Holist NursPract. Mar-Apr; 21(2), 2007, pp. 79-84.

ISBN 978-83-60810-27-9
ISSN 1896-7094

A Study on the Expectations and Actual Satisfaction about Mobile Handset before and after Purchase

JiBum Jung,
National IT Industry Promotion
Agency.567 Expo-ro,
Yuseong-Gu, Daejeon,
305-348, Korea.
Email: jung@nipa.kr

seungpyo Hong,
National IT Industry Promotion
Agency.567 Expo-ro,
Yuseong-Gu, Daejeon, 305-348

***Abstract*—This thesis is intended to examine factors that affect customer satisfaction in the domestic mobile communication terminal market as to expectations before purchase and actual satisfaction after purchase. Also, how the factors that affect customer satisfaction about mobile phones influence the customer base was theoretically and positively examined. The mobile communication terminal industry, which has been the driving force behind the development of Korea as a great power in the information and communication industry, has a great influence on the global market as well as the domestic economy. Nevertheless, research efforts on the existing mobile communication market havebeen focused on the mobile communication service market rather than the mobile communication terminal market. In addition, research on customer satisfaction about mobile communication terminals, which has been done by a few scholars in the related fields, has been limited to prices and brands. That is, the traditional research efforts on customer satisfaction about mobile communication terminal products have been focused on the influences of prices and brands on the purchase of products rather than on the evaluation of the unique quality attribute of each product.Therefore, this thesis is intended to examine factors expected to enhance customer satisfaction about products and expand the customer base other than the external factors including the prices andbrand images of mobile phones so that customer-oriented mobiles phones can be developed and manufactured.**

Keywords-component: Mobile Communication Industry, Customer Satisfaction, Mobile Communication Terminal

I. Introduction

KOREA succeeded in developing the world's first commercial CDMA system in the year 1995 and started to provide commercial services in the year 1996. Ever since then, the domestic mobile communication industry has marked an unprecedented growth on a consistent basis. As of November, 2006, the number of subscribers to the domestic mobile communication services is over 40 million, which comes to one mobile phone per person as to the economic activity ratio per population, a high supply rate of mobile communication services (the Ministry of Information and Communication, 2006). 5). The rapid supply of mobile phones, which have become a necessity in modern life, and the development of communication service technology have led Korea to become a great power in the world's communication industry where communication services are available in the most convenient manner. Services activated by the development of CDMA equipment have made the domestic mobile communication market rapidly grow and develop, having a huge influence on the global and domestic economies. The birth of Korea as a great power in information technology has resulted in an unprecedented growth within a short span of time and helped Korea take the lead in the export of information technology worth 100 billion dollars. Also, information technology, now a representative industry of Korea at the peak of the "Korea Brand Premium", is being admired by a number of nations in the world. Nevertheless, research efforts on the existing mobile communication market have been focused on the mobile communication service market rather than the mobile communication terminal market. As stated above, the mobile communication market is a combination of the service industry where services are provided through a network of mobile communication and the manufacture industry where systems and terminals are produced and supplied. Thus, research on the mobile communication market must be accompanied by the analysis of the mobile communication terminal market. Moreover, the quality evaluation of tangible terminals intended to use intangible services must be done besides the quality evaluation of mobile communication services. In general, the mobile phone industry is a production-led industry where the manufacturer's ideas stimulate the user's desires to purchase the supplied products. Also, new technology developed by mobile phone manufacturers has consistently educated consumers. Accordingly, customer satisfaction is formed on the basis of the nature of products suggested by manufacturers, and it is our tendency to understand the level of customer satisfaction as unmatched loyalty to the corresponding products. In addition, research on customer satisfaction about mobile communication terminals, which has been done by a few scholars in the related fields, has been limited to prices and brands. That is, the traditional research efforts on customer satisfaction about mobile communication terminal products have been focused on the influences of prices and brands on the purchase of products rather than on the evaluation of the unique quality attribute of each product. This research is designed to analyze factors that affect customer satisfaction in the mobile communication terminal market. Furthermore, the influences of expectations before purchase and actual satisfaction after purchase on customer loyalty will be analyzed so that factors expected to enhance customer satisfaction about products and improve customer loyalty as a result can be suggested and customer-oriented mobile phones can be manufactured with priority given to the unique quality attribute of each product.

II. Theoretical Backgrounds

A. *Customer Satisfaction*

(1) Quality Attributes of Mobile Communication Terminal Products

The previous studies evaluated the quality of tangible materials as having a long life and a number of functions. However, the concept of quality is now being reinterpreted with the focused placed on "meeting the needs of customers". Parasuraman, Zeithaml and Berry (1985, 1988) claim that quality perceived by customers is determined by the comparison between customers' expectations for the level of quality to be met by manufacturers and their actual perception of quality supplied by manufacturers. Such studies mainly use "Servqual"as a means to evaluate customer service quality. The 10 categories of corporeality, credibility, response, capability, courtesy, reliability, accessibility, communication, and customer understanding are expounded within the limit of corporeality, credibility, reactivity, assurance, and response. The quality evaluation of corporeal materials like mobile communication terminals mainly uses the attribute clue (Zeithaml, 1988), and the criteria for the attribute clue vary according to the nature of corporeal materials. SERVQUAL is customized in accordance with the criteria used by service providers, and the attributes like "polite and tidy-looking staff" and "accurate customer record maintenance" are very much different from the quality evaluation of corporeal materials.Thus, this research selected 8 attribute clues from 22 attribute clues of SERVQUAL closely associated with the unique attributes of mobile communication terminals through interviews with experts in mobile communication.

Table I. Quality Attributes of Mobile Communication Terminal Products

Corporeality	Shape and Size of Terminals / Design of Terminals
Performanc e	Various Functions of Terminals / Up-to-date Functions of Terminals
Convenience	Convenience of Character Input / Convenience of Subsidiary Functions of Terminals
Durability (Credibility)	Life of Terminals / Breakdown Frequency of Terminals

(2) Customer Satisfaction

Customer satisfaction wasperceived differently by a number of scholars in their previous studies, and it has been also evaluated in different manners. Research efforts on customer satisfaction can be roughly categorized into two perspectives. Yi (1990) divided customer satisfaction into resultant satisfaction obtained from consumer experience and interim satisfaction in the middle of the evaluation process. Oliver (1980) defined customer satisfaction as the function of expectations met and perceptions of discrepancies, suggesting a discrepancy paradigm. Westbrook & Reilly (1983) referred to customer satisfaction as a psychological response caused by experiences associated with the purchase of products or services. Churchill and Surprenant (1982) defined customer satisfaction as a conceptual, practical meaning and regarded the conceptual implication of customer satisfaction as the comparison between expected results of purchase, use, and consumption and price and compensation. In addition, Churchill and Surprenant regarded the practical implication of customer satisfaction as the sum of satisfaction about various attributes of products or services. In addition, they claimed that the quality of products and services increases when it exceeds the demands, needs, and expectations of customers in that customer satisfaction relies on the perceptions and expectations of customers. On the basis of such studies, this research is intended to analyze customer satisfaction about mobile communication terminal products as to expected satisfaction in the middle of evaluation before purchase and actual satisfaction coming from purchase experiences. The previously mentioned quality attributes of mobile communication terminal products will be measuredas a major factor that determines customer satisfaction based on the level of customer perception (Bitner, 1990; Zeithamal and Bitner, 1996).

(3) Customer Loyalty

Customer loyalty has been studied with a wide scope by a number of scholars. The existing studies on customer loyalty can be roughly categorized into behavioral, attitudinal, and integrated approaches (Oh, 1995). The behavioral approach examines the continuity of customer purchase history and deals with customer loyalty in association with the rate, frequency, and probability of purchase (Jeuland, 1979; Raj, 1982). The attitudinal approach interprets customer loyalty as psychological immersion, preference, and favorable attitude toward particular products or services (Jacoby and Chestnut, 1978; Oh, 1995; Oliva et al., 1992). Combining these two approaches, the integrated approach uses customer behaviors and attitudes to conceptualize customer loyalty and is used as a valuable means to understand customer loyalty in various industrial fields (Dick and Basu, 1994). Customer loyalty is sometimes manipulated as behaviors and attitudes, and the attitudinal criteria include recommendations of brands, resistance against superior competitive alternatives, intention to buy again, and intention to pay for the premium (Anderson & Sullivan, 1993; Boulding et al., 1993; Cronin & Taylor, 1992; Narayandas, 1996; Zeithaml et al., 1996). Based on such studies, this thesis defines customer loyalty as the sum of 7 behavioral patterns and uses them as a means of measurement: superiority as to mobile phone brand images, favorable attitudes toward mobile phone brands, pride in using mobile phones, resistance against superior competitive alternatives, recommendations of products being used, intention to buy again, and intention to pay for the premium.

III. Positive Analysis

A. *Research Model and Hypothesis*

On the basis of the results of the previous studies, this thesis categorizes customer satisfaction into expected satisfaction in the middle of evaluation before purchase and actual satisfaction comingfrom purchase experiences and illustrates the relationship with customer loyalty as shown in Figure 1.To emphasize use Emphasis and Strong styles. For code fragments use Code style, for URLs standard Hyperlink style, for file names – File Name style.

Satisfaction about products is divided into expected satisfaction in the course of evaluation before purchase and actu-

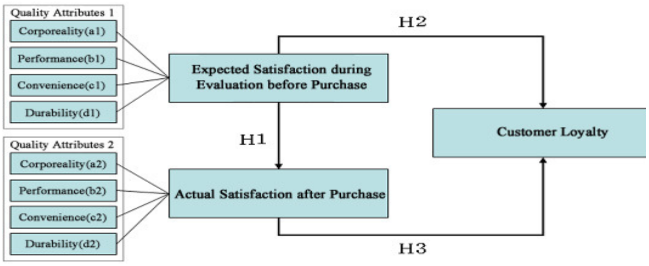


Fig. 1 Research Model

al satisfaction coming from purchase experiences (Yi, 1990), and customer satisfaction about products relies on customer expectations, increasing when such expectations exceed the quality of products (Churchill& Surprenant, 1982). In addition, Oliver (1980) defined customer satisfaction as the function of expectations met and perceptions of discrepancies, suggesting a discrepancy paradigm. Based on such studies, hypotheses 1 and 2 were established as follows:H1. The level of expected satisfaction in the course of evaluation before the purchase of mobile communication terminals is expected to have a positive influence on the level of actual satisfaction coming from purchase experiences. H2. The level of expected satisfaction in the course of evaluation before the purchase of mobile communication terminals will have a positive influence on the level of customer loyalty. According to the studies done by Westbrook & Reilly (1983), customer satisfaction is defined as a psychological response resulted from experiences associated with the purchase of products or services, and such attitudes are related to customer loyalty including psychological immersion, preference, and favorable attitude toward the corresponding products (Jacoby and Chestnut, 1978; Oh, 1995; Oliva et al., 1992). Hypothesis 3 was then established based on the results of such studies.H3. The level of actual satisfaction obtained from the purchase of mobile communication terminals is expected to have a positive influence on the level of customer loyalty. On the basis of the 3 hypotheses stated above, this thesis will analyze the relationship between variables through the SEM (Structural Equation Model) with the focus placed on factors like the level of expected satisfaction in the course of evaluation before purchase, the level of customer satisfaction obtained from purchase experiences, and the interrelated influence on and causal relationship between various levels of customer loyalty.

B. *Makeup of Measured Items*

The manipulative definition of variables used to test the hypotheses established in this thesis is based on the suggestions made by the previous studies. This research selected 8 attribute clues from22 attribute clues of SERVQUAL closely associated with the unique attributes of mobile communication terminals through interviews with experts in mobile communication. The 4 factors that affect the level of customer satisfaction in the course of evaluation before purchase consist of 8 items: corporeality (2 items), performance (2 items), convenience (2 items), and durability (2 items). In addition, the 4 factors that affect the level of customer satisfaction after purchase consist of 8 items: corporeality (2 items), performance (2 items), convenience (2 items), and durability (2 items), making the total of 8 factors and 16 items. Also, the factors that affect customer loyalty were measured as 7 items. Expected satisfaction in the course of evaluation before purchase, actual satisfaction obtained from purchase experiences, and customer loyalty were measured by the Likert 7-score criteria, with score 1 being "Not at all" and score 5 being "Very much".

C. *Data Collection and Analysis*

1,000 male and female consumers ranging from 10's to 60's across the nation including small and medium-sized cities who own mobile communication terminals were interviewed in person, and samples were assigned and extracted in consideration of gender, age, region, and the occupancy ratio in the domestic mobile communication service market. SPSS was used to statistically process the questionnaire results, and an analysis was made as follows: First, the demographic density of respondents was analyzed based on the collected data so that the representativeness of the population could be checked. Second, credibility was checked with the Cronbach's alpha value by using SPSS in order to check whether the actual concept measured corresponds to the abstract concept to be measured in the measurement model (measurement variable) of this research, and then the research model was analyzed by using Amos 4.0 of the covariance structure analysis in order to examine the determinant factors that affect customer loyalty.

D. *Results of the Analysis*

(1) Characteristics of the Sample Data

(A) Distribution of Gender and Age

Among the 1,000 respondents who own mobile communication terminals, there were 472 men (47.2%) and 528 women (52.8%) while there were 155 people in the 10's (15.5%), 328 people in the 20's (32.8%), 291 people in the 30's (29.1%), 193 people in the 40's (19.3%), 30 people in the 50's (3%), and 3 people in the 60's (0.3%).

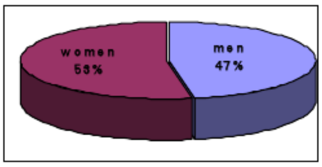


Fig. 2 Research Model

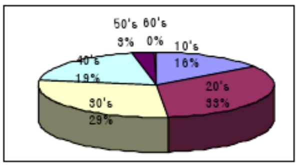


Fig. 3 Research Model

TABLE 2
DISTRIBUTION OF GENDER AND AGE

	10's	20's	30's	40's	50's	60's	Total (ratio)
Number of Men	71	153	142	95	9	2	472 (47.2%)
Number of Women	84	175	149	98	21	1	528 (52.8%)
Number of Samples (ratio)	155 (15.5%)	328 (32.8%)	291 (29.1%)	193 (19.3%)	30 (3%)	3 (0.3%)	1,000 (100%)

(B) Distribution of Terminal Manufacturers

As for the terminal manufactures that are currently selling mobile communication terminals in the domestic market, those whose names start with S, L, ST, P, M, and K were targeted, and 453 respondents turned out to own terminals manufactured by company S (45.3%), 179 by company L (17.9%), 107 by company ST (10.7%), 109 by company P (10.9%), 77 by company K (7.7%), 65 by company M (6.5%), and 10 by other manufacturers (1%).

Table 3
Distribution of Mobile Phone Manufacturers

	S	L	ST	P	K	M	Others	Total
Number of Samples (%)	453 (45.3)	179 (17.9)	107 (10.7)	109 (10.9)	77 (7.7)	65 (6.5)	10 (1)	1000 (100)
Population Ratio	46.87	20.5	16.38	6.92	4.66	2.66	2.00	100%

(C) Distribution of Mobile Communication Service Providers

The mobile communication service providers in the domestic market include SK Telecom, KTF, and LG Telecom, and 507 respondents turned out to have made a contract with SK Telecom (50.7%), 328 with KTF (32.8%), and 165 with LG Telecom (16.5%) in consideration of the occupancy ratio of each provider in the domestic market.

Table 4 Distribution of Mobile Communication Service Providers

Category	SK Telecom	KTF	LG Telecom	Total
Number of Samples (%)	507 (50.7)	328 (32.8)	165 (16.5)	1,000 (100)
Population (ratio)	52%	32%	16%	100%

(2) Analysis of Credibility

As means to test the credibility of variables, the split-half method and the internal consistency analysis are used to compute the credibility coefficient as to measured items. The most widely used method is the internal consistency analysis in that it tests credibility in consideration of the average relationship between items within each measurement device in case a number of items are used to measure the identical concept. In particular, in case a measurement variable consists of a number of items, the method using the Cronbach's alpha tests whether the component items are formed to measure the identical concept, enhancing credibility by excluding the items that hamper credibility from the measurement device. In this research credibility was tested by the Cronbach's alpha, and the results are shown in Table 5.

Table 5
Credibility Coefficients

Name of Variables		Number of Items	Cronbach' Alpha
Expected Satisfaction during Evaluation before Purchase	Corporeality	2	.7679
	Performance	2	.7457
	Convenience	2	.6877
	Durability	2	.6279
Actual Satisfaction after Purchase	Corporeality	2	.8928
	Performance	2	.7228
	Convenience	2	.7457
	Durability	2	.6993

The standards for credibility vary among a number of scholars, but in general, if the Cronbach's alpha coefficient exceeds 0.6, no one doubts the level of credibility, and if the Cronbach's alpha coefficient is over 0.8, the level of credibility is said to be significantly high. Most of the Cronbach's alpha coefficients for the measurement variables used in this research exceed 0.6, making all variables credible.

(3) Testing Hypotheses

The method used to analyze the relationship between a number of variables using the correlation matrix or the covariance matrix in a synthetic manner is referred to as the "multivariate analysis", and the covariance structure model is a model that integrates various methods of the multivariate analysis. That is, the multiple regression analysis is possible if there are more than 2 independent variables and only 1 dependent variable whereas the multivariate analysis is a proper method to analyze the causal relationship between variables if there are more than 2 dependent variables. LISREL has been widely used up until now for the covariance analysis, but it has several drawbacks including the complexity in entering data or processing statements. To complement such drawbacks, therefore, Amos was used to analyze the structural equation model in that the interface works well on the Windows platform and the SPSS or Excel worksheet data are easily retrieved. The causal relationship between variables is obtained by the causal coefficient (estimation) or the correlation coefficient of each path, and the level of significance needs to be evaluated. As a means of evaluation, the Wald test is often used where the hypothesis that the path coefficient is zero or there is no causal relationship is tested. This is based on the fact that the estimation divided by the standard error becomes the t distribution. If the number of samples is significantly large, the distribution is regarded normal, so if the estimation divided by the standard error is over 1.96, the hypothesis is accepted at 5% of the significance level and it can be said that there is a causal relationship. In Amos this is output as the critical ratio. The structural equation model analyzed in this thesis to test the hypotheses is shown in Figure 6.

.Fig. 4
Covariance Structural Model

Research Hypothesis		Estimate	S.E	C.R	P	Acceptance
H1	Expected Satisfaction during Evaluation before Purchase → Actual Satisfaction after Purchase	0.737	0.070	10.545	0.000	Accepted
H2	Expected Satisfaction during Evaluation before Purchase → Customer Loyalty	0.411	0.100	4.125	0.000	Accepted
H3	Actual Satisfaction after Purchase → Customer Loyalty	0.915	0.115	7.987	0.000	Accepted

The results of the covariance structural analysis between variables are shown in Table 6. * Accepted if the critical ratio is over 1.96 and the level of significance (P) is below 0.05, C.R. = Estimate / S.E.As a result of the covariance structural analysis as shown in Table 6, the hypotheses proposed in this research (H1, H2, and H3) were all accepted.

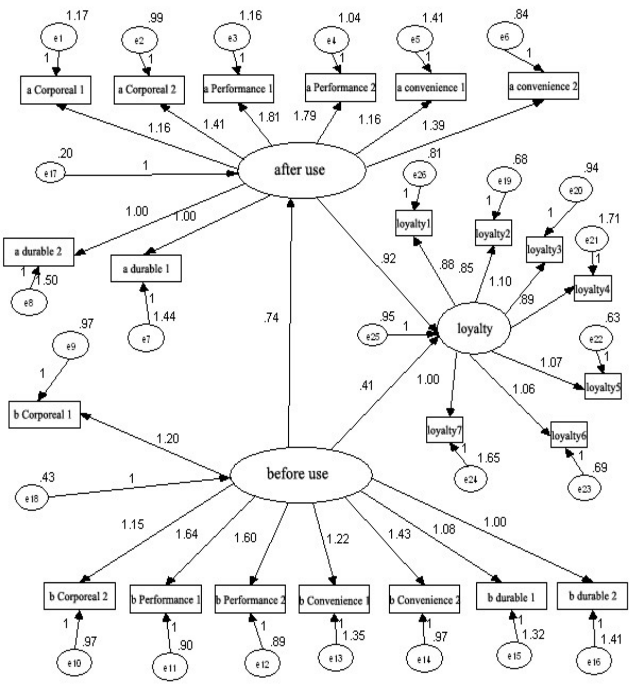


Table 6 Results of the Covariance Structural Analysis

First, hypothesis 1 states that the level of expected satisfaction in the course of evaluation before the purchase of mobile communication terminals is expected to have a positive influence on the level of actual satisfactionobtained from purchase experiences. As a result of the analysis, the level of expected satisfaction in the course of evaluation before the purchase of mobile communication terminals turned out to have 0.737 of the estimation and 10.545 of the critical ratio, which are similar to those of actual satisfaction obtained from purchase experiences. Accordingly, hypothesis 1 is eligible for acceptance. Second, hypothesis 2 states that the level of expected satisfaction in the course of evaluation before the purchase of mobile communication terminals is expected to have a positive influence on the level of customer loyalty. As a result of the analysis, the level of expected satisfaction in the course of evaluation before purchase turned out to have 0.411 of the estimation and 4.125 of the critical ratio, which are similar to those of customer loyalty. Accordingly, hypothesis 2 is eligible for acceptance. H3. The level of actual satisfaction obtained from the purchase of mobile communication terminals is expected to have a positive influence on the level of customer loyalty. As a result of the analysis, the level of actual satisfaction obtained from purchase experiences turned out to have 0.915 of the estimation and 7.985 of the critical ratio, which are similar to those of customer loyalty. Accordingly, hypothesis 3 is eligible for acceptance

IV. Conclusion

This research was intended to positively analyze factors that affect customer satisfaction about products in the mobile communication terminal market as to the influences of expected satisfaction and actual satisfaction before and after purchase on customer loyalty. As a means to measure the relationship between related variables, Amos 4.0 was implemented to test the structural equation model hypothesized in this research. Hypotheses 1, 2, and 3 were all accepted while each of the measurementvariables describing the level of expected satisfaction before the purchase of products, the level of actual satisfaction after the purchase of products, and the level of customer loyalty was also over 1 with a high level of credibility. The results of this research point out many important aspects of product designs, sales, and consistent customer services managed by the R&D department and the marketing department at a mobile communication terminal manufacturer. First, for hypothesis 1 the influence of expected satisfaction before the purchase of mobile communication terminals on actual satisfaction after the purchase of products turned out to be 0.737, which has an extremely high level of significance. This implies that when customers planning to buy mobile phones have a high expectation for the shape, design, performance, convenience, and durability, actual satisfaction about such quality elements after the purchase of products also increases. Thus, mobile phone manufacturers must pour their efforts into the elevation of customer satisfaction about the unique quality attributes of mobile communication terminals at the stages of product design and production. In addition, accurate information on the basic attributes of mobile phones must be provided in that the level of customer satisfaction may drop rapidly in case the actual functions and performance of mobile phones differ from those publicly advertised. Second, as for the influence of expected satisfaction before the purchase of mobile communication terminals on customer loyalty, the estimation is 0.411, which has a fairly high level of significance. As customers have a certain level of expectation for the quality of products through the preliminary product information provided by mobile phone manufacturers, strategic methods must be established so that the information on the unique features of products can be provided to customers through advertisements. As the mobile communication terminal technology rapidly develops, new attributes start to take their forms while convergence products that combine the existing attributes also appear. Wibro, developed locally first in the world and ready for commercialization, is now breaking the boundary between wired and wireless communication by combining the highly efficient transmission capability of wired communication and the mobility of wireless communication. Moreover, the terrestrial DMB service expected to lead the era when communication and broadcasting are combined was commercially provided in the domestic market, first in the world, drawing attention from all around the world. According to such convergence trends for mobile communication terminals, customers acquire information on new products with attributes different from those of the existing products through a variety of channels. Furthermore, as suggested by this research, customer expectations for the quality of products whose information is obtained before purchase through various channels were analyzed to have a direct influence on customer loyalty. Therefore, mobile phone manufacturers must enhance the level of customer satisfaction by providing information on the unique quality attributes of products ultimately to improve the level of customer loyalty. Third, as for the influence of actual satisfaction after the purchase of mobile communication terminals on the level of customer loyalty, the estimation turned out to be 0.915, which has a fairly high level of significance, and this helps conclude that mobile phone manufactures need to pour their efforts into the enhancement of product quality in that customer satisfaction after purchase is directly proportional to customer loyalty and that customer loyalty may stimulate the intention to buy again. As suggested by the positive analysis, mobile phone manufacturers must strive to develop and manufacture mobile communication terminals with enhanced designs, performance, convenience, and durability. Mobile phone manufacturers have recently focused on the R&D and designs of mobile communication terminals. Nokia, with the No. 1 occupancy rate in the global market, have been making utmost efforts to manufacture products ranging from monoblock to cramshell and slide devices under the motto, "To satisfy the customer and the operator". Ever since the year 1993, Samsung Electronics, one of the domestic manufacturers, has hired more than 1,000 designers for the Design Management Center, differentiating a group of products, establishing a consistent identity, and developing a design first.. In this research the uniquequality attributes of mobile phones were analyzed besides prices and brand images regarded, in general, as important factors that affect customer satisfaction. Prices and brand images are the most influential attributes applicable to almost all purchase circumstances. However, the two factors were excluded from this research in that such external factors had a possibility to cause conflict in the analysis of the unique quality attributes of products. Nevertheless, in the future research the previous studies need to be carefully reexamined while the balance of such factors is considered for a broader understanding and diverse research attempts. On the basis of the results of this research as well as the following research efforts as to customer satisfaction about mobile communication terminals, customer expectations for the quality of products must be met while actual satisfaction is achieved through the use of products. Also, manufacturers must pay close attention at all manufacturing stages to improve customer loyalty and maintain the competitive power of the domestic mobile communication terminal industry in the global market.

References

[1] IITA, Technology Policy Research Team, Mobile Communication Industry Monthly Statistics, Each Month, 2005, 2006

[2] IITA, Technology Policy Research Team, Information & Communication Technology Policy Research, Final Conclusion Report, 2005. 12

[3] IITA, Technology Policy Research Team, Domestic Mobile Communication Terminals, Investigation into and Analysis of Consumer Use Status, 2005 12

[4] Anderson, E. and M. W. Sullivan, "The Antecedents and Consequences of Customer Satisfaction for Firms", Marketing Science, 12(2), 1993, pp.125-43.

[5] Bitner, M., "Evaluating Service Encounters: The Effects of Physical Surroundings and Employee Responses," Journal of Marketing, 54(Apr), 1990, pp. 69-82.

[6] Boulding, W., A. Kalra, R. Staelin and V. A. Zeithaml, "A Dynamic Process Model of Service Quality from Expectations to Behavioral Intentions," Journal of Marketing Research, 30, Feb., 1993, 7-27.

[7] Cronin, J. J. and S. A. Taylor, "Measuring Service Quality: A Reexamination and Extension," Journal of Marketing, Jul., 1992, pp. 55-68.

[8] Dick, Alan S. and Kunal Basu., "Customer Loyalty : Toward an Integrated Conceptual Framework," Journal of the Academy of Marketing Science, 22(Spring), 1994, pp. 99-113.

[9] Jacoby, J. and R. W. Chestnut., Brand Loyalty: Measurement and Management, New York : Wiley, 1978.

[10] Jeuland, A. P., "Brand Choice Inertia as One Aspect of the Notion of Brand Loyalty," Management Science,25(Jul.), 1979, pp. 671-682.

[11] Narayandas, N., "The Link between Customer Satisfaction and Customer Loyalty: An Emprical Investigation", Working paper: 97-017, Harvard Business School, 1996.

[12] Oh, H. C., An Empirical Study of the Relationship Between Restaurant Image and Customer Loyalty, Unpublished Ph. D. Dissertation, Virginia Polytechnic Institute and State University, 1995.

[13] Oliver, R. L., "A Cognitive Model of The Antecedents and Consequences of Satisfaction Decisions," Journal of Marketing Research, Vol. 17, 1980, pp. 46-49.

[14] Oliva, Terence A., Richard L. Oliver, and Ian C. MacMillan, "A Catastrophe Model for Developing Service Satisfaction Strategies," Journal of Marketing, 56(July), 1992, pp. 83-95.

[15] Parasuraman, Zeithaml and Berry, "A Conceptual Model of Service Quality and Its Implications for Future Research," Journal of Marketing, Fall 1985, pp. 41-50.

[16] Parasuraman, Zeithaml and Berry, "SERVQUAL: A Multiple-Item Scale for Measuring Customer Perceptions of Service Quality," Journal of Retailing, Spring 1988, pp. 12-40.

[17] Raju, J. S., V. Srinivasan and R. Lal, "The Effect of Brand Loyalty on Competitive Price Promotional Strategies," Management Science, 36(3), 1990, pp. 267-304.

[18] Westbrook & Reilly (1983)Westbrook, R. A. and D. Reily, "Value-PerceptDisparity: An Alternative to the Disconfirmation of Expectations Theory of Consumer A Satisfaction," in Advances in Consumer Research, Richard P. Baggozi and Alice M. Tybout, eds. Ann Arbor, MI: Association for Consumer Research, 1983, pp. 256-261.

[19] Yi, Y., "A Critial Review of Customer Satisfaction", in Zeithaml, V. A.(Ed), Review of Marketing 1989, American Marketing Association, Chicago, IL., 1990.

[20] Zeithaml, V. A., "Consumer Perceptions of Price, Quality and Value: A Means-End Model and Synthesis of Evl-dence," Journal of Marketing, Vol. 52, 1988, pp.2-22.

[21] Zeithaml, V. A., L. L. Berry and A. Parasuraman, "The Behavioral Consequences of Service Quality," Journal of Marketing, 60(2) Apr., 1996.

[22] W.-K. Chen, Linear Networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[23] H. Poor, An Introduction to Signal Detection and Estimation. New York: Springer-Verlag, 1985, ch. 4.

[24] B. Smith, “An approach to graphs of linear forms (Unpublished work style),” unpublished.

[25] E. H. Miller, “A note on reflector arrays (Periodical style—Accepted for publication),” IEEE Trans. Antennas Propagat., to be published.

[26] J. Wang, “Fundamentals of erbium-doped fiber amplifiers arrays (Periodical style—Submitted for publication),” IEEE J. Quantum Electron., submitted for publication.

[27] C. J. Kaufman, Rocky Mountain Research Lab., Boulder, CO, private communication, May 1995.

[28] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interfaces(Translation Journals style),” IEEE Transl. J. Magn.Jpn., vol. 2, Aug. 1987, pp. 740–741 [Dig. 9th Annu. Conf. Magnetics Japan, 1982, p. 301].

[29] M. Young, The Techincal Writers Handbook. Mill Valley, CA: University Science, 1989.

Workshop on Ad-Hoc Wireless Networks

The extent to which wireless networking has succeeded as a practicable approach to telecommunication depends on the class of applications and the underlying technology. On one end of the "success spectrum" we have cellular telephony with its avalanche of existing and forthcoming services. On the other end, there are various (so far mostly hypothetical) applications and embodiments of ad-hoc networking, usually connotated with wireless sensing, which, despite enormous attention of the academic community, find it difficult to materialize in the real world. Our workshop will focus on practical aspects of wireless telecommunication, with stress on mobility, survivability, resilience, quality of service, and security. Its leitmotif will be an identification of the problems hampering progress in the "practically challenged" areas of wireless networking. Case studies, especially ones illustrating unorthodox solutions adopted in the context of specific practical applications are particularly encouraged. The non-exclusive list of topics includes:

- performance studies and comparisons
- network protocols
- cross-layered protocol design
- design and programming methodologies for network protocols and applications
- wireless sensor networks
- virtual execution platforms
- simulation and modeling methodologies
- security and reliability in wireless networks
- traffic patterns, including multimedia traffic
- multimodal networks
- satellite networks
- high-altitude platform networks

PROGRAM COMMITTEE

Dietmar Bruckner, Institute of Computer Technology, Vienna University of Technology, Austria

Franco Davoli, University of Genoa, Department of Communication, Computer, and System Sciences, Italy

Dietmar Dietrich, Institute of Computer Technology, Vienna University of Technology, Austria

Hyeonsang Eom, Seoul National University, KOREA

Pawel Gburzynski, University of Alberta, Department of Computing Science, Canada

Christoph Grimm, Institute of Computer Technology, Vienna University of Technology, Austria

Janelle Harms, University of Alberta, Department of Computing Science, Canada

Jerzy Konorski, Gdansk University of Technology, Poland

Taek Jin Kwon, Telcordia Technologies, Inc., USA

Ioannis Nikolaidis, University of Alberta, Department of Computing Science, Canada

Wladek Olesinski, Sun Labs Oracle, USA

Wlodzimierz Olesinski, Olsonet Communications, Canada

Joon-sang Park, Hongik University, KOREA

Ha Yoon Song, Hongik University, Korea, Republic of

Youjip Won, Hanyang University, KOREA

Jozef Wozniak, Gdansk University of Technology, Poland

Heimo Zeilinger, Institute of Computer Technology, Vienna University of Technology, Austria

Gerhard Zucker, Institute of Computer Technology, Vienna University of Technology, Austria

ORGANIZING COMMITTEE

Pawel Gburzynski, University of Alberta, Department of Computing Science, Canada

Jerzy Konorski, Gdansk University of Technology, Poland

Jozef Wozniak (Chairman), Gdansk University of Technology, Poland

ISBN 978-83-60810-27-9
ISSN 1896-7094

Wireless Transceiver for Control of Mobile Embedded Devices

Jan Kordas, Petr Wagner, Jiri Kotzian
Department of Measurement and Control, Faculty of Electrical Engineering and Computer Science
VŠB – Technical University of Ostrava
17. listopadu 15, Ostrava-Poruba 708 33, Czech Republic
{jan.kordas, petr.wagner, jiri.kotzian}@vsb.cz

***Abstract*—This article deals with the control of mobile embedded devices via wireless transceiver. The only way how to control the mobile devices such as robots, is to use a wireless data transfer. The possible wireless transceiver solution using the nRF24L01 transceiver by Nordic Semiconductor, which works in license-free worldwide 2.4 GHz ISM frequency band, is presented. Overview of this chip is included at the beginning of this article. The main part of this article deals with the design of a communication protocol. Then some optimizations of this protocol to improve its performance and determinism are discussed. In the last part, the results of measurement of some data transfer characteristics are presented.**

I. TIntroduction

HE wireless data transfer has been a very popular topic in recent years. It replaces the wire or optical connection in some applications. However, this article is focused on applications where a wire is impossible to use. The wireless connection is the only possibility in applications such as the control of the mobile embedded devices [1], e. g. robots. We can distinguish the two basic types of communication. In the first configuration of the wireless network, one stationary device is controlling one or more mobile devices. In the second configuration, all the devices are mobile.

This article discusses possible approaches of the control of the mobile embedded devices by the one stationary control system. This is intended to be used in the real-time application where a group of five robots is controlled by the stationary control system.

II. Design of Wireless Transceiver

The designed wireless transceiver is based on the nRF24L01 transceiver [3] which ensures the data transmission over the air. The nRF24L01 provides SPI (Serial Peripheral Interface) for connection with a target device. SPI is appropriate for the connection of the nRF24L01 with a microcontroller of the mobile device which has this interface also (this is the case of the mobile robots). However, it is not the suitable interface for the connection of control computer. So that SPI has to be converted to another more suitable interface for the connection of the nRF24L01 with a personal computer. USB (Universal Serial Bus) was chosen because each modern personal computer has available at least a few unused USB ports. The conversion from SPI to USB is done in two steps. At first, SPI is converted to UART. The conversion is made by the Freescale MC9S12D64 microcontroller which controls the nRF24L01 also. Any other microcontroller with the same or higher performance can be used. Then UART is converted to USB by the FTDI FT232RL converter. The block diagram of the wireless transceiver is shown in Fig. 1.

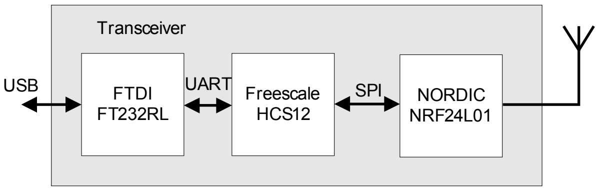


Fig. 1 Block diagram of wireless transceiver

I. Nordic nRF24L01

The nRF24L01 is a single chip 2.4GHz transceiver with an embedded baseband protocol engine (version with integrated MCU based on i8051 is also available), designed for ultra low power wireless applications. The nRF24L01 is designed for operation in the worldwide 2.4 GHz ISM frequency band. The air data rate is configurable to either 1 Mbps or 2 Mbps. The radio front end uses GFSK modulation. The chip has user configurable parameters like frequency channel, output power and air data rate.

1) Enhanced Shock-Burst™

The nRF24L01 provides Enhanced Shock-Burst™ mode beside manual operation mode. This mode allows 1 to 32 bytes dynamic payload length, automatic packet handling, auto packet transaction handling, data pipe MultiCeiver for 1:6 star networks.

2) MultiCeiver™

MultiCeiver™ is a feature used in receive mode that contains a set of 6 parallel data pipes with unique addresses. A data pipe is a logical channel in the physical RF channel. The nRF24L01 configured as primary receiver can receive data addressed to six different data pipes in one frequency channel. Each data pipe has its own unique address and can be configured for individual behavior. All data pipe address-

This work was supported by the GAČR project 102/08/1429 "Safety and security of networked embedded system applications" and by the internal project SP/2010165 "Development and using new methods for design and realization of embedded systems"

es are searched for simultaneously. The length of the messages can vary among data pipes [3].

II. Freescale MC9S12D64

The MC9S12D64 microcontroller [4] is a 16-bit device composed of standard on-chip peripherals including a 16-bit central processing unit, 64 Kbytes of Flash EEPROM for user program, 4 Kbytes of RAM for application data, 1 Kbyte of EEPROM as persistent data storage, two UARTs, one SPI, 29 discrete digital I/O channels, and so on. The inclusion of a PLL circuit allows power consumption and performance to be adjusted to suit operational requirements.

III. FTDI FT232RL

The FT232R is a single chip USB to asynchronous serial data transfer interface. UART interface supports 7 or 8 data bits, 1 or 2 stop bits and odd, even, mark, space, or no parity. Data transfer rates from 300 baud up to 3 Mbaud at TTL levels are supported. The entire USB protocol is handled on the chip, so that no USB specific firmware programming is required. The FT232RL supports bus powered, self powered, and high power bus powered USB configurations. For complete specification of the FTDI FT232RL please refer to [5].

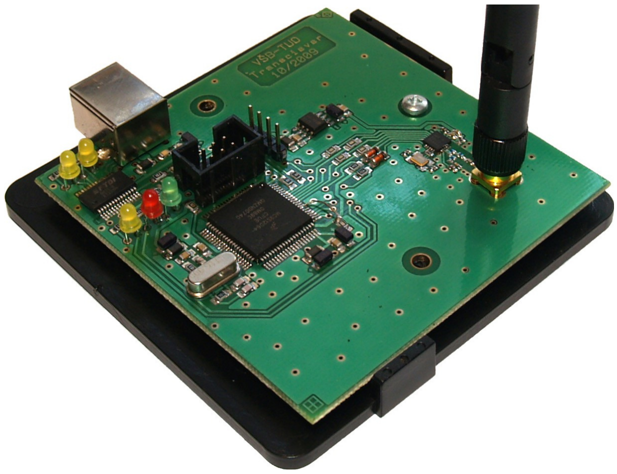


Fig. 2 Realized transceiver

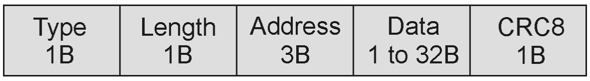
III. Communication Protocol

For systems where the real-time requirement should be met a deterministic access method should be used for channels which are not dedicated for the only two communicating devices [2]. The master-slave access method is appropriate for the robot control application because the prevailing data transfer direction is from the stationary device to the mobile devices. So the stationary device is the master and the mobile devices are the slaves.

We can distinguish the three types of messages which are sent from the personal computer to the wireless transceiver and then to the remote device:

1. Data for the mobile device.
2. Request for the mobile device status data.
3. Service messages to the wireless transceiver (these data are not sent over the air).

The data for mobile device are sent from the computer via the USB serial port to the microcontroller. This data are encapsulated by some extra information such as the type of the message, its length, CRC, etc. The structure of the message is shown in Fig. 3.



Type 1B	Length 1B	Address 3B	Data 1 to 32B	CRC8 1B

Fig. 3 Message structure

The data that are received and successfully checked by the microcontroller are unpacked (the header and footer are removed) and then they are sent directly to the nRF24L01. The nRF24L01 adds the destination address and CRC to the message and after that it sends this message over the air. A device that receives a correct message with its address sends the received message to the connected microcontroller on its request. The nRF24L01 is configured to send data with the 3 bytes address (the shortest possible option) and to secure the data by CRC16.

I. Protocol optimization

In the robot control application, two data pipes with different addresses can be used to distinguish the unicast and broadcast messages. The first data pipe is used for the unicast messages and the second data pipe is used for the broadcast messages. The unicast address is unique for each device in the net. The broadcast address is the same for all devices in the net.

The broadcast messages can be used only for the control messages where a response is not expected. The use of the broadcast messages instead of the unicast messages eliminates the delivery time delay and improves the determinism of the transmission time. The determinism is improved because all robots receive the message in the exactly same time. The improvement was tested and the results are presented in the next paragraph of this article.

The particular structure of the broadcast message is in Fig. 5. This message is inserted to the data part of the message shown in Fig. 3. The broadcast message contains data for up to 6 mobile robots. The data for each robot are identified by its address. The left and right wheel revolutions are the typical data for the robots. The command for each robot is the same hence it appears only once in the message. The counter is increased for the every new message. For better reliability of transmission, each message can be sent twice or more times without increasing of this counter so the remote device can recognize if the message has been sent before.

IV. Testing of wireless transceivers

Two wireless transceivers developed according to the previous design were tested. Two similar transceivers with the same configuration of the wireless part and the same antenna with gain 2 dBi were used for testing. The first wireless

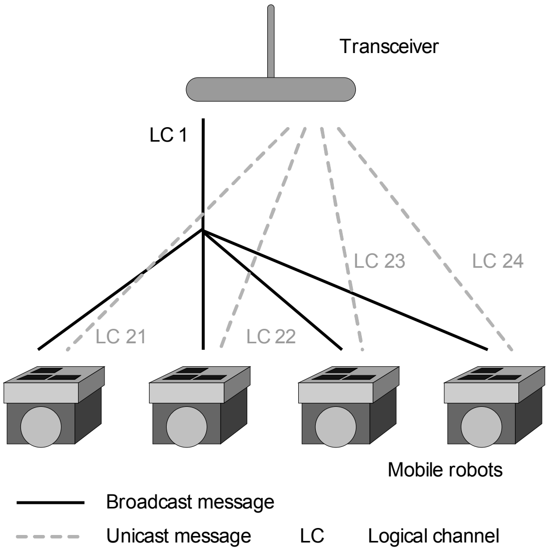


Fig. 4 Unicast and broadcast configuration

Counter 1B	Command 1B	1 to 6 times	
		Address 1B	Data 4B

Fig. 5 Structure of broadcast message

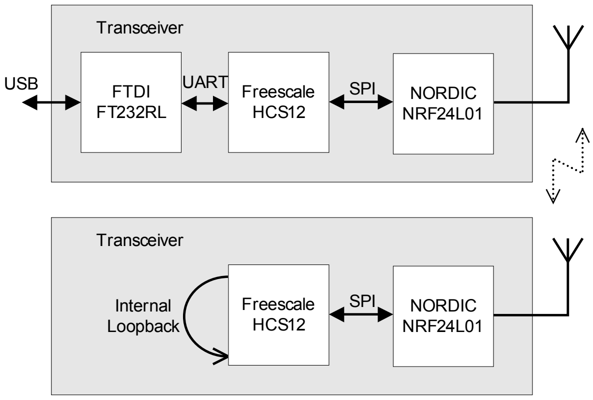


Fig. 6 Configuration of the experiment

transceiver had implemented the internal loopback on wireless interface while the second sent the data received from the wireless interface to USB and vice versa. The test configuration is shown in Fig. 6.

I. Delivery time

The first of all, the message delivery times via wireless interface were tested. For this measurement, the oscilloscope was used because very short delivery times were expected. The microcontroller, which is on wireless transceiver, sets (for 3 milliseconds) one of its outputs on the data sent event and the other output on the data received event. These events were measured by oscilloscope. The measurement was performed for various packet lengths and for the both air data rates supported by the nRF24L01. The results are in Table 1.

Table I.
Comparison of 1 Mbps and 2 Mbps

number of bytes	1 Mbps	2 Mbps
5	224 µs	202 µs
10	289 µs	220 µs
15	325 µs	239 µs
20	367 µs	259 µs
25	406 µs	277 µs
30	448 µs	297 µs

Notice that the delivery times for 2 Mbps are not a half of the delivery times for 1 Mbps. This is caused by the constant time that is necessary to set up the nRF24L01 to the transmit mode. This setup takes 150 microseconds according to the performed test. It is possible to send more than one message after setting the nRF24L01 to the transmit mode but the nRF24L01 must not be in the transmit mode for more than 4 milliseconds [3].

One byte at 1 Mbps is transmitted in 8 microseconds. At 2 Mbps, this time is a half. According to the test, there is 7 bytes overhead on the wireless interface. This overhead includes 3 bytes of address, 2 bytes of CRC, and 2 extra bytes (probably preamble).

1) Unicast vs. broadcast messages delivery time

From the results in Table 1, we can calculate the difference in the delivery time DT between the unicast and broadcast messages. For this calculation, we use equation (1), where the setup_time is 150 us, the overhead_length is 7 and the byte_time is 8 us for 1 Mbps and 4 us for 2 Mbps.

When we use the unicast messages we have to send five messages with 5 bytes payload. This takes 630 us at 1 Mbps and 390 us at 2 Mbps. When we use the broadcast messages we have to send one message with 32 bytes. This takes 462 us at 1 Mbps and 306 us at 2 Mbps.

$$DT = setup_{time} + msgs_{cnt} \cdot \text{¿}$$
$$\text{¿¿}(overhead_{length} + paylod_{length}) \cdot byte_{time} \quad (1)$$

The calculated unicast delivery times are only theoretical because all messages must be sent together after one setup of the nRF24L01 to the transmit mode. In reality, each message is send separately and this takes 1.2 ms at 1 Mbps and 1.0 ms at 2 Mbps.

The theoretical difference between the unicast messages and the broadcast messages is not significant but the practical difference is considerable. The real delivery time is three times greater for the unicast messages than for the broadcast messages.

II. Lost messages

The lost of the messages was tested by sending one message per second from the PC during 10 minutes to the wireless transceiver that sent this message to the second wireless

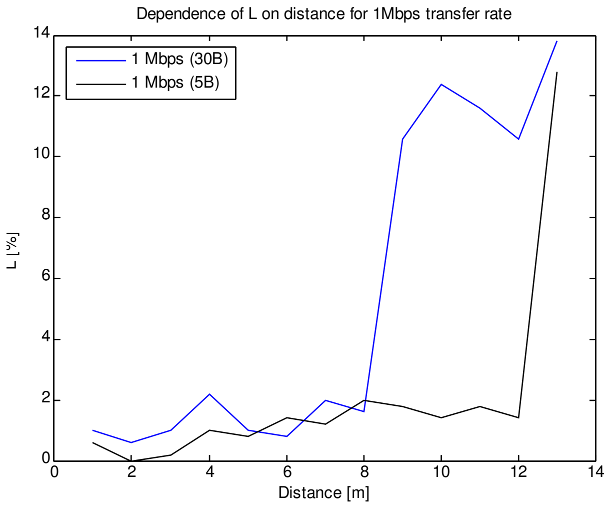


Fig. 7 Message lost ratio (1 Mbps)

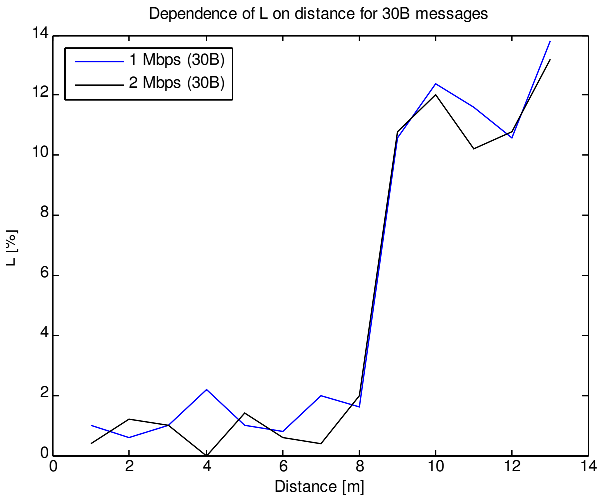


Fig. 8 Message lost ratio (30B messages)

transceiver with internal loopback. The second wireless transceiver only sent the received message back so sender should receive the sent message back. The message lost ratio L is given by equation (2).

$$L=1-\frac{received_{messages}}{sent_{messages}} \tag{2}$$

The lost of the messages was tested for both air data rates supported by the nRF24L01, for two lengths of message, and for various wireless transceiver distances. The dependence of the lost messages ratio on distance for 1 Mbps air data rate and for 5 and 30 bytes messages is shown in Fig. 7. And the dependence on distance for 30B messages for 1 and 2 Mbps air data rate is shown in Fig. 8. We can see that the dependence of L on the distance becomes significant for a distance over 8 meters for 30 bytes messages and for a distance over 12 meters of 5 bytes messages, respectively. The air data rate does not significantly affect the lost ratio.

V. Conclusion

The described wireless transceivers were constructed from components and then they were tested. According to the tests, the new design of the wireless system for the control of mobile embedded devices is suitable. Using the communication protocol optimization, the significant improvement has been done. The same information can be sent (using the broadcast messages and air data rate 2 Mbps) in one third time in comparison to the previous solution (using the unicast messages and air data rate 1 Mbps). Each of the controlled robots receives its piece of information in the same time as its team-mates. Furthermore, the use of the broadcast messages allows sending the same message more times so the information delivery probability can be increased. All of these optimizations help to control the mobile embedded devices more accurately which was the objective of our work.

References

[1] V. Srovnal Jr., Z. Machacek, V. Srovnal, “Wireless Communication for Mobile Robotics and Industrial Embedded Devices”, in *Proc. Eighth International Conference on Networks*, pp. 253-258, 2009.

[2] S. Sliva, V. Srovnal, “Distributed processing applicable to embedded systems”, in *Proc. 7th International Conference on Automatic Control, Modeling and Simulation*, pp. 75-58, 2005.

[3] *nRF24L01 Product Specification*, Nordic Semiconductor, rev. 2.0, 2007.

[4] *MC9S12DJ64 Device User Guide*, Freescale Semiconductor, rev. 1.20, 2005.

[5] *FT232R USB UART IC Datasheet*, Future Technology Devices International, rev. 2.02, 2009.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Efficient Coloring of Wireless Ad Hoc Networks With Diminished Transmitter Power.

Krzysztof Krzywdziński
Faculty of Mathematics and Computer Science
Adam Mickiewicz University, 60–769 Poznań, Poland
Email: kkrzywd@amu.edu.pl

***Abstract*—In our work we present a new approach to the problem of channel assignment in Wireless Ad Hoc Network. We introduce a new algorithm which works in distributed model of computations on Unit Disc Graphs modeling the Wireless Ad Hoc Network. The algorithm first modifies the transmitting power of the devices constituting the network (radii of the vertices of the Unit Disc Graph) and then it assigns the frequencies in the network adjusted to our demands. We assume that initially all the devices have the same transmission range and we are able to reduce the transmission range of some of them in order to decrease the number of necessary frequencies. We are able to diminish the number of communication links without loosing connectivity of the network, i.e. reduce the number of possible interference threats without risk of loosing possibility to exchange information. In addition, the diminution of communication range reduce the power consumption of the transmitting devices.**

***Index Terms*—Diminished Transmitter Power, Distributed Algorithm, Wireless Ad Hoc Networks, Coloring**

I. Introduction

THE WIRELESS Ad Hoc Network consists of the devices having computational ability placed on the given area (for example wireless cellphone network formed by the cellphones). We consider the problem of channel assignment in the Wireless Ad Hoc Network. In the proper channel assignment each vertex transmits information using one channel and without interference. An interference may occur due to two possible collisions. The primary one have place when a vertex simultaneously transmits and receives signals over the same channel. The secondary one occurs when a vertex simultaneously receives more than one signals over the same channel. Thus, to prevent the primary collision, two vertices may be assigned the same channel if neither of them is within the transmission range of the other. Similarly, to prevent the secondary collision, two vertices can be assigned the same channel if no other vertex is located in the intersection of their transmission ranges. The crucial problem concerning Wireless Ad Hoc Networks is finding the appropriate channel assignment. This problem is equivalent to determining optimal proper coloring of the square of graph representing the network.

In the standard model of the Ad Hoc Network each device of the network is placed in some point of the given area and all the devices have the same transmitting power. Therefore a widely used model for such a network is the Unit Disc Graph (UDG) model. The Unit Disc Graph is a graph in which the set of vertices is a set of vertices on the plane and two vertices are connected by an edge if and only if they are at the distance at most one in Euclidian norm. The optimal coloring (which bound from below minimum number of channels in channel assignment) of any Unit Disk Graph uses at least $\omega(G)$ colors, where $\omega(G)$ is the size of the largest clique. Moreover the chromatic number of UDG G $(\chi(G))$ is between $\omega(G)$ and $3\omega(G)$ (see [1]). Obviously, for dense graphs, the number $\omega(G)$ may be large, thus the number of channels necessary to ensure appropriate communication is large as well. The algorithms, working in various models, finding proper colorings of the Disk Graphs may be found in [3], [5], [6] and [11].

Surely, the absorbing issue is how to change the characteristics of the Wireless Ad Hoc Network which is represented by the graph with a large chromatic number in order to assure good communication using small number of frequencies. An effective approach to minimize the number of assigned channels is to diminish power of the transmitters installed in the devices (make the radius of transmission smaller). This concept has already appeared to be beneficial in total energy reducing problem (see [4],[9],[8],[10],[14],[17],[18],[19]) and interference reduction (see [2],[7],[12]). Diminishing power of transmitters enables to save energy of the batteries in devices, since during the transmission the full power of the transmitters is not used. Moreover such procedure reduce number of edges in the corresponding Unit Disc Graph and, since the graph is sparser, decrease the chromatic number. However diminishing transmission power may have disadvantageous impact on the connectivity of the network. Due to reduction of the number of active links the network may be partitioned (i.e. the corresponding graph would become disconnected) and some messages may never be delivered. Therefore our goal is to create an efficient algorithm which diminish powers of transmitters keeping strong connectivity of the graph and significantly reducing the chromatic number of the graph. To the best of our knowledge, in our algorithm the concept of diminishing transmitting power for solving the problem of channel assignment is used for the first time. Moreover this is fast distributed topology control algorithm which combine ideas of coloring and reducing total energy.

Our work is organized as follows: in Section II we introduce the model of computation in which the algorithm works. In Section III we sketch the main results of our work. In

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Section IV we present the main algorithm and show that it works properly. In the last section we show that the number of colors used by our algorithm is the best possible.

II. Model

The presented algorithm will work in distributed synchronous model of computations on Unit Disc Graphs with possible reduction of radii length. The Unit Disc Graphs (UDG) is a wiely used Wireless Ad Hoc Networks. The set of vertices of UDG is a set of vertices on the plane and any vertex v is connected by an edge with the vertex w if and only if the vertex w is contained in the disk of radius 1 with a center in v. In our model we make also additional assumption that every vertex can diminish its radius. We introduce a function $f : V(G) \to (0, 1]$ assigning to each vertex its reduced radius. The value $f(v)$ will be interpreted as the power of the transmitter v. Given the Unit Disc Graph G and the function $f : V(G) \to (0, 1]$ we define the directed Disc Graph $G\langle f\rangle$ as follows: $V(G\langle f\rangle) = V(G)$ and in $G\langle f\rangle$ there is an edge pointing from v to w if and only if w is contained in the circle of radius $f(v)$ and a center in v (i.e. $vw \in E(G\langle f\rangle) \Leftrightarrow \|v, w\| \leq f(v)$). The existence of the edge vw in $G\langle f\rangle$ is interpreted as the fact that vertex v can send information to w (w is in the communication range of v).

We make also additional assumption that all the devices constituting the network are equipped with the Global Positioning System (GPS), or know their position on the plane by other sources. We also assume that the network is synchronized and in one round a vertex can send, receive messages from its neighbors and can perform some local computations. Neither the amount of local computations nor the size of messages sent is restricted in any way. Concluding, the algorithm will work in a synchronous, message-passing model of computations introduced in [13], called LOCAL model, in which we additionally know the position on the plane of each element of the network.

III. Previous Work and Our Result

The presented algorithm DiminishPower works in the model introduced in Section II and finds a function f and channel assignment such that the network represented by $G\langle f\rangle$ is connected and in the network the number of necessary frequencies and energy consumption is significantly diminished comparing to the network with initial characteristics.

In order to formalize our arguments we give two additional definitions. We say that the directed graph $G\langle f\rangle$ is *strongly connected* if for all $v, w \in V(G\langle f\rangle)$ there exists a directed path form v to w. Strong connectivity of $G\langle f\rangle$ will imply the possible communication between any two devices constituting the network represented by $G\langle f\rangle$. The second definition concerns the amount of energy consumed by the network. By *total energy cost* we will mean the value $TE(G\langle f\rangle) = \sum_{v\in V(G\langle f\rangle)} f(v)^2$. The concept of saving energy in wireless networks was introduced in [15], [20]. Our definition of total energy cost includes the value f in second power since in the real two dimensional space the area of transmission is of order square of the power of the transmitter.

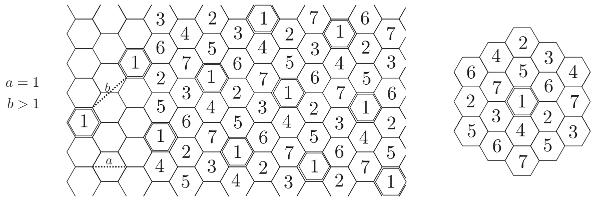


Fig. 1. Division of the plane into 7 classes of hexagons.

The idea of the algorithm DiminishPower is to divide the plane into 7 classes of hexagons with diameter 1 as in Figure 1. To each class we will attribute a different palette of colors. The subgraph of UDG G induced by the vertices contained in one hexagon form a clique. We will use that property of the division and in each hexagon we will run independently the procedure OnClique, which will locally color the vertices of the hexagon with colors from the palette attributed to that hexagon and assign the value of the function f to each vertex v from the given hexagon. The function will be such that the subgraph of $G\langle f\rangle$ induced by the vertices from a hexagon will be strongly connected and the coloring constructed by the procedure will be proper coloring of that subgraph. Finally we will join local colorings and obtain the proper colloring of $G\langle f\rangle$.

We will prove that for a given connected Unit Disc Graph G the algorithm DiminishPower finds in constant number of synchronous rounds in LOCAL model (see Theorem 8) a function $f : V(G) \to (0, 1]$ such that:

1) $G\langle f\rangle$ is strongly connected (see Theorem 5);
2) $G\langle f\rangle$ has proper coloring with $O(\log(\chi(G)))$ colors (see Theorem 6);
3) total energy cost of the wireless network represented by $G\langle f\rangle$ is of range

$$TE(G\langle f\rangle) = O\left(|MIS(G)| \log \frac{|V(G)|}{|MIS(G)|}\right)$$

(see Theorem IV) , where $|MIS(G)|$ is the size of the maximum independent set in G.

Moreover there exists a channel assignment with the above properties (see Remark 9). Another important property of the construction is possibility of defining an effective routing protocol in $G\langle f\rangle$ (see Remark 10).

In comparison M. Fussen, R. Wattenhofer and A. Zollinger in [7] present the Algorithm NCC which finds strongly connected $G_{ncc}\langle f\rangle$, i.e. the graph which has property 1. However the maximum degree in $G_{ncc}\langle f\rangle$ is $O(\log(|V(G)|))$ (in $G\langle f\rangle$ it is $O(\log(\omega(G)))$). In their approach they do not solve problems of coloring, channel assignment and do not compute total energy. The Algorithm NCC strictly base on one predefined global sink and its distributed implementation takes $O(diameter(G))$ synchronous rounds (our algorithm takes $O(1)$ time). Also the graph constructed by the Algorithm NCC do not allow to define fast routing protocol (see Remark 10).

Moreover vertices in one hop distance in G can be at distance of $O(diameter(G)\log(|V(G)|))$ hops in $G_{ncc}\langle f\rangle$ (in $G\langle f\rangle$ they are at most $O(\log(\omega(G)))$ hops away).

IV. Main algorithm

First we present the procedure OnClique, which will be implemented independently on each subgraph induced by the vertices contained in a hexagon. We assume that the input graph K is a clique, therefore information about all the vertices may be send to one vertex of the clique and all the computations may be performed by that vertex. In the algorithm and later on we will call a pair of vertices vw a *double directed edge* if in the directed graph there are edges pointing from v to w and pointing from w to v.

```
OnClique
Input: Unit Disc Graph $K$ which is a clique and palette $P(K)$ of colors.
Output: Disc Graph $K\langle f\rangle$ and its coloring.
 (1) Let $i := 1$, $M_i = V(K)$, $f \equiv 1$ and $K\langle f\rangle[M_i]$ be the graph induced by the set of vertices $M_i$.
 (2) For all $v \in M_i$ set $f(v) := \min_{w\in M_i, w\neq v}\|v,w\|$.
 (3) Color graph $K\langle f\rangle[M_i]$ using new 5 colors from palette $P(K)$.
 (4) Let $N$ be a subgraph of $K\langle f\rangle[M_i]$ induced by all double directed edges.
 (5) $i := i+1$.
 (6) Set $M_i$ to be the set of leaders of the connected components of $N$.
 (7) If $|M_i| \geq 2$ then go to step (2).
 (8) If $M_i = \{v\}$ then $f(v) := 1$ and color $v$ with a new color from $P(K)$.
 (9) Return $K\langle f\rangle[V(K)]$ with coloring.
```

It should be mentioned that it is possible to implement the step (3) of the procedure, since the graph $K\langle f\rangle[M_i]$ is planar (see the proof of Lemma 3).

In addition the graph N is always well defined (i.e. has at least one vertex) and, as a consequence, for all i the set M_i contains at least one element. It is necessary for correctness of the algorithm. It is enough to notice that for all M_i such that $|M_i| \geq 2$ and f defined as in (2) the graph $K\langle f\rangle[M_i]$ has at least one double directed edge. From definition of f in $K\langle f\rangle[M_i]$ each vertex has out–degree at least one, therefore in $K\langle f\rangle[M_i]$ there is at least one cycle (the last vertex from the longest path has an out–neighbour on that path). Moreover, from definition of f each cycle in $K\langle f\rangle[M_i]$ consists of double directed edge. More precisely if $v_1,\ldots,v_t,v_1$ is a cycle, then: $f(v_1) = \min_{w\in M_i, w\neq v_1}\|v_1,w\| = \|v_1,v_2\| \geq f(v_2) = \|v_2,v_3\| \geq \ldots \geq f(v_t) = \|v_t,v_1\| \geq f(v_1)$ thus all the inequalities above are equalities and all the edges of the cycle are double directed.

Lemma 1. *Let $V(K) = M_1 \supseteq M_2 \supseteq M_3 \supseteq \ldots \supseteq M_k$ be the sequence of all the subsets of $V(K)$ constructed by the algorithm* OnClique. *Then $|M_i| \geq 2\left|M_{(i+1)}\right|$, $k \leq \log_2(|V(K)|)$ and $|M_k| = 1$.*

Proof: Since N is a subgraph of $K\langle f\rangle[M_i]$ induced by some edges, each connected component of N contains at least two vertices. M_{i+1} contains exactly one vertex from each component of N. Therefore $|M_i| \geq 2\left|M_{(i+1)}\right|$ for all $1 \leq i \leq k$. Consequently we have $k \leq \log_2(|V(K)|)$. $|M_k| = 1$ since for all i the set M_i contains at least one element. ■

Lemma 2. *The graph $K\langle f\rangle$ constructed by the algorithm* OnClique *is strongly connected.*

Proof: Let $V(K) = M_1 \supseteq M_2 \supseteq M_3 \supseteq \ldots \supseteq M_k = \{m\}$ be the sequence of all the subsets of $V(K)$ constructed by the algorithm OnClique. Let $w = w_1$ be any vertex from $V(K)$. We only need to prove that in $K\langle f\rangle[V(K)]$ after the last iteration there is a directed path from w to m. This combined with the fact that in $K\langle f\rangle$ there are directed edges pointing from m to all other vertices (since $f(m) = 1$) implies strong connectivity.

In the first iteration in (2) we set $f(v) := \min_{w\in K, w\neq v}\|v,w\|$ for all $v \in V(K) = M_1$. By definition of $f(v)$ every vertex v in $K\langle f\rangle[M_1]$ have out-degree at least 1. Notice that this implies that there exists a directed path from $w = w_1$ to a vertex w_1' which is incident to a double directed edge. We just have to take w_1' – the last vertex of the longest directed path with origin in w_1 and notice that w_1' have to have a neighbour on that path. Thus w_1' is contained in directed cycle and, as mentioned before, cycles in $K\langle f\rangle[M_1]$ consist of double directed edges or are a double directed edges. If w_2 is the leader of the connected component of N (defined as in (4) during the first iteration) containing w_1', then there exists a directed path from w_1 to $w_2 \in M_2$.

Using the same reasoning but replacing M_1 by M_i ($i > 1$) and the first iteration by the i-th one, we can prove that there exists a directed path from $w_i \in M_i$ to $w_{i+1} \in M_{i+1}$ in $K\langle f\rangle[M_i]$ defined in the i-th iteration. Since $f(v)$ in the next iterations may only grow, therefore the above–considered directed paths from $w = w_1$ to w_2, from w_2 to w_3, …, from w_{k-1} to $w_k = m$ are also directed paths in the final graph $K\langle f\rangle[V(K)]$. ■

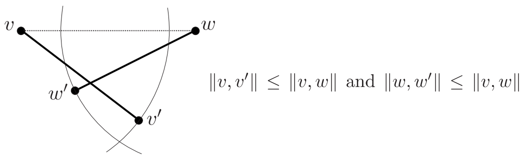


Fig. 2. $\|v,w'\| < \|v,v'\|$

Lemma 3. *The algorithm* OnClique *finds a coloring of the graph $K\langle f\rangle$ using at most $5\log_2(|V(K)|)$ colors from the palette.*

Proof: First observe that in the i-th iteration of the algorithm the graph $K\langle f\rangle[M_i]$ is plane. Otherwise there would exist two directed edges vv' and ww' which would cross each other. So by definition of f we know that $\|v,v'\| \leq \|v,w\| \,\&\, \|v,v'\| \leq \|v,w'\|$ and $\|w,w'\| \leq \|w,v\| \,\&\, \|w,w'\| \leq \|w,v'\|$. However if we suppose that $\|v,v'\| \leq \|v,w\|$ and $\|w,w'\| \leq \|v,w\|$ then w' would lie inside the circle with a center in v and the radius of length $\|v,v'\|$ (see Figure 2). So $\|v,w'\| < \|v,v'\|$, a contradiction.

It is possible to color any planar graph using 5 colors in polynomial time. In each iteration we use at most 5 new colors and if in iteration we enlarge the value of $f(v)$ then in the same iteration the color of v is changed and a new color have not been used in previous iterations. Therefore the output coloring is a proper coloring of $K\langle f\rangle$. By Lemma 1 we know that there are at most $\log_2(|V(K)|)$ iterations, thus the coloring use at most $5\log_2(|V(K)|)$ colors. ■

Lemma 4. *The graph $K\langle f\rangle$ constructed by the algorithm* OnClique *has total energy cost $TE(K\langle f\rangle) \leq 16\log_2(|V(K)|)$*

Proof: Let $V(K) = M_1 \supseteq M_2 \supseteq M_3 \supseteq \ldots \supseteq M_k$ be the sequence of the subsets of $V(K)$ constructed by the algorithm OnClique. Let $1 \leq i \leq k$ and $\{v_1, v_2, \ldots, v_{|M_i|}\} = M_i$. During the i-th iteration we set $f(v) := \min_{w\in M_i, w\neq v} \|v,w\|$ for all $v \in M_i$, therefore $f(v_t) \leq \|v_t,v_l\|$ and $f(v_l) \leq \|v_t,v_l\|$ for all $1 \leq k \neq l \leq |M_i|$. This implies $f(v_t)/2 + f(v_l)/2 \leq \|v_t,v_l\|$ for all $1 \leq k \neq l \leq |M_i|$. So, if we denote by $c(v,r)$ the disk with the center in v and the radius of length r, then all the disks $c(v_1, f(v_1)/2)$, $c(v_2, f(v_2)/2)$, …, $c(v_{|M_i|}, f\left(v_{|M_i|}\right)/2)$ are pairwise disjoint. Moreover K is a clique, thus v_1 is connected by an edge with all other vertices of K. Therefore, since K is UDG, all of the vertices from K are contained in $c(v_1, 1)$. This implies that all the disks $c(v_1, f(v_1)/2)$, $c(v_2, f(v_2)/2)$, …, $c(v_{|M_i|}, f\left(v_{|M_i|}\right)/2)$ lie inside $c(v_1, 2)$. Therefore

$$\pi 2^2 \geq \sum_{j=1,2,\ldots|M_i|} \pi \left(\frac{f(v_j)}{2}\right)^2.$$

So $TE(M_i) \leq 16$.

We use the same reasoning for all iterations and for all $v \in V(K)$ the value $f(v)$ from the outcome graph is set in one of those iterations, therefore we have $TE(K\langle f\rangle) \leq \sum_{j=0,1,\ldots,k} TE(M_i) \leq \sum_{j=0,1,\ldots,k} 16 = 16\log_2(|V(K)|)$. By Theorem 1 the number of iterations $k \leq \log_2(|V(K)|)$. ■

Now we are ready to introduce the main algorithm. In the algorithm, for all $i = 1, 2, \ldots 7$, we will define C_i to be the set of all connected components of the subgraph of the input UDG G induced by the vertices contained in the haxagons of the i-th class (see Figure 1). Obviously each of those connected components is a clique induced by the vertices from one hexagon of the i-th class.

```
DiminishPower
Input: Connected Unit Disc Graph $G$
Output: Disc Graph $G\langle f\rangle$ with proper coloring.
(1) Divide the set of vertices of $G$ into 7 classes as in Figure 1.
(2) For each $K \in C_i$ define palette of colors $P(K)$ as follows $P(K) = \{x \in \mathbb{N} : x = i \pmod 7\}$.
(3) Let $\mathcal{C} := C_1 \cup C_2 \cup \ldots \cup C_7$ For each $K \in \mathcal{C}$ parallel do:
    (a) Run algorithm OnClique on graph $K$ and color it using palette $P(K)$.
    (b) Define $\mathcal{K}' := \{K' \in \mathcal{C} : K' \neq K$ and $\exists_{k\in V(K), k'\in V(K')} kk' \in E(G)\}$. Denote the graphs from $\mathcal{K}'$ by $K'_1, K'_2, \ldots, K'_{|\mathcal{K}'|}$.
    (c) For $j = 0, 1, \ldots |\mathcal{K}'|$ do:
        (c1) Select one vertex $w_j \in V(K)$ such that $\exists_{w'_j \in V(K'_j)} w_j w'_j \in E(G)$.
        (c2) Color $w_j$ taking first free color from the palette $P(K)$
        (c3) Set $f(w_j) = 1$.
```

The following theorems show that the algorithm has the properties claimed in Section III.

Theorem 5. *Given the connected Unit Disk Graph G as an input, the algorithm* DiminishPower *finds the graph $G\langle f\rangle$ which is strongly connected.*

Proof: Let $K, K' \in \mathcal{C}$. By Lemma 2 and the fact that during the algorithm DiminishPower in step 3(c) we only may enlarge the radius of some vertices from $V(K)$ and $V(K')$, thus in the output graph $G\langle f\rangle$ the subgraphs induced on vertices from $V(K)$ and $V(K')$ are strongly connected. Moreover, if in G there were edges between $V(K)$ and $V(K')$, then after step 3(c3) in $G\langle f\rangle$ there is at least one edge pointing from $V(K)$ to $V(K')$ and at least one edge pointing from $V(K')$ to $V(K)$. Therefore, if G was connected, then $G\langle f\rangle$ is strongly connected. ■

Theorem 6. *Given Unit Disc Graph G algorithm* DiminishPower *finds a proper coloring for graph $G\langle f\rangle$ which use $O(\log(\chi(G)))$ colors.*

Proof: By Lemma 3 we know that in step (a) of the algorithm DiminishPower for each $K \in \mathcal{C}$ we use at most $5\log_2(|V(K)|)$ colors from palette $P(K)$. Moreover $|\mathcal{K}'| \leq 18$ (see Figure 1), therefore for any graph $K \in \mathcal{C}$ in step 3(c2) we use at most 18 colors from palette. Since there are 7 types of palettes and $|V(K)| \leq \omega(G)$ (K is a clique), during the whole algorithm at most $\max_{K\in\mathcal{C}} 7\left(5\log_2(|V(K)|) + 18\right) = O\left(\log(\omega(G))\right)$ colors are used. The result follows from the fact that $\omega(G) \leq \chi(G) \leq 3\omega(G)$ (see [1]).

By Lemma 3 the coloring constructed by the procedure OnClique is a proper coloring of the output graph of the procedure. Moreover in step 3(c), if we change the radius of the vertex, then we color that vertex with a new color, which have not been used before in K. Therefore, for any $K \in \mathcal{C}$, the coloring constructed by the algorithm DiminishPower is a proper coloring of the subgraph of $G\langle f\rangle$ induced by the vertices from $V(K)$. To conclude that this coloring is also a proper coloring of $G\langle f\rangle$ we only have to notice that in the communication range of the vertices from K exept other vertices from K there are only vertices from neighboring hexagons (see Figure 1). Those vertices are from different type then K, therefore their colors are from different palettes. ■

Theorem 7. *Given as an input a connected Unit Disc Graph G the algorithm* DiminishPower *finds graph $G\langle f\rangle$ such that the total energy cost of $G\langle f\rangle$ equals*

$$TE(G\langle f\rangle) = O\left(|MIS(G)|\log\left(\frac{|V(G)|}{|MIS(G)|}\right)\right),$$

where $|MIS(G)|$ is the size of the maximum independent set in G.

Proof: For each $K \in \mathcal{C}$, by Lemma 4 and the fact that in step 3(c) we enlarge the radius of at most 18 vertices we have that for the function f constructed by the algorithm DiminishPower

$$TE\left(G\langle f\rangle[V(K)]\right) \leq 16(\log_2(|V(K)|) + 18.$$

Therefore

$$\begin{aligned}TE(G\langle f\rangle) &= \sum_{K\in\mathcal{C}} [16(\log_2(|V(K)|) + 18] \\ &= 16\log_2\left(\prod_{K\in\mathcal{C}} |V(K)|\right) + 18|\mathcal{C}|.\end{aligned}$$

Now notice that $|MIS(G)| \leq |\mathcal{C}| \leq 7|MIS(G)|$, since in each $K \in \mathcal{C}$ there is at most one vertex from $MIS(G)$ and we can construct an independent set by choosing one vertex from each hexagon from one class. Moreover for any positive numbers a_i and a natural number n we have: $((a_1+a_2+\ldots+a_n)/n)^n \geq a_1a_2\ldots a_n$, therefore:

$$\begin{aligned}TE(G\langle f\rangle) &\leq 16\log_2\left(\frac{|V(G)|}{|\mathcal{C}|}\right)^{|\mathcal{C}|} + 18|\mathcal{C}| \\ &\leq 16\cdot 7|MIS(G)|\log_2\left(\frac{|V(G)|}{|MIS(G)|}\right) + 18|\mathcal{C}|\end{aligned}$$

■

Theorem 8. *The algorithm* DiminishPower *can be implemented in $LOCAL$ model in constant number of synchronous rounds.*

Proof: Since we assume that each vertex knows its position on the plane, step (1) (dividing into 7 classes) and step (2) (defining palettes of colors) of the algorithm DiminishPower can be easily implemented in one synchronous round. In addition each $K \in \mathcal{C}$ is a clique, thus information about the whole graph K may be send to a leader of K, which run the algorithm OnClique. From the properties of the $LOCAL$ model the leader can execute the algorithm OnClique in one synchronous round. Therefore OnClique can be implemented in four synchronous rounds: first we select a leader, next we send to him all information about the graph, then the leader run the algorithm OnClique and finally the leader send to the vertices of K the output of the algorithm). Finally $|\mathcal{K}'| \leq 18$, therefore the step (c) consists of at most 18 iterations. ■

As we claimed in Section I channel assignment problem is equivalent to coloring of the square of the graph. To obtain a proper channel assignment we will modify two steps in algorithm DiminishPower.

- In step (1) of the algorithm DiminishPower divide the set of vertices of G into 12 (instead of 7) hexagons (see Figure 1 in [16]).
- In step (3) of the algorithm OnClique we color square of the graph $K\langle f\rangle[M_i]$ with 37 colors from palette $P(K)$ (instead of 5) using greedy algorithm.

Remark 9. *The algorithm* DiminishPower *with above modifications constructs in constant number of synchonous rounds a graph $G\langle f\rangle$ with a proper channel assignment. Moreover it uses $O(\log(\chi(G)))$ channels and have properties claimed in Theorems IV and 5.*

Proof:

Firstly, in the partition into 12 classes any two vertices in the different hexagons of the same class are at least two hops away. So there is no channel collision between channels of vertices from different hexagons.

Secondly, in the i-th iteration of the algorithm OnClique the graph $K\langle f\rangle[M_i]$ has maximum degree bounded by 6. It follows from this observation that the maximal angle between edges in the graph $K\langle f\rangle[M_i]$ is $\frac{\pi}{3}$. Suppose that there would be two directed edges $wv, w'v \in E(K\langle f\rangle[M_i])$ such that angle $\angle(wvw') < \frac{\pi}{3}$ and $\|w,v\| \geq \|w',v\|$ then $\|w,w'\| < \|w,v\|$ and we arrive at the contradiction with a fact that $wv \in K\langle f\rangle[M_i]$. Therefore square of the graph $K\langle f\rangle[M_i]$ have degree bounded by 36. Thus using standard greedy algorithm it is simple to color square of $K\langle f\rangle[M_i]$ graph using 37 colors in polynomial time.

Concluding in channel assignment constructed by the modified algorithm DiminishPower we use at most $12\,(37\log_2(|\omega(G)|) + 18)$ channels. ■

Remark 10. *If we have a routing protocol in Unit Disc Graph G then we can easily define the routing protocol in $G\langle f\rangle$. Namely, if the routing protocol use the edge vw then we can use the directed path from v to w constructed as in the proof of Theorem 5. More precisely as w'_i we may take the first vertex from any path with origin in w_i, which is incident to a double directed edge in $G\langle f\rangle[M_i]$ defined as in the i-th iteration.*

V. Lower Bound

The chromatic number (and analogously channel assignment) of the graph constructed by the algorithm DiminishPower is best up to a constant factor. Namely there exists a connected Unit Disc Graph graph G such that for any function $f : V(G) \to (0,1]$ such that $G\langle f\rangle$ is strongly connected we have $\chi(G\langle f\rangle) = \Omega\left(\log\left(\chi(G)\right)\right)$. Our lower bound on chromatic number (and also channel assignment) is based on the idea of Theorem (4.1) [7] where authors shown a lower bound for maximum interference of $G_{ncc}\langle f\rangle$.

The construction of the example will be inspired by the construction of the Cantor Set. More precisely the set A_i will be the set of the end points of the segments of the i-th step of the construction of the Cantor Set. Formally $A_1 = \{0,1\}$, $A_2 = \left\{0,\frac{1}{3},\frac{2}{3},1\right\}$, $A_3 = \left\{0,\frac{1}{9},\frac{2}{9},\frac{1}{3},\frac{2}{3},\frac{7}{9},\frac{8}{9},1\right\}$ etc. as on Figure 3.

Let G be Unit Disc Graph with the vertex set A_k. Obviously G is a clique and have exactly 2^k vertices.

Theorem 11. *Let G be the Unit Disc Graph with the vertex set A_k and $f : V(G) \to (0, 1]$ be any function. If $G\langle f\rangle$ is strongly connected then $\chi(G\langle f\rangle) = \Omega(\log(\chi(G)))$.*

Proof: First define $G\{x_1, x_2\}$ to be the set of vertices of the graph G with x-coordinate between x_1 and x_2. Since $G\langle f\rangle$ is connected then there exists at least one directed edge e_1 from $v_1 \in G\langle f\rangle\{\frac{2}{3}, 1\}$ to $w_1 \in G\langle f\rangle\{0, \frac{1}{3}\}$. Without loosing generality suppose that $v_1 \in G\langle f\rangle\{\frac{2}{3}, \frac{7}{9}\}$ have color 1 (see Figure 3). Observe now that since $f(v_1) \geq \frac{1}{3}$ then all other vertices from $G\langle f\rangle\{\frac{2}{3}, 1\}$ cannot have color 1. Analogously there exists at least one directed edge e_2 from $v_2 \in G\langle f\rangle\{\frac{8}{9}, 1\}$ to $w_2 \in G\langle f\rangle\{\frac{2}{3}, \frac{7}{9}\}$. Without loosing generality suppose that $v_2 \in G\langle f\rangle\{\frac{26}{27}, 1\}$ and have color 2 (see Figure 3). Observe now that since $f(v_2) \geq \frac{1}{9}$ then all others vertices from $G\langle f\rangle\{\frac{8}{9}, 1\}$ cannot have color 2. If we continue the reasoning, then in the k-th step we notice that we must use at least k colors to color $G\langle f\rangle$. Since G is a clique, $\chi(G) = V(G) = 2^k$, so $\chi(G\langle f\rangle) \geq \log_2(\chi(G))$. ■

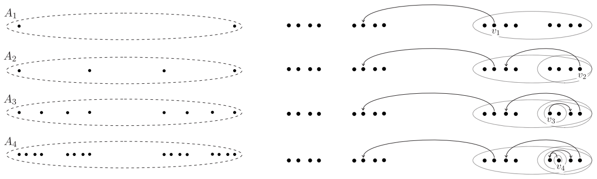


Fig. 3.

VI. Conclusion

In this paper we have studied the problem of channel assignment in Wireless Ad Hoc Network represented by unit disk graphs. We have diminished the number of communication links without loosing connectivity of the network. In addition we have shown, how the diminution of communication range reduces power consumption of transmitting devices. We also show that the number of colors used by our algorithm is the best possible.

References

[1] M. Stumpf A. Gräf and G. Weienfels. On coloring unit disk graphs. *Algorithmica*, 20:277293, 1998.

[2] Martin Burkhart, Pascal von Rickenbach, Roger Wattenhofer, and Aaron Zollinger. Does topology control reduce interference? In *MobiHoc '04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 9–19, New York, NY, USA, 2004. ACM.

[3] Ioannis Caragiannis, Aleksei V. Fishkin, Christos Kaklamanis, and Evi Papaioannou. A tight bound for online colouring of disk graphs. *Theor. Comput. Sci.*, 384(2-3):152–160, 2007.

[4] Mihaela Cardei, Jie Wu, and Shuhui Yang. Topology control in ad hoc wireless networks using cooperative communication. *IEEE Transactions on Mobile Computing*, 5(6):711–724, 2006.

[5] Mathieu Couture, Michel Barbeau, Prosenjit Bose, Paz Carmi, and Evangelos Kranakis. Location oblivious distributed unit disk graph coloring. In Giuseppe Prencipe and Shmuel Zaks, editors, *SIROCCO*, volume 4474 of *Lecture Notes in Computer Science*, pages 222–233. Springer, 2007.

[6] Thomas Erlebach and Jiří Fiala. Independence and coloring problems on intersection graphs of disks. *Efficient Approximation and Online Algorithms*, Volume 3484:135–155, 2006.

[7] Martin Fussen, Roger Wattenhofer, and Aaron Zollinger. Interference Arises at the Receiver. In *International Conference on Wireless Networks, Communications, and Mobile Computing (WIRELESSCOM), Maui, Hawaii, USA*, June 2005.

[8] Ting-Chao Hou and Victor Li. Transmission range control in multihop packet radio networks. *Communications, IEEE Transactions on Volume 34, Issue 1, Jan 1986 Page(s): 38 - 44*, 34(1):38–44, 1986.

[9] Marwan Krunz, Alaa Muqattash, and Sung ju Lee. Transmission power control in wireless ad hoc networks: challenges, solutions, and open issues. *IEEE Network*, 18:8–14, 2004.

[10] Ning Li, Jennifer C. Hou, and Lui Sha. Design and analysis of an mst-based topology control algorithm. In *INFOCOM*, 2003.

[11] M. V. Marathe, H. Breu, H. B. Hunt Iii, S. S. Ravi, and D. J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25:59–68, 1995.

[12] Kousha Moaveni-nejad and Xiang yang Li. Low-interference topology control for wireless ad hoc networks. In *ACM Wireless Networks*. IEEE Press, 2005.

[13] David Peleg. *Distributed computing: a locality-sensitive approach*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[14] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *Proc. of the 19 th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2000.

[15] V. Rodoplu and T. H. Meng. Minimum energy mobile wireless networks. *Selected Areas in Communications, IEEE Journal on*, 17(8):1333–1344, 1999.

[16] Xiaohua Jia Scott C. H. Huang, Peng-Jun Wan and Hongwei Du. Low-latency broadcast scheduling in ad hoc networks. *Lecture Notes in Computer Science*, 4138:527–538, 2006.

[17] H. Takagi and L. Kleinrock. Optimal transmission ranges for randomly distributed packet radio terminals. *Communications, IEEE Transactions on [legacy, pre - 1988]*, 32(3):246–257, 1984.

[18] Szu-Chi Wang, Chi-Yi Lin, and Sy-Yen Kuo. A localized topology control algorithm for constructing power efficient wireless ad hoc networks. In *World Wide Web Conference Series*, 2003.

[19] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi min Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. In *IEEE INFOCOM*, pages 1388–1397, 2001.

[20] Xiang yang Li and Peng jun Wan. Minimum energy mobile wireless networks revisited. In *In Proc. IEEE International Conference on Communications (ICC*, pages 278–283, 2001.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 879–884

ISBN 978-83-60810-27-9
ISSN 1896-7094

Fast Construction of Broadcast Scheduling and Gossiping in Dynamic Ad Hoc Networks

Krzysztof Krzywdziński
Faculty of Mathematics and Computer Science
Adam Mickiewicz University, 60–769 Poznań, Poland
Email: kkrzywd@amu.edu.pl

Abstract—**This paper studies the minimum latency broadcast schedule (MLBS) problem in ad hoc networks represented by unit disk graphs. In our approach we use an algorithm, which does not need BFS tree. We introduce a construction, which does not depend on a source, can be found in constant number of synchronous rounds, uses only short messages and produces broadcast schedule with latency at most 258 times optimum. The advantage of our construction over known algorithms is its ability to adapt fast to the changes in the network, such as adding, moving or deleting vertices (even during the broadcasting).**

We also study the minimum-latency gossiping (all-to-all broadcast) problem in unit disk graphs. Our algorithm is the best result for gossiping in unit disk graph in unbounded-size messages model.

Since our construction of broadcast scheduling does not depend on the source, it may be also used to solve other problems concerning broadcasting in unit disk graphs, such as single source multiple message broadcasting and multi channel broadcast scheduling.

I. Introduction

DUE TO a wide range of applications, such as military surveillance, emergency disaster relief or environmental monitoring, solving problems considering the communication in wireless ad hoc networks (such as sensor networks or cell phone networks) became an important issue. Generally the wireless ad hoc network is modeled by an undirected connected graph G, where the set of vertices $V(G)$ represents the set of computational units constituting the network (for example sensors, cellphones) and $E(G)$ contains unordered pairs of distinct vertices, such that $(v,w) \in E(G)$ if and only if the transmissions of the computational unit represented by the vertex v can directly reach the computational unit represented by the vertex w and vice versa (the reachability of transmissions is assumed to be a symmetric relation). For simplicity in the following considerations frequently we will say vertex v instead of the computational unit represented by the vertex v. Also if $(v,w) \in E(G)$ we will say that the vertices v and w are neighbours in G.

Usually it is assumed that communication in the wireless ad hoc network is synchronous and consists of a sequence of communication steps. Moreover in a wireless ad hoc network, a message transmitted by a vertex is always sent to all of its neighbours. In each step, a vertex v either transmits or listens. If v transmits, then the transmitted message reaches each of its neighbours by the end of the step. However, a vertex w adjacent to v successfully receives this message if and only if in this step w is listening and v is the only transmitting vertex among the neighbours of w. If vertex w is a neigbour of more than one transmitting vertex, then due to the interference, w does not receive any message in this step.

The classical problem of the information dissemination in wireless networks is the broadcasting problem. In the broadcasting problem the aim is to disseminate a particular message from a distinguished source vertex to all other vertices in the network. Due to the interference threat the brute force algorithm may not be the effective tool to solve this problem. The classical approach is to construct the broadcast schedule in order to minimize the adverse influence of interferences. The broadcast schedule of latency l is the sequence of subsets $(U_1, U_2 \dots U_l)$ of $V(G)$ satisfying

(1) $U_1 = \{s\}$;
(2) $U_i \subseteq \bigcup_{j=1}^{i-1} Inf(U_j)$ for each $2 \leq i \leq l$;
(3) $V(G) \setminus \{s\} \subseteq \bigcup_{j=1}^{l} Inf(U_j)$,

where for any $U \subseteq V$, $Inf(U)$ is the set of vertices in $V(G) \setminus U$ each of which has exactly one neighbour in U. By the definition we have that, given the broadcast schedule $(U_1, U_2 \dots U_l)$, we may disseminate information from s into the whole network in l synchronous rounds. Namely, in order to do so, in the i-th round the information should be transmitted only by vertices from U_i.

In this paper we consider the problem of minimum latency broadcast schedule (MLBS), i.e., the problem of minimization of the time needed to complete the task of dissemination of the information from a single source. Therefore we seek a fast algorithm which constructs the broadcast schedule with the minimum latency.

For general graphs the best approximation algorithms construct the broadcast schedules with latency: $O(R\Delta)$ [14], $O(R\log^2(n/R))$ [13], $O(R\log n + \log^2 n)$ [4], $R + O(\sqrt{R}\log^2 n)$ [18], $O(R + \log^6 n)$ [15], $R + O(\log^3 n)$ [17], $R + O(\log^3 n / \log\log n)$[6] and $O(R + \log^2 n)$ [16].

Here by R we mean the radius of the graph (the maximum distance between the source s and the vertices from $V(G)$) and $n = |V(G)|$.

The MLBS problem in unit disk graphs has been considered in [8], [5],[7],[20] and [11].

In [8] Gąsieniec, Kowalski, Lingas and Wahlen show that $R + \Omega(\log(n-R))$ rounds are required to accomplish broadcasting in UDG. In [5] Dessmark and Pelc presented a broadcast schedule of latency at most $2400R$. In [7] Gandhi, Parthasarathy and Mishra claim the NP-hardness of MLBS in unit disk graphs and construct broadcast schedule of latency at most $648R$. In [20] Scott, Huang, Wan, Jia and Du improve latency to $51R$, $24R$, and $R+O(\sqrt{R}\log^{1.5} R)$. In [11] Huang, Wan, Jia, Du and Shang propose three different algorithms which produce broadcast schedules with latency at most $24R+23$, $16R+15$ and $R+O(\log R)$. However implementation of these algorithms in distributed model depend on the source and is based on BFS algorithm, therefore using these algorithms $\Omega(R)$ rounds to build the broadcast structure are needed. This is not efficient enough in many applications.

Here we present HexagonsBroadcasting, which is the first distributed algorithm solving MLBS problem without using BFS tree. In fact HexagonsBroadcasting needs only $O(1)$ synchronous rounds to prepare the vertices for the transmission. Our algorithm has latency $258R$. The structure constructed by HexagonsBroadcasting does not depend on the source, therefore the additional advantage of our algorithm is that the source can be easily changed or even chosen randomly. Previous algorithms do not take into account possible changes in network during propagation of information. If a vertex is delete, they fail to deliver a message to a part of graph, despite the fact that graph might be still connected. In our algorithm bad impact of the movement, addition or deletion of the vertex in the network during the broadcasting may be rapidly diminished (Subsection III-A).

Finally the HexagonsBroadcasting approach can be applied to solve real problems such as gossiping, multiple messages and multi channel broadcast schedule.

Our paper is organized as follows. In Section II we present basic definitions and facts. In Section III we introduce the auxiliary algorithms SelectBroadcastNodes and BroadcastSets and the main algorithm HexagonsBroadcasting. In Section IV we present the algorithm GossipingInUDG and in Section V other two important generalizations of MLBS are presented.

II. Preliminaries

The natural theoretical model for wireless ad hoc network is a unit disk graph (UDG). By definition, the set of vertices of UDG is a set of points on the plane and two vertices of UDG, v and w, are connected by an edge if and only if they are at the distance at most one in Euclidean norm ($\|v, w\| \leq 1$). UDG vertices represent computational units and edges represent communication links between them.

Our algorithms will work on unit disk graphs, in distributed, synchronous, message-passing model of computations. We assume that local clocks of computational units (vertices of UDG) can be synchronized i.e., we assume that we perform computations in rounds (model LOCAL defined in [19]). In each round a computational unit represented by a vertex of UDG can send, receive messages from its neighbours, and can perform some local computations. Moreover we assume that every computational unit is either equipped with the Global Positioning System (GPS), or knows own position on the plane by other sources.

In the algorithm HexagonsBroadcasting we will use a partition of UDG based on grid tiling of the plane into hexagons of side equal $1/2$ (see Figure 1). Since each vertex of UDG knows its position on the plane, it also knows the hexagon of the grid to which it belongs. We define spanning subgraph G' of G where $V(G') = V(G)$ and $vw \in E(G')$ if and only if v and w lie in the same hexagon. We define $Hex(G)$ to be the set of all connected components of its spanning subgraph G' (notice that each hexagon lies inside a circle of radius $\frac{1}{2}$ so each $K \in Hex(G)$ is a clique).

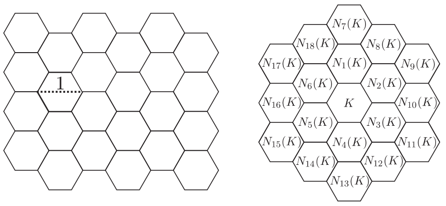
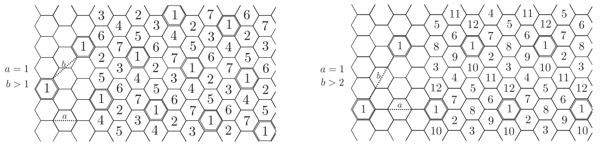


Fig. 1. Division into hexagons and numeration of the 18 neighbouring hexagons of the hexagon.

Let $K \in Hex(G)$ be any clique. Obviously K may have communication links only with vertices belonging to its 18 neighbouring hexagons (see Figure 1). We denote them by $N_1(K), N_2(K), \ldots, N_{18}(K)$ as on Figure 1 (the index of the clique depends on its position in the hexagon grid). Notice that, for example, $N_2(N_5(K)) = K$ and $N_{11}(N_{17}(K)) = K$. Therefore for any $k \in \{1, \ldots, 18\}$ we define a value $\bar{k}$ to be the number such that $N_{\bar{k}}(N_k(K)) = K$ (for example $\bar{5} = 2$, $\bar{1}7 = 11$). In addition we set

$$N(K) = \\ \{K' \in Hex(G) : K' \neq K, \exists_{v \in K, v' \in K'} vv' \in E(G)\}.$$

For comparison, by $N(v)$ we denote the set of neighbours of the vertex v.



$f^{(7)} : Hex(G) \to \mathbb{N}$ $f^{(12)} : Hex(G) \to \mathbb{N}$

Fig. 2. Concept of $f^{(7)}$ and $f^{(12)}$ functions.

Finally we define two functions $f^{(7)} : Hex(G) \to \{1, \ldots, 7\}$ and $f^{(12)} : Hex(G) \to \{1, \ldots, 12\}$ (see Figure 2). Notice that in the labeling with the numbers $\{1, \ldots, 7\}$ any two vertices from different hexagons with the same label are at distance greater than 1. Similarly in the labeling with the numbers $\{1, \ldots, 12\}$ any two vertices from different hexagons with the same label are at distance geater than 2. This implies two simple facts.

Fact 1. *Any two vertices from different cliques $K_1, K_2 \in Hex(G)$ such that $f^{(7)}(K_1) = f^{(7)}(K_2)$ are not connected by an edge in G.*

Fact 2. *Any two vertices from different cliques $K_1, K_2 \in Hex(G)$ such that $f^{(12)}(K_1) = f^{(12)}(K_2)$ do not have a common neighbour in G.*

III. Main Algorithm

The main algorithm selects a small number of vertices, which we call broadcast vertices. In fact the algorithm assigns to each vertex two sets $F_{out}(\cdot)$ and $F_{in}(\cdot)$ and *broadcast* vertices are those for which $F_{out}(\cdot)$ and $F_{in}(\cdot)$ are nonempty. The first function F_{out} is used when $K \in Hex(G)$ sends information to cliques from the neighbouring hexagons. The second one F_{in} is used in dissemination of the message inside the clique. The auxiliary algorithm SelectBroadcastNodes selects broadcast vertices.

```
SelectBroadcastNodes
Input: Unit disk graph $G$
Output: Functions $F_{in} : V(G) \to \mathcal{P}(\{1,\dots,18\})$, $F_{out} : V(G) \to \mathcal{P}(\{1,\dots,12\})$, where $\mathcal{P}(A)$ denotes the set of all subsets of $A$.
1) For every $v \in V(G)$ set $F_{in}(v) \equiv \emptyset$ and $F_{out}(v) \equiv \emptyset$.
2) For $i = 1$ to 7 do : For $k = 1$ to 18 do
   For every $K \in Hex(G)$ such that $f^{(7)}(K) = i$ and $f^{(7)}(N_k(K)) > f^{(7)}(K)$ parallel do
     a) Let $V_k(K) = \{v \in V(N_k(K)) : N(v) \cap K \neq \emptyset\}$
     b) If $V_k(K) \neq \emptyset$ then
        i) Select a vertex $v_k \in V_k(K)$ and set
              $F_{in}(v_k) := F_{in}(v_k) \cup \{\bar{k}\};$
              $F_{out}(v_k) := F_{out}(v_k) \cup \{f^{(12)}(K)\}.$
        ii) Select a vertex $w \in N(v_k) \cap K$ and set
              $F_{in}(w) := F_{in}(w) \cup \{k\};$
              $F_{out}(w) := F_{out}(w) \cup \{f^{(12)}(N_k(K))\}.$
     c) We call $v_k$ and $w$ coupled broadcast vertices.
```

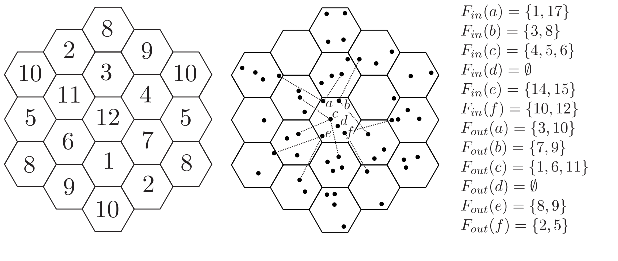


Fig. 3. Vertices from one hexagon with respective sets $F_{in}(\cdot)$ and $F_{out}(\cdot)$ and vertices from neighbouring hexagons coupled with them.

Notice that the algorithm SelectBroadcastNodes assigns exactly one pair of coupled broadcast vertices connected by an edge to each pair of the cliques $\{K, K'\}$ connected by at least one edge.

Basing on functions $F_{in} : V(G) \to \mathcal{P}(\{1,\dots,18\})$, $F_{out} : V(G) \to \mathcal{P}(\{1,\dots,12\})$ we construct sets $U_{out}(\cdot,\cdot)$ and $U_{in}(\cdot,\cdot)$ containing broadcast vertices, which will be used to construct the broadcast schedule.

The vertices in $U_{out}(i,k)$ send information from hexagons K with $f^{(12)} = i$ to hexagons K' with $f^{(12)}(K') = k$ and the vertices $U_{in}(i,1),\dots,U_{in}(i,18)$ are used in propagating information inside hexagons K with $f^{(7)} = i$.

```
BroadcastSets
Input: Unit disk graph $G$, Functions $F_{in} : V(G) \to \mathcal{P}(\{1,\dots,18\})$, $F_{out} : V(G) \to \mathcal{P}(\{1,\dots,12\})$
Output: Sets $U_{out}(i,k)$ and $U_{in}(i,k)$
1) Let $K_v$ denote clique $K \in Hex(G)$ such that $v \in K$.
2) For $i = 1$ to 12 : For $k = 1$ to 12 do
      $U_{out}(i,k) :=$
          $\Big\{v \in V : k \in F_{out}(v)$ and $f^{(12)}(K_v) = i\Big\}$
3) For $i = 1$ to 7 : For $k = 1$ to 18 do
      $U_{in}(i,k) :=$
          $\Big\{v \in V : k \in F_{in}(v)$ and $f^{(7)}(K_v) = i\Big\}$
```

Note that for all $1 \le t \le 12$ the set $U_{out}(t,t)$ is empty. Hence the number of construced nonempty sets is at most $11 \times 12 + 7 \times 18 = 258$ and in the main algorithm sets $U_{out}(t,t)$ may be omitted.

Lemma 1. *Let $1 \le i \le 12$, $1 \le k \le 12$ and $A \subseteq U_{out}(i,k)$. If $B \subseteq V(G)$ is the set of all vertices coupled with vertices from A and contained in cliques on which the value of $f^{(12)}$ equals k, then $B \subseteq Inf(A)$.*

Proof: Set $1 \le i \le 12$, $1 \le k \le 12$. Let $v_1 \dots v_t \in K$ be vertices from $U_{out}(i,k)$. For each vertex v_j the set $F_{out}(v_j)$ contains k if and only if v_j is coupled with at least one vertex from a neighbouring clique K' such that $f^{(12)}(K') = k$. Denote by W the set of vertices coupled with vertices from $\{v_1 \dots v_t\}$. Each vertex from W is contained in distinct clique (since to each pair of cliques at most one coupled pair of broadcast vertices is attributed). Combining Fact 2 with observation that on those distinct cliques the value of the function $f^{(12)}$ is k we have that no pair of vertices from the set W has a common neighbour in G, in particular no common neigbour in $\{v_1 \dots v_t\}$. Therefore, if $v_1 \dots v_t$ send information simultaneously into the network, then the vertices coupled with them (i.e vertices from W) receive it without interference. Now consider the vertices k_1, k_2 from $U_{out}(i,k)$ contained in two cliques K_1, K_2 (recall that, by the definition of U_{out}, $f^{(12)}(K_1) = f^{(12)}(K_2) = i$). By the Fact 2 vertices k_1 and k_2 do not have a common neighbour. Concluding, if vertices from the set $U_{out}(i,k)$ send information into the network, then the vertices coupled with them must get it without interference. ■

Lemma 2. *Let $1 \le i \le 7$, $1 \le k \le 18$. Suppose that vertices from the set $A \subseteq U_{in}(i,k)$ send information into the network* *simultaneously. Then for every $K \in Hex(G)$ such that $K \cap A \neq \emptyset$ we have that $K \subseteq Inf(A)$.*

Proof: Set $1 \leq i \leq 7$, $1 \leq k \leq 18$. Note that, for each broadcast vertex v, the set $F_{in}(v)$ contains k if and only if v is coupled with a vertex w from $N_k(K)$. So the set $V(K \cap U_{in}(i,k))$ has at most one element. Moreover any two distinct verices $v, v' \in U_{in}(i,k)$ are contained in cliques on which the function $f^{(7)}$ has value i. Therefore, from Fact 1 if any vertices from $U_{in}(i,k)$ sends simultaneously information into the network, then the vertices from cliques in which those vertices are contained, receive information without interference. ■

Now we are ready to introduce the main algorithm of propagating the information from the source $s \in V(G)$.

```
HEXAGONSBROADCASTING
Input: Unit disk graph $G$ and a vertex $s \in V(G)$
Output: Dissemiantion of the information from the source vertex $s$ to $V(G)$.
1) Run SELECTBROADCASTNODES algorithm on a graph $G$.
2) Run BROADCASTSETS algorithm on $G$.
3) Vertex $s$ transmits information to all his neighbours.
4) For $t = 1$ to $R$ do
    a) For $i = 1$ to 12; For $k = 1$ to 12; $k \neq i$ do
        - vertices from $U_{out}(i,k)$ which have received information transmit it in parallel
    b) For $i = 1$ to 7 : For $k = 1$ to 18 do
        - vertices from $U_{in}(i,k)$ which have received information transmit it in parallel
```

Such approach to the problem enables us to construct a broadcast schedule in constant number of rounds, since the algorithm SELECTBROADCASTNODES obviously takes constant number of synchronous rounds and also building sets U_{out} and U_{in} takes constant time.

Theorem 3. *Let R be the radius of unit disk graph G. Latency of the broadcast schedule $U_1, U_2, U_3, \ldots$ used by the algorithm* HEXAGONSBROADCASTING *is at most $258R + 1$.*

Proof: Let $U_1, U_2, U_3, \ldots$ be broadcast schedule used by HEXAGONSBROADCASTING algorithm and $K \in Hex(G)$ be such that $s \in K$. In the first step the vertex s sends information to all the vertices from K (i.e. $K \subseteq Inf(U_1)$). Then, by Lemma 1, in the next $11 \times 12 = 132$ steps broadcast vertices contained in K send information to all vertices coupled with them in cliques from $N(K)$. By Lemma 2, in the next 7×18 steps broadcast vertices propagate information inside the cliques from $N(K)$. Therefore, surely, $\bigcup_{l=1}^{258+1} Inf(U_l)$ contains all vertices from K and all vertices from cliques $N(K)$. Therefore all neighbours of vertex s in G receive information in the first $258 + 1$ steps. Now, we may repeat the reasoning replacing the broadcast vertices from clique K by broadcast vertices contained in cliques from $N(K)$. Therefore in the next 258 steps all vertices at distance 2 from s will receive the message (i.e. all vertices at distance two from s are contained in $\bigcup_{l=1}^{2\times 258+1} Inf(U_l)$). So by the same reasoning we get that all vertices at distance at most R from s are contained in $\bigcup_{l=1}^{258R+1} Inf(U_l)$. ■

A. Modification of the network during the broadcasting

The presented algorithm can be adjusted to construct the broadcast scheduling in the dynamic network in which appearance or deletion of vertices is possible. More precisely, we assume that at any time one of the following events may happen:

1) A vertex v, such that $G \cup \{v\}$ is connected, can be added.
2) A vertex $v \in V(G) \setminus \{s\}$, such that $G \setminus \{v\}$ is connected, can be deleted.
3) A vertex $v \in V(G)$ can change position on the plane (can be moved), if after this operation G is still connected.

Let s be a source and a be some constant. The following broadcasting algorithm broadcast information in the dynamic network.

```
DYNAMICBROADCASTING
1) Run HEXAGONSBROADCASTING and break step 4 of HEXAGONSBROADCASTING after $a$ iterations.
2) Go to step 1.
```

Since procedures SELECTBROADCASTNODES and BROADCASTSETS takes one round then using those algorithms it is possible to refresh rapidly the broadcast structure using the algorithm DYNAMICBROADCASTING. Therefore the algorithm DYNAMICBROADCASTING is an efficient tool to propagate information in dynamic network. If in some round the modifications (add, delete and move events) stop, then in the next $258R + (1/a)R + O(a)$ rounds all vertices receive information from the source. In reality the information will reach all the vertices even faster and it is possible, that the information will reach all vertices even if the modifications happen all the time.

In all known distributed algorithms (which are based on BFS tree) building the broadcast structure takes $O(R)$ time. Thus the analogous modification of those algorithm will not give comparable results. Namely, even if in some synchronous round the modifications (add, delete and move events) stop, then it would take at least $O(R^2/a + a)$ time to propagate the information from the source.

IV. GOSSIPING

In the gossiping problem each vertex in the network is initially given a message and the objective is to design a minimum-latency schedule such that each vertex distributes its message to all other vertices. We assume that during distribution of the messages interferences may occur.

There are two important models of the problem regarding whether or not we can combine two or more messages as a single message: unit-size and unbounded-size models. In unit-size trivial lower bound for latency is $V(G) + D - 1$ (D is a diameter of the graph G) and in unbounded-size model lower bound is $\Delta(G) + D - 1$ ($\Delta(G)$ is a maximum degree of a graph G).

For general graphs gossiping problem in unbounded-size model was investigated in: [1], [2], [3], [9], [10]. The known algorithms construct broadcast schedules which are approximations of the optimal one with approximation factor larger than any constant. For unit disk graphs in [7] the authors present constant approximation for gossiping. Although in the paper the exact ratio of approximation was not given it may be estimated to be at least $1944(D+|V(G)|)$. The best known algorithm for unit-size messages gossiping in unit disk graph is given in [12]. The authors present algorithm for gossiping with latency $27(|V(G)|+D-1)$. We should add that all known algorithms are based on the BFS tree therefore the time to build gossiping schedule is $\Omega(D)$.

We present here the algorithm for gossiping in unit disk graph in unbounded-size messages model which have latency $7\Delta + 258D$. The algorithm GOSSIPINGINUDG builds gossiping structure in constant time thus it can be adapted to become insensitive to adding, deleting or moving vertices (see Subsection III-A).

```
GOSSIPINGINUDG
1) Set function $F_{out}$ and $F_{in}$ using algorithm SELECT-BROADCASTNODES.
2) For each $K \in Hex(G)$ and vertices $v_1, v_2, v_3 \ldots = V(K)$ define $c(v_i) = 7i + f^{(7)}(K)$ for all $1 \leq i \leq |V(K)|$.
3) Let $v$ be a broadcast vertex. Define
       $c_2(v) = \max\{c(w) : \exists_u u \in N(v) \text{ and } w \in N(u)\}$
   (maximal value of the function $c$ in 2-neighbourhood of $v$).
4) Set all broadcast vertices (vertices which have $F_{out} \neq \emptyset$ and $F_{in} \neq \emptyset$ ) as inactive.
5) Parallel do:
   a) Each inactive vertex $v$ send his initial message in $c(v)$-th synchronous round.
   b) Every broadcast vertex $v'$ is activated in $c_2(v') + 1$ synchronous round and then sends all combined messages in rounds defined by point 4 of HEXAGONSBROADCASTING.
```

Theorem 4. *The algorithm* GOSSIPINGINUDG *delivers all messages to all vertices in* $7\Delta + 258D$ *synchronous rounds.*

Proof: Let $v \in V(K)$ and $v' \in V(K')$ be two distinct vertices such that $c(v) = c(v')$. From definition of the function c we have $f^{(7)}(K) = f^{(7)}(K')$ and $K \neq K'$. Therefore it follows from Fact 1 that in Step 5a no vertex from K will receive the initial message from v' while all vertices from K will receive the initial message from v. In step 5b broadcast vertices send messages after all vertices from their 2-neighbourhood (vertices at distance 2) have sent the initial message. Therefore there is no interference produced by broadcast vertices and the vertices sending the initial messages.

We use at most 7Δ synchronous rounds to send initial messages in Step 5a (there are 7 classes of hexagons and vertices from each hexagon form a clique) The number of rounds needed to complete 5b equals to the lattency of the schedule constructed by HEXAGONSBROADCASTING. Namely after the vertices from 2-neighbourhood have sent their initial message, all broadcast vertices send information according to HEXAGONSBROADCASTING. Properties of this broadcasting follow from Theorem 3. Therefore 5b takes at most $258D$ synchronous rounds.

Concluding the algorithm GOSSIPINGINUDG have latency at most $7\Delta + 258D$. ■

V. GENERALIZATIONS OF MLBS

A. Single Source Multiple Messages

Ghandi et al. in [7] introduce the problem of the single sources multiple messages broadcasting. They assume that the source may send multiple messages in different intervals of time and define latency as number of synchronous rounds in which all messages are received by all vertices. Let M be the number of the messages to be sent. In [7] latency is at least $194(M-1) + 7128(R-2)$.

We present algorithm which is a modification of the HEXAGONSBROADCASTING algorithm and construct Single Source Multiple Messages Broadcasting schedule of latency $518M+258R$. We only have to use the following modification of the points 3 and 4 of the algorithm HEXAGONSBROADCASTING.

```
3. Repeat $M + \lceil R/2 \rceil$ times
(a) If $s$ contains a new message, then send it
(b) Repeat two times
    – For $i = 1$ to 12; For $k = 1$ to 12; $k \neq i$ do
        - vertices from $U_{out}(i,k)$, which have received information and have not sent it before, send the information
    – For $i = 1$ to 7 : For $k = 1$ to 18 do
        - vertices from $U_{in}(i,k)$, which have received information and have not sent it before, send the information
```

Theorem 5. *If in the modified algorithm* HEXAGONSBROADCASTING *the source sends messages every* 259 *synchronous rounds then every vertex in network receive all the messages in at most* $518M + 258R$ *rounds, where* M *is the number of messages.*

Proof: Divide the broadcast schedule used by HEXAGONSBROADCASTING (omitting the first round) into intervals of 258 rounds. Notice that, if the broadcast vertex receive information in the t-th interval, then in the $t+1$-th interval the information is sent to all its neighbours. Therefore this vertex may send a new information in the $t+2$-th interval. The lack of bad impact of the interference follows from the proof of Theorem 3. Thus the gap of two intervals plus one round (to give time for s to send the message) is enough to broadcast several messages. ■

B. Multi Channel Broadcast Scheduling

In some applications the vertices can use n different channels and if two neighbours of the vertex v send information using different channels then vertex v receives it without interference. This is idea of multi channel broadcast scheduling. To the best of our knowledge this problem is considered for the first time.

Notice that in the previous considerations, if for any pair of indices $(i_1, k_1) \neq (i_2, k_2)$, vertices from sets $U_{out}(i_1, k_1)$ and $U_{out}(i_2, k_2)$ use different channels, then they may send information simultaneously in all the above mentioned algorithms without interference. The same thing concerns the sets $U_{in}(i_1, k_1)$ and $U_{in}(i_2, k_2)$ $((i_1, k_1) \neq (i_2, k_2))$. Therefore, if we use 132 channels ($11 \times 12 \leq 132$, $7 \times 18 \leq 132$) and attribute to each set of type U_{out} different channel and the same thing with the sets U_{in}, then we may rewrite all the presented algorithms and reduce significantly the time needed to disseminate information. If we define the latency in the analogous manner as in the classical definition, then for example the multi channel broadcast schedule used in the algorithm HEXAGONSBROADCASTING has latency $2R$.

VI. CONCLUSION

In this paper we have studied the MLBS problem in ad hoc networks which are represented by unit disk graphs. We have introduced a construction which produces broadcast schedule with latency at most 258 times optimum. The advantage of our construction over known algorithms is its ability to adapt fast to the changes in the network. We have also studied the minimum-latency gossiping, single source multiple message broadcasting and multi channel broadcast scheduling problem in unit disk graphs.

REFERENCES

[1] M. Chrobak, L. Gasieniec, and W. Rytter. Fast broadcasting and gossiping in radio networks. In *FOCS '00: Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, page 575, Washington, DC, USA, 2000. IEEE Computer Society.

[2] Marek Chrobak, Leszek Gasieniec, and Wojciech Rytter. A randomized algorithm for gossiping in radio networks. In *COCOON '01: Proceedings of the 7th Annual International Conference on Computing and Combinatorics*, pages 483–492, London, UK, 2001. Springer-Verlag.

[3] Marek Chrobak, Leszek Gasieniec, and Wojciech Rytter. Fast broadcasting and gossiping in radio networks. *J. Algorithms*, 43(2):177–189, 2002.

[4] A. D. Kowalski, A. Pelc. Centralized deterministic broadcasting in undirected multi-hop radio network. *In Random-Approx*, page 171182, 2004.

[5] Anders Dessmark and Andrzej Pelc. Tradeoffs between knowledge and time of communication in geometric radio networks. In *SPAA '01: Proceedings of the thirteenth annual ACM symposium on Parallel algorithms and architectures*, pages 59–66, New York, NY, USA, 2001. ACM.

[6] F. Manne F. Cicalese and Q. Xin. Faster centralized communication in radio networks. *LNCS*, 4288:339–348, 2006.

[7] Rajiv Gandhi, Srinivasan Parthasarathy, and Arunesh Mishra. Minimizing broadcast latency and redundancy in ad hoc networks. In *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 222–232, New York, NY, USA, 2003. ACM.

[8] Leszek Gąsieniec, Dariusz R. Kowalski, Andrzej Lingas, and Martin Wahlen. Efficient broadcasting in known geometric radio networks with non-uniform ranges. In *DISC '08: Proceedings of the 22nd international symposium on Distributed Computing*, pages 274–288, Berlin, Heidelberg, 2008. Springer-Verlag.

[9] Leszek Gąsieniec and Andrzej Lingas. On adaptive deterministic gossiping in ad hoc radio networks. *Inf. Process. Lett.*, 83(2):89–93, 2002.

[10] Leszek Gąsieniec and Igor Potapov. Gossiping with unit messages in known radio networks. In *TCS '02: Proceedings of the IFIP 17th World Computer Congress - TC1 Stream / 2nd IFIP International Conference on Theoretical Computer Science*, pages 193–205, Deventer, The Netherlands, The Netherlands, 2002. Kluwer, B.V.

[11] Huang, P. J. Wan, X. Jia, H. Du, and W. Shang. Minimum-latency broadcast scheduling in wireless ad hoc networks. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 733–739, 2007.

[12] Scott C.-H. Huang, Hongwei Du, and E-K. Park. Minimum-latency gossiping in multi-hop wireless networks. In *MobiHoc '08: Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pages 323–330, New York, NY, USA, 2008. ACM.

[13] O. Weinstein I. Chlamtac. The wave expansion approach to broadcasting in multihop radio networks. *IEEE Trans. on Communications*, 39:26–433, 1991.

[14] S. Kutten I. Chlamtac. On broadcasting in radio networks - problem analysis and protocol design. *IEEE Transactions on Communications*, 33:12401246, 1985.

[15] Y. Mansour I. Gaber. Centralized broadcast in multihop radio networks. *Journal of Algorithms*, 46:120, 2003.

[16] D. Kowalski and A. Pelc. Optimal deterministic broadcasting in known topology radio networks. *Distributed Computing*, 19(3:185–195, 2007.

[17] D.Peleg L.Gasieniec and Q.Xin. Faster communication in known topology radio networks. In *Proceedings of The 24th Annual ACM Symposium on Principles of Distributed Computing*, 2005.

[18] G. Kortsarz M. Elkin. Improved broadcast schedule for radio networks. In *Symposium on Discrete Algorithms (SODA)*, 2005, 222-231.

[19] David Peleg. *Distributed computing: a locality-sensitive approach*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[20] Xiaohua Jia Scott C. H. Huang, Peng-Jun Wan and Hongwei Du. Low-latency broadcast scheduling in ad hoc networks. *Lecture Notes in Computer Science*, 4138:527–538, 2006.

Workshop on Computational Optimization

Many 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 and multimodal optimization
- dynamic and noisy optimization
- large scale optimization
- parallel and distributed approaches in optimization
- random search algorithms, simulated annealing, tabu search and other derivative free optimization methods
- interval 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
- memetic algorithms
- optimization methods for learning processes and data mining
- computational optimization methods in statistics, econometrics, finance, physics, medicine, biology, engineering etc.

Program Committee

Le Thi Hoai An, Paul Verlaine University, Metz, France

Dirk Arnold, Dalhousie University, Canada

Vasile Berinde, North University of Baia Mare, Romania

Janez Brest, University of Maribor, Slovenia

Roberto Cominetti, University of Chile, Chile

Biswa Nath Datta, Northern Illinois University, USA

Andries Engelbrecht, University of Pretoria, South Africa

Frederic Guinand, Le Havre University, France

Hiroshi Hosobe, National Institute of Informatics, Japan

Olgierd Hryniewicz, Polish Academy of Sciences, Poland

Hideaki Iiduka, Kyushu Institute of Technology, Japan

Hubert Jongen, University of Aachen, Germany

Joaquim Judice, University of Coimbra, Portugal

Michal Kocvara, University of Birmingham, United Kingdom

Igor Konnov, Kazan University, RUSSIA

Jaromir Kukal, Czech Technical University, Prague, Czech Republic

Jouni Lampinen, University of Vaasa, Finland

Jose Mario Martinez, State University of Campinas, BRASIL

Juan Enrique Martinez Legaz, Universitat Autònoma de Barcelona, Spain

Stefan Maruster, West University of Timisoara, Romania

Radomil Matousek, University of Technology, Brno, Czech Republic

Kaisa Miettinen, University of Jyvaskylan, Finland

Zbigniew Nahorski, Warsaw School of Information Technology, Poland

Ferrante Neri, University of Jyvaskylan, Finland

Panos Pardalos, University of Florida, USA

Kalin Penev, Southampton Solent University, United Kingdom

Petr Posik, Czech Technical University, Czech Republic

Kenneth Price, USA

Nick Sahinidis, Carnegie Melon University, USA

Klaus Schittkowski, University of Bayreuth, Germany

Patrick Siarry, Universite Paris XII Val de Marne, France

Krzysztof Sikorski, USA

Mikhail Solodov, Instituto de Matematica Pura e Aplicada, BRASIL

Stefan Stefanov, Neofit Rilski University, Bulgaria

Tomas Stuetzle, Universite Libre de Bruxelles, Belgium

Ponnuthurai Nagaratnam Suganthan, Nanyang Technological University, Singapore

Pham Dinh Tao, Institut National des Sciences Apliquees de Rouen , France

Michael Vrahatis, University of Patras, Greece

Antanas Zilinskas, Research Institute of Mathematics and Informatics, Lithuania

Organizing (Steering) Committee

Stefka Fidanova, Academy of Sciences, Bulgaria

Josef Tvrdik, University of Ostrava, Czech Republic

Daniela Zaharie, West University of Timisoara, Romania

ISBN 978-83-60810-27-9
ISSN 1896-7094

ACO with semi-random start applied on MKP

Stefka Fidanova
and Pencho Marinov
Institute for Parallel Processing
Bulgarian Academy of Sciences
Acad. G. Bonchev str. bl 25A
1113 Sofia, Bulgaria
Email: {stefka, pencho }@parallel.bas.bg

Krassimir Atanassov
Central Laboratory for Bio-Medical Engineering
Bulgarian Academy of Sciences
Acad. G. Bonchev str. bl 25A
1113 Sofia, Bulgaria
Email: krat@bas.bg

***Abstract*—Ant Colony Optimization (ACO) is a stochastic search method that mimics the social behavior of real ants colonies, which manage to establish the shortest route to feeding sources and back. Such algorithms have been developed to arrive at near-optimal solutions to large-scale optimization problems, for which traditional mathematical techniques may fail. On this paper semi-random start is applied. A new kind of estimation of start nodes of the ants is made and several start strategies are prepared and combined. The idea of semi-random start is better management of the ants. This new technique is tested on Multiple Knapsack Problem (MKP). Benchmark comparison among the strategies is presented in terms of quality of the results. Based on this comparison analysis, the performance of the algorithm is discussed. The study presents ideas that should be beneficial to both practitioners and researchers involved in solving optimization problems.**

I. Introduction

Many combinatorial optimization problems are fundamentally hard. This is the most typical scenario when it comes to realistic and relevant problems in industry and science. Examples of optimization problems are Traveling Salesman Problem [10], Vehicle Routing [12], Minimum Spanning Tree [8], Multiple Knapsack Problem [5], etc. They are NP-hard problems and in order to obtain solution close to the optimality in reasonable time, metaheuristic methods are used. One of them is Ant Colony Optimization (ACO) [3].

ACO algorithms have been inspired by the real ants behavior. In the nature, ants usually wander randomly, and upon finding food return to their nest while laying down pheromone trails. If other ants find such a path, they are likely not to keep traveling at random, but to instead follow the trail, returning and reinforcing it if they eventually find food. However, as time passes, the pheromone starts to evaporate. The more time it takes for an ant to travel down the path and back again, the more time the pheromone has to evaporate and the path to become less prominent. A shorter path, in comparison will be visited by more ants and thus the pheromone density remains high for a longer time.

ACO is implemented as a team of intelligent agents which simulate the ants behavior, walking around the graph representing the problem to solve using mechanisms of cooperation and adaptation. ACO algorithm requires to define the following [1], [4]:

- The problem needs to be represented appropriately, which would allow the ants to incrementally update the solutions through the use of a probabilistic transition rules, based on the amount of pheromone in the trail and other problem specific knowledge. It is also important to enforce a strategy to construct only valid solutions corresponding to the problem definition.
- A problem-dependent heuristic function, that measures the quality of components that can be added to the current partial solution.
- A rule set for pheromone updating, which specifies how to modify the pheromone value.
- A probabilistic transition rule based on the value of the heuristic function and the pheromone value, that is used to iteratively construct a solution.

The structure of the ACO algorithm is shown by the pseudocode below (Figure 1). The transition probability $p_{i,j}$, to choose the node j when the current node is i, is based on the heuristic information $\eta_{i,j}$ and the pheromone trail level $\tau_{i,j}$ of the move, where $i, j = 1, \ldots, n$.

$$p_{i,j} = \frac{\tau_{i,j}^a \eta_{i,j}^b}{\sum_{k \in allowed} \tau_{i,k}^a \eta_{i,k}^b}, \tag{1}$$

The higher the value of the pheromone and the heuristic information, the more profitable it is to select this move and resume the search. In the beginning, the initial pheromone level is set to a small positive constant value τ_0; later, the ants update this value after completing the construction stage. ACO algorithms adopt different criteria to update the pheromone level.

The pheromone trail update rule is given by:

$$\tau_{i,j} \leftarrow \rho \tau_{i,j} + \Delta \tau_{i,j}, \tag{2}$$

where ρ models evaporation in the nature and $\Delta \tau_{i,j}$ is new added pheromone which is proportional to the quality of the solution.

Our novelty is to use estimations of start nodes with respect to the quality of the solution and thus to better manage the search process. On the basis of the estimations we offer several start strategies and their combinations. Like a benchmark problem is used Multiple Knapsack Problem (MKP) because a

978-83-60810-27-9/09/$25.00 © 2010 IEEE

```
Ant Colony Optimization
Initialize number of ants;
Initialize the ACO parameters;
while not end-condition do
    for k=0 to number of ants
        ant k choses start node;
        while solution is not constructed do
            ant k selects higher probability node;
        end while
    end for
    Update-pheromone-trails;
end while
```

Fig. 1. Pseudocode for ACO

lot of real world problems can be represented by it and MKP arises like a subproblem in many optimization problems.

The rest of the paper is organized as follows: in section 2 several start strategies are proposed. In section 3 the MKP is introduced. In section 4 the strategies are applied on MKP and the achieved results are compared and strategies are classified. At the end some conclusions and directions for future work are done.

II. Start Strategies

The known ACO algorithms create a solution starting from random node. But for some problems, especially subset problems, it is important from which node the search process starts. For example if an ant starts from node which does not belong to the optimal solution, probability to construct it is zero. Therefore we offer several start strategies.

Let the graph of the problem has m nodes. We divide the set of nodes on N subsets. There are different ways for dividing. Normally, the nodes of the graph are randomly enumerated. An example for creating of the nodes subsets, without loss of generality, is: the node number one is in the first subset, the node number two is in the second subset, etc. the node number N is in the $N-th$ subset, the node number $N+1$ is in the first subset, etc. Thus the number of the nodes in the subsets are almost equal. We introduce estimations $D_j(i)$ and $E_j(i)$ of the node subsets, where $i \geq 2$ is the number of the current iteration. $D_j(i)$ shows how good is the j^{th} subset and $E_j(i)$ shows how bad is the j^{th} subset. $D_j(i)$ and $E_j(i)$ are weight coefficients of $j-th$ node subset $(1 \leq j \leq N)$, which we calculate by the following formulas:

$$D_j(i) = \phi.D_j(i-1) + (1-\phi).F_j(i), \tag{3}$$

$$E_j(i) = \phi.E_j(i-1) + (1-\phi).G_j(i), \tag{4}$$

where $i \geq 1$ is the current process iteration and for each j $(1 \leq j \leq N)$:

$$F_j(i) = \begin{cases} \frac{f_{j,A}}{n_j} & \text{if } n_j \neq 0 \\ F_j(i-1) & \text{otherwise} \end{cases}, \tag{5}$$

$$G_j(i) = \begin{cases} \frac{g_{j,B}}{n_j} & \text{if } n_j \neq 0 \\ G_j(i-1) & \text{otherwise} \end{cases}, \tag{6}$$

$f_{j,A}$ is the number of the solutions among the best $A\%$, $g_{j,B}$ is the number of the solutions among the worst $B\%$, where $A + B \leq 100$, $i \geq 2$ and

$$\sum_{j=1}^{N} n_j = n, \tag{7}$$

where n_j $(1 \leq j \leq N)$ is the number of solutions obtained by ants starting from nodes subset j, n is the number of ants. Initial values of the weight coefficients are: $D_j(1) = 1$ and $E_j(1) = 0$. The parameter ϕ, $0 \leq \phi \leq 1$, shows the weight of the information from the previous iterations and from the last iteration. When $\phi = 0$ only the information from the last iteration is taken in to account. If $\phi = 0.5$ the influence of the previous iterations versus the last is equal. When $\phi = 1$ only the information from the previous iterations is taken in to account. When $\phi = 0.25$ the weight of the information from the previous iterations is three times less than this one of the last iteration. When $\phi = 0.75$ the weight of the previous iterations is three times higher than this one of the last iteration. The balance between the weights of the previous iterations and the last is important. At the beginning when the current best solution is far from the optimal one, some of the node subsets can be estimated as good. Therefore, if the value of the parameter ϕ is too high the estimation can be distorted. If the weight of the last iteration is too high then information for good and bad solutions from previous iterations is ignored, which can distort estimation too.

We try to use the experience of the ants from previous iteration to choose the better starting node. Other authors use this experience only by the pheromone, when the ants construct the solutions [4]. Let us fix threshold E for $E_j(i)$ and D for $D_j(i)$, than we construct several strategies to choose start node for every ant, the threshold E increases every iteration with $1/i$ where i is the number of the current iteration:

1 If $E_j(i)/D_j(i) > E$ then the subset j is forbidden for current iteration and we choose the starting node randomly from $\{j \ |j$ is not forbidden$\}$;

2 If $E_j(i)/D_j(i) > E$ then the subset j is forbidden for current simulation and we choose the starting node randomly from $\{j \ |j$ is not forbidden$\}$;

3 If $E_j(i)/D_j(i) > E$ then the subset j is forbidden for K_1 consecutive iterations and we choose the starting node randomly from $\{j \ |j$ is not forbidden$\}$;

4 Let $r_1 \in [0.5, 1)$ is a random number. Let $r_2 \in [0, 1]$ is a random number. If $r_2 > r_1$ we randomly choose node from subset $\{j \ |D_j(i) > D\}$, otherwise we randomly chose a node from the not forbidden subsets, r_1 is chosen and fixed at the beginning.

5 Let $r_1 \in [0.5, 1)$ is a random number. Let $r_2 \in [0, 1]$ is a random number. If $r_2 > r_1$ we randomly choose node

from subset $\{j \mid D_j(i) > D\}$, otherwise we randomly chose a node from the not forbidden subsets, r_1 is chosen at the beginning and increase with r_3 every iteration.

Where $0 \leq K_1 \leq$"number of iterations" is a parameter. If $K_1 = 0$, than strategy 3 is equal to the random choose of the start node. If $K_1 = 1$, than strategy 3 is equal to the strategy 1. If $K_1 =$"maximal number of iterations", than strategy 3 is equal to the strategy 2.

We can use more than one strategy for choosing the start node, but there are strategies which can not be combined. We distribute the strategies into two sets: $St1 = \{strategy1,\ strategy2,\ strategy3\}$ and $St2 = \{strategy5,\ strategy6\}$. The strategies from same set can not be used at once. Thus we can use strategy from one set or combine it with strategies from the other set. Exemplary combinations are $(strategy1)$, $(strategy2;\ strategy5)$, $(strategy3;\ strategy6)$. When we combine strategies from $St1$ and $St2$, first we apply the strategy from $St1$ and according it some of the regions (node subsets) become forbidden, and after that we choose the starting node from not forbidden subsets according the strategy from $St2$

III. Multiple Knapsack Problem

We test the ideas for controlled start on MKP. MKP is a real world problem and is a representative of the class of subset problems. The MKP has numerous applications in theory as well as in practice. It also arises as a subproblem in several algorithms for more complex problems and these algorithms will benefit from any improvement in the field of MKP. The following major applications can be mentioned: problems in cargo loading, cutting stock, bin-packing, budget control and financial management. Sinha and Zoltner [9] proposed to use the MKP in fault tolerance problem and in [2] is designed a public cryptography scheme whose security realize on the difficulty of solving the MKP. Martello and Toth [7] mention that two-processor scheduling problems may be solved as a MKP. Other applications are industrial management, naval, aerospace, computational complexity theory.

The MKP can be thought as a resource allocation problem, where there are m resources (the knapsacks) and n objects and every object j has a profit p_j. Each resource has its own budget c_j (knapsack capacity) and consumption r_{ij} of resource i by object j. The aim is maximizing the sum of the profits, while working with a limited budget.

The MKP can be formulated as follows:

$$\max \textstyle\sum_{j=1}^{n} p_j x_j$$

$$\text{subject to} \textstyle\sum_{j=1}^{n} r_{ij} x_j \leq c_i \quad i = 1, \dots, m \tag{8}$$

$$x_j \in \{0, 1\} \quad j = 1, \dots, n$$

x_j is 1 if the object j is chosen and 0 otherwise.

There are m constraints in this problem, so MKP is also called m-dimensional knapsack problem. Let $I = \{1, \dots, m\}$ and $J = \{1, \dots, n\}$, with $c_i \geq 0$ for all $i \in I$. A well-stated MKP assumes that $p_j > 0$ and $r_{ij} \leq c_i \leq \sum_{j=1}^{n} r_{ij}$ for all $i \in I$ and $j \in J$. Note that the $[r_{ij}]_{m \times n}$ matrix and $[c_i]_m$ vector are both non-negative.

In the MKP one is not interested in solutions giving a particular order. Therefore a partial solution is represented by $S = \{i_1, i_2, \dots, i_j\}$ and the most recent elements incorporated to S, i_j need not be involved in the process for selecting the next element. Moreover, solutions for ordering problems have a fixed length as one search for a permutation of a known number of elements. Solutions for MKP, however, do not have a fixed length. The graph of the problem is defined as follows: the nodes correspond to the items, the arcs fully connect nodes. Fully connected graph means that after the object i one can chooses the object j for every i and j if there are enough resources and object j is not chosen yet.

IV. Computational Results

The computational experience of the ACO algorithm is shown using 10 MKP instances from "OR-Library" available within WWW access at **http://people.brunel.ac.uk/mastjjb/jeb/orlib/**, with 100 objects and 10 constraints. To provide a fair comparison for the above implemented ACO algorithm, a predefined number of iterations, $k = 100$, is fixed for all the runs. Thus we can observe which strategy reaches good solutions faster. If the value of k (number of iterations) is too high, the achieved results will be very close to the optimal solution and will be difficult to appreciate different strategies. We apply strategies on MMAS [11], because it is one of the best ACO approach. The developed technique has been coded in C++ language and implemented on a Pentium 4 (2.8 Ghz). The parameters are fixed as follows: $\rho = 0.5$, $a = 1$, $b = 1$, number of used ants is 20, $A = 30$, $B = 30$, $D = 1.5$, $E = 0.5$, $K_1 = 5$, $r_3 = 0.01$. The values of ACO parameters (ρ, a, b) are from [6] and experimentally is found that they are best for MKP. The tests are run with 1, 2, 4, 5 and 10 nodes within the nodes subsets and values for ϕ are 0, 0.25, 0.5 and 0.75. For every experiment, the results are obtained by performing 30 independent runs, then averaging the fitness values. The computational time which takes start strategies is negligible with respect to the computational time which takes solution construction.

Tests with all possible combinations of strategies and with random start (12 combinations at all), four value for ϕ and five kind of node subsets are run and every test 30 times. Thus the all runs are 72 000. One can observe that sometimes all nodes subsets become forbidden and the algorithm stops before performing all iterations (strategies 1, 2, 3 and combinations with them). So if all nodes subsets become forbidden the algorithm performs several iterations without any strategy with random start till some of the subsets become not forbidden. Then the algorithm continue to apply the chosen strategy.

The problem which arises is how to compare the achieved solutions by different strategies and different node-devisions. Therefore the difference (interval) d between the worst and best average result for every problem is divided to 10. If the

TABLE I
ESTIMATON OF STRATEGIES AND NODES DEVISIONS FOR $\phi = 0$

number nodes	10	5	4	2	1
random	32	32	32	32	32
strat. 1	84	84	87	83	83
strat. 2	33	31	36	53	74
strat. 3	79	86	86	88	86
strat. 4	86	86	86	86	86
strat. 5	86	86	86	86	86
strat. 1-4	83	89	84	81	89
strat. 1-5	83	89	84	81	89
strat. 2-4	33	36	35	53	82
strat. 2-5	33	36	35	63	82
strat. 3-4	69	89	88	87	**90**
strat. 3-5	69	89	88	87	**90**

average result for some strategy, node devision and ϕ is in the first interval with borders the worst average result and worst average plus $d/10$ it is appreciated with 1. If it is in the second interval with borders the worst average plus $d/10$ and worst average plus $2d/10$ it is appreciated with 2 and so on. If it is in the 10th interval with borders the best average minus $d/10$ and the best average result, it is appreciated with 10. Thus for a test problem the achieved results for every strategy, every nodes devision and every ϕ is appreciated from 1 to 10. After that is summed the rate of all test problems for every strategy, every nodes devision and ϕ. So the rate of the strategies/node-devision/ϕ becomes between 10 and 100, because the benchmark problems are 10. It is mode of result classification.

On Table I is shown the rate of the strategies/node-devision when parameter $\phi = 0$, which means that only the achieved results from the last iteration are taken in to account in the node-subsets estimation, in bold is the best rate. We observe that the rate of the ACO algorithm with start strategies outperforms the traditional ACO with completely random start. Comparing the strategies, the worst rate has strategy 3 and their combinations with strategies 4 and 5. In strategy 3 the nodes-subsets with high value of estimation $E_j(i)$ become forbidden for current simulation. So if at the beginning iterations of the algorithm from some node-subset start only bad solutions it will be forbidden, but it is possible from this node subset to start good solutions too. The best rate have the combinations of strategies 1 and 3 with strategies 4 and 5. It means that it is better the node subsets which are appreciated like bad to be forbidden for a fixed number of iterations and it is better to forbid some node-subsets and to stimulate ants to start from other which looks to be good, than to apply only one strategy (forbidden or stimulated). The worst rate with respect of node devision is when there are 10 nodes in the node-subsets. When there

TABLE II
ESTIMATON OF STRATEGIES AND NODES DEVISIONS FOR $\phi = 0.25$

number nodes	10	5	4	2	1
random	32	32	32	32	32
strat. 1	83	88	86	90	90
strat. 2	32	31	36	61	81
strat. 3	62	86	84	84	96
strat. 4	86	86	86	86	86
strat. 5	86	86	86	86	86
strat. 1-4	84	91	87	92	96
strat. 1-5	84	91	87	92	96
strat. 2-4	34	33	35	59	85
strat. 2-5	34	33	35	59	85
strat. 3-4	69	83	86	84	**97**
strat. 3-5	69	83	86	84	**97**

TABLE III
ESTIMATION OF STRATEGIES AND NODES DEVISIONS FOR $\phi = 0.5$

number nodes	10	5	4	2	1
random	32	32	32	32	32
strat. 1	78	86	88	92	96
strat. 2	34	35	38	51	78
strat. 3	61	86	88	94	**97**
strat. 4	86	86	86	86	86
strat. 5	86	86	86	86	86
strat. 1-4	79	90	87	94	**97**
strat. 1-5	79	90	87	94	**97**
strat. 2-4	35	40	44	56	83
strat. 2-5	35	40	44	56	83
strat. 3-4	68	92	88	92	96
strat. 3-5	68	92	88	92	96

are to many nodes in the node subset then it is possible from this subset to start good and bad solutions and it is difficult to appreciate it. The best rate with respect to the node devision is when in the node subsets is only one node.

On Tables II, III and IV are shown the rate of the strategies/node-devision when parameter $\phi = 0.25, 0.5$ and 0.75. We can make similar to the $\phi = 0$ conclusions. For all values of the parameter ϕ the best rate according node devision is when there is only one node in node-subsets. So we put in Table V the rate of the start strategies when the node subsets consist one node, with bold is the best rate.

On Table V we observe that the worst rate according value of the parameter ϕ is when we take in to account only the achieved solutions from the last iteration ($\phi = 0$). The rate

TABLE IV
ESTIMATION OF STRATEGIES AND NODES DEVISIONS FOR $\phi = 0.75$

number nodes	10	5	4	2	1
random	32	32	32	32	32
strat. 1	71	81	85	89	92
strat. 2	35	55	52	60	87
strat. 3	56	76	88	95	95
strat. 4	86	86	86	86	86
strat. 5	86	86	86	86	86
strat. 1-4	67	83	89	94	95
strat. 1-5	67	83	89	94	95
strat. 2-4	39	47	48	58	85
strat. 2-5	39	47	48	58	85
strat. 3-4	56	81	87	94	**97**
strat. 3-5	56	81	87	94	**97**

TABLE V
ESTIMATION OF STRATEGIES AND PARAMETER ϕ

ϕ	0	0.25	0.5	0.75
random	32	32	32	32
strat. 1	83	93	96	92
strat. 2	74	81	78	87
strat. 3	86	96	**97**	95
strat. 4	86	86	86	86
strat. 5	86	86	86	86
strat. 1-4	89	96	**97**	95
strat. 1-5	89	96	**97**	95
strat. 2-4	82	85	83	85
strat. 2-5	82	85	83	85
strat. 3-4	90	**97**	96	**97**
strat. 3-5	90	**97**	96	**97**

of strategies when $\phi = 0.5$ is slightly better than rate when $\phi = 0.25$ and $\phi = 0.75$. So we can conclude that the balance between information from previous iterations and from last iteration is very important. According to the strategies, the worst rate is when is applied traditional ACO with random start and with strategy 2 when the subsets stay forbidden for current simulation. The best rate are combinations of strategy 3 with strategies 4 and 5. So for better performance of the ACO algorithm is advisable to forbid bad regions for several iterations and to stimulate ants to start construction of the solutions from good regions.

V. CONCLUSION

In this paper we address the modeling of the process of ant colony optimization method by using estimations, combining five start strategies. So, the start node of each ant depends of the goodness of the respective region. We focus on parameter settings which manage the starting procedure. We investigate on influence of the parameter ϕ to algorithm performance. The best solutions are achieved when ”bad” regions are forbidden for several iterations and the probability the ants to start from ”good” regions is higher. In future we will apply our modification of ACO algorithm on various classes of problems. We will investigate the influence of the estimations and start strategies on the achieved results.

ACKNOWLEDGMENT

This work has been partially supported by the Bulgarian National Scientific Fund under the grants Modeling Processes with fixed development rules - DID 02/29 and Effective Monte Carlo Methods for large-scale scientific problems - DTK 02/44.

REFERENCES

[1] E. Bonabeau, M. Dorigo and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, New York,Oxford University Press, 1999.

[2] W. Diffe and M.E. Hellman, *New direction in cryptography*, IEEE Trans Inf. Theory. IT-36,1976, 644-654.

[3] M. Dorigo and L.M. Gambardella, *Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem*, IEEE Transactions on Evolutionary Computation 1, 1997, 53-66.

[4] M. Dorigo and T. Stutzle, *Ant Colony Optimization*, MIT Press, 2004.

[5] S. Fidanova, *Evolutionary Algorithm for Multiple Knapsack Problem*, Int. Conference Parallel Problems Solving from Nature, Real World Optimization Using Evolutionary Computing, ISBN No 0-9543481-0-9,Granada, Spain, 2002.

[6] S. Fidanova, *Ant colony optimization and multiple knapsack problem*, in: Renard, J.Ph. (Eds.), Handbook of Research on Nature Inspired Computing for Economics ad Management, Idea Group Inc., ISBN 1-59140-984-5, 2006, 498-509.

[7] S. Martello and P. Toth, *A mixtures of dynamic programming and branch-and-bound for the subset-sum problem*, Management Science 30, 1984, 756-771.

[8] M. Reiman and M. Laumanns, *A Hybrid ACO algorithm for the Capacitate Minimum Spanning Tree Problem*, In proc. of First Int. Workshop on Hybrid Metahuristics, Valencia, Spain, 2004, 1-10.

[9] A. Sinha and A.A. Zoltner, *The multiple-choice knapsack problem*, J. Operational Research 27, 1979, 503-515.

[10] T. Stutzle and M. Dorigo, *ACO Algorithm for the Traveling Salesman Problem*, In K. Miettinen, M. Makela, P. Neittaanmaki, J. Periaux eds., Evolutionary Algorithms in Engineering and Computer Science, Wiley, 1999, 163-183.

[11] T. Stutzle and H.H. Hoos, *MAX-MIN Ant System*, In Dorigo M., Stutzle T., Di Caro G. (eds). Future Generation Computer Systems, Vol 16, 2000, 889–914.

[12] T. Zhang, S. Wang, W. Tian and Y. Zhang, ACO-VRPTWRV: A New Algorithm for the Vehicle Routing Problems with Time Windows and Re-used Vehicles based on Ant Colony Optimization, Sixth International Conference on Intelligent Systems Design and Applications, IEEE press, 2006, 390-395.

Proceedings of the International Multiconference on
Computer Science and Information Technology pp. 893–900

ISBN 978-83-60810-27-9
ISSN 1896-7094

On the PROBABILISTIC MIN SPANNING TREE problem (Extended abstract)

Nicolas Boria, Cécile Murat, Vangelis Th. Paschos
LAMSADE, CNRS and Université Paris-Dauphine, France
Email: {boria,murat,paschos}@lamsade.dauphine.fr

***Abstract*—We study a probabilistic optimization model for MIN SPANNING TREE, where any vertex v_i of the input-graph $G(V,E)$ has some presence probability p_i in the final instance $G' \subset G$ that will effectively be optimized. Supposing that when this "real" instance G' becomes known, a decision maker might have no time to perform computation from scratch, we assume that a spanning tree T, called *anticipatory* or *a priori* spanning tree, has already been computed in G and, also, that a decision maker can run a quick algorithm, called *modification strategy*, that modifies the anticipatory tree T in order to fit G'. The goal is to compute an anticipatory spanning tree of G such that, its modification for any $G' \subseteq G$ is optimal for G'. This is what we call PROBABILISTIC MIN SPANNING TREE problem. In this paper we study complexity and approximation of PROBABILISTIC MIN SPANNING TREE in complete graphs as well as of two natural subproblems of it, namely, the PROBABILISTIC METRIC MIN SPANNING TREE and the PROBABILISTIC MIN SPANNING TREE 1,2 that deal with metric complete graphs and complete graphs with edge-weights either 1, or 2, respectively.**

I. Introduction

The basic problematic of probabilistic combinatorial optimization (in graphs) is the following. We are given a graph $G(V,E)$ on which we have to solve some optimization problem Π. But, for some reasons depending on the reality modeled by G, Π is only going to be solved for some subgraph G' of G (determined by the vertices that will finally be present) rather than for the whole of G. The measure of how likely it is that a vertex $v_i \in V$ will belong to G' (i.e., will be present for the final optimization) is expressed by a probability p_i associated with v_i. How we can proceed in order to solve Π under this kind of uncertainty?

A first very natural idea that comes to mind is that one waits until G' is specified (i.e., it is present and ready for optimization) and, at this time, one solves Π in G'. This is what is called *reoptimization*.

But what if there remains very little time for such a computation? In this case, another way to proceed is the following. One solves Π in the whole of G in order to get a feasible solution (denoted by S), called *a priori* or *anticipatory solution*, which will serve her/him as a kind of benchmark for the solution on the effectively present subgraph G'. One has also to be provided with an algorithm that modifies S in order to fit G'. This algorithm is called *modification strategy* (let us denote it by $\mathtt{M}$). The objective now becomes to compute an anticipatory solution that, when modified by $\mathtt{M}$, remains "good" for any subgraph of G (if this subgraph is the one where Π will be finally solved).

This amounts to computing a solution that optimizes a kind of expectation of the value of the modification of S over all the possible subgraphs of G, i.e., the sum of the products of the probability that G' is the finally present graph multiplied by the value of the modification of S in order to fit G' over any subgraph G' of G. This expectation, depending on both the instance of the deterministic problem Π, the vertex-probabilities, and the modification strategy adopted, will be called the *functional*. Obviously, the presence-probability of G' is the probability that all of its vertices are present and the other vertices outside G' are absent.

Seen in this way, the probabilistic version PROBABILISTIC Π of a (deterministic) combinatorial optimization problem Π becomes another equally deterministic problem Π', the solutions of which have the same feasibility constraints as those of Π but with a different objective function where vertex-probabilities intervene.

In this sense, probabilistic combinatorial optimization is very close to what in the last couple of years has been called "one stage optimisation under independent decision models", an area very popular in the stochastic optimization community.

What are the main mathematical problems dealing with probabilistic consideration of a problem Π in the sense discussed above? One can identify at least five interesting mathematical and computational problems dealing with probabilistic combinatorial optimization:

1) write the functional down in an analytical closed form;
2) if such an expression of the functional is possible, prove that its value is polynomially computable (this amounts to proving that the modified problem Π' belongs to **NP**);
3) determine the complexity of the computation of the optimal *a priori* solution, i.e., of the solution optimizing the functional (in other words, determine the computational complexity of Π');
4) if Π' is **NP**-hard, study polynomial approximation issues;
5) always, under the hypothesis of the **NP**-hardness of Π', determine its complexity in the special cases where Π is polynomial, and in the case of **NP**-hardness, study approximation issues.

Let us note that, although curious, point 2 in the above list is neither trivial nor senseless. Simply consider that the summation for the functional includes, in a graph of order n, 2^n terms (one for each subgraph of G). So, polynomiality of the computation of the functional is, in general, not immediate.

978-83-60810-27-9/09/$25.00 © 2010 IEEE

Several optimization frameworks have been introduced by the operations research community for handling data uncertainty, the most well developed being *Stochastic programming* (see [1], [2] for basics) and *Robust discrete optimization* (see, for example, [3]).

The framework of *Probabilistic combinatorial optimization* where our work lies at, was introduced by [4], [5]. In [6], [5], [7], [8], [9], [4], [10], [11], [12], restricted versions of routing and network-design probabilistic minimization problems (in complete graphs) have been studied under the robustness model dealt here (called *a priori optimization*). In [13], the analysis of the probabilistic minimum travelling salesman problem, originally performed in [5], [4], has been revisited and refined.

Several other combinatorial problems have been recently treated in the probabilistic combinatorial optimization framework, including minimum coloring ([14], [15]), maximum independent set and minimum vertex cover ([16], [17]), longest path ([18]), Steiner tree problems ([19], [20]). Note also that probabilistic minimum spanning tree has also studied by [8] but under a very different probabilistic model.

We apply in this paper the probabilistic combinatorial optimization setting just described in the minimum spanning tree problem.

Given an edge-weighted graph $G(V, E)$, with positive edge weights $d : E \rightarrow \mathbb{Q}^+$, the minimum spanning tree problem (MIN SPANNING TREE) consists of determining a minimum total edge-weight tree spanning V.

MIN SPANNING TREE is a celebrated problem, very frequently modeling several kinds of networks in transports, communications, energy, logistics, etc. MIN SPANNING TREE has been actively studied under several optimization models like on-line computation, dynamic optimization, etc. Its study always motivates numerous researchers in theoretical computer science and in operational research.

In what follows, we first design a modification strategy and derive an analytic expression of the expected value (called *functional* in what follows) of a spanning tree of G, under this modification strategy.

We next show that the problem of *a priori optimization*, i.e., the problem of determining an anticipatory solution minimizing the functional, is **NP**-hard in general complete graphs (Section II).

Subsequently, we study complexity of the PROBABILISTIC MIN SPANNING TREE problem when dealing with particular cases of vertex-probabilities values and/or edge weights and particular cases of anticipatory solutions (Section IV).

We next derive polynomial-time approximation results for metric graphs and for graphs where edge-weights are either 1 or 2 (Section V). Because of the limits to the paper's size, some of the results are given without their proofs.

II. THE MODIFICATION STRATEGY, THE FUNCTIONAL ASSOCIATED WITH, AND THE COMPLEXITY OF PROBABILISTIC MIN SPANNING TREE

Consider a complete[1] weighted graph $G(V, E)$ on n vertices, with edge weights given by a function $d : E \rightarrow \mathbb{Q}^+$. Set $V = \{v_1, v_2, \ldots, v_n\}$. Each vertex $v_i \in V$, is associated with a presence probability $p_i \in \mathbb{Q}^+$ measuring, as already mentioned, how likely is that v_i will be present in the instance where PROBABILISTIC MIN SPANNING TREE will really be solved. We assume that subgraph $G'(V', E')$ of G materializes as the outcome of n independent Bernoulli trials, one per vertex $v_i \in V$: $v_i \in V'$ with probability p_i. Then, $E' = \{(u, v) \in E : u \in V' \text{ and } v \in V'\}$. Let us note that it seems to be natural that, for a fixed modification strategy M, given an anticipatory spanning tree T, some basic properties of its structure must be preserved in any tree T' built when M adjusts T to $G'(V', E')$, for any $V' \subseteq V$. Such a basic property is, for instance, the relation "predecessor-successor" in T. In order that this relation is preserved in any T', we assume that there exists a vertex, denoted by v_1 with $p_1 = 1$.

To motivate this assumption, consider a pacifist version of an application in [5]. Let G be a graph of order n and let its vertices represent researchers of a research network. The weight of an edge linking researcher i to researcher j quantifies the "inefficiency" risk incurred when i and j have to accomplish a common task. We wish to determine an organizational structure of this network where all the researchers implied accomplish some tasks and where the total "inefficiency" risk is minimized.

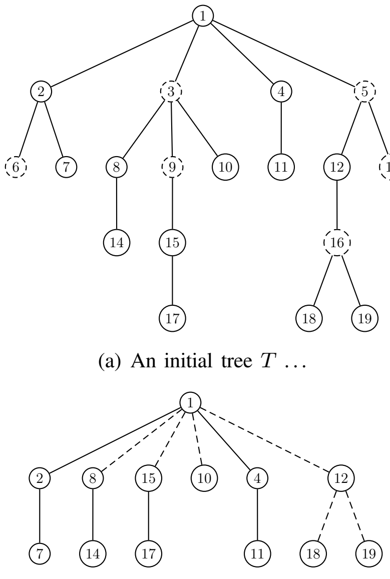
This can be modeled as a minimum spanning tree T^* of G. Let us now suppose that the project at hand involves several tasks where only a subset of researchers are implied, the project manager (the omnipresent vertex v_1) being involved to all of them. The probability associated with a researcher $v_i \neq v_1$, is the probability that v_i participates to an arbitrary task undertaken by the research network. Each such task will be represented by some subgraph G' of G and its "inefficiency" risk is the cost of a minimum spanning tree of G'. The modification strategy that is to be adopted for such a model must allow us to keep the same structure from a task to another one in order that the "inefficiency" risk remains as low as possible. Other applications from distributed systems also justify similar assumptions.

Formally, let $G(V, E)$ be a complete graph with $|V| = n$ and $(p_i)_{i=1,\ldots,n}$ a vertex-probability system with $p_1 = 1$ (in other words, vertex v_1 is assumed to be always present). Consider a tree T spanning V and number vertices in T in a left-to-right breadth-first-search (bfs) way. Consider a subgraph $G'(V', E') = G[V']$ of G induced by a set $V' \subseteq V$. The modification strategy (adjusting T to a spanning tree T' of G' and denoted by LEV in what follows) that will be analyzed in the sequel works as follows:

[1] The assumption that the input-graph is complete is made in order to ensure connectivity of the tree T' for any subgraph $G' \subseteq G$; in any case if the real-world problem at hand implies non-complete graphs, one can complete them by "appropriately heavy" edges.

1) remove the vertices of $V \backslash V'$ and the edges of E incident to these vertices; let $F(V') = \{T_1, T_2, \dots, T_k\}$ be the so-obtained forest and assume that $v_1 \in V(T_1)$ and that, for $i, j = 2 \dots, k$, $i < j$ if the index of the root of T_i is smaller than that of the root of T_j;
2) for $i = 2, \dots, k$ add as father of the root of T_i its largest-index ancestor that is still present in $V(T)$.

Note that, given two vertices v_j and v_l with $j < l$ (in the bfs numbering of T), if v_j is not an ancestor of v_l in T, then edge (v_j, v_l) will never belong to any T' modification of T for any $V' \subseteq V$.



(a) An initial tree T ...

(b) ...and its modification by LEV

Fig. 1. An anticipatory spanning tree T and its adjustment.

Figure 1(b) gives an example of how strategy LEV works starting from an initial tree T shown in Figure 1(a) and assuming that vertices 3, 5, 6, 9, 13 and 16 are absent from V'.

The functional $E(G,T)$ associated with LEV is defined by:

$$E(G,T) = \sum_{V' \subseteq V} \Pr\left[V'\right] m\left(G', T'\right)$$

where $\Pr[V'] = \prod_{v_i \in V'} p_i \prod_{v_i \in V \backslash V'} (1 - p_i)$ is the distribution describing probability of occurrence of a specific subset $V' \subseteq V$, i.e., of the graph $G[V']$ and $m(G', T')$ is the value of the tree T' spanning V' produced by application of LEV on the anticipatory tree T.

Notice that, since there exist $2^{|V|}$ distinct sets V', any of them inducing a distinct subgraph $G[V']$ of G, both polynomial computation of $E(G,T)$ and tight combinatorial characterization of the optimal anticipatory solution are not always obvious or easy to perform.

Our goal is to study the following problem: *find an algorithm for taking a priori decisions, i.e., that determines a spanning tree T^*, that optimizes $E(G,T)$"*; this is PROBABILISTIC MIN SPANNING TREE.

In what follows, we show that this problem is **NP**-hard in general complete graphs with $p_1 = 1$. We then study approximation of this problem in metric graphs as well as in a particular subclass of them where edge-weights are either 1 or 2. The approximation ratio is defined as $E(G,T)/E(G,T^*)$.

The following result holds for the functional $E(G,T)$ associated with an anticipatory spanning tree T and the modification strategy LEV.

Proposition 1: Consider a complete graph $G(V,E)$, provided with a vertex-probability system $(p_i)_{i=1,\dots,n}$ with $p_1 = 1$, any edge (v_i, v_j) of which has weight d_{ij} and a spanning tree T of G. Then, the expectation associated with LEV can be expressed by:

$$\begin{aligned} E(G,T) = & \\ & \sum_{(v_i,v_j)\in T} p_i p_j d_{ij} \\ & + \sum_{v_i \in V} \sum_{\substack{v_j \in D(v_i) \\ (v_i,v_j) \notin T}} p_i p_j \times \prod_{v_k \in \mu[v_i,v_j]} \left(1 - p_k\right) d_{ij} \quad (1) \end{aligned}$$

where $D(v_i)$ denotes the set of successors of v_i in T and $\mu[v_i, v_j]$ denotes the set of vertices in the (unique) path of T from v_i to v_j not including neither of them. Expression (1) can be computed in polynomial time. Consequently, PROBABILISTIC MIN SPANNING TREE $\in$ **NPO**, the class of the optimization problems the decision versions of which are in **NP**.

Proof: Following LEV, if $(v_i, v_j) \in T$, then this edge will be in T' iff $v_i, v_j \in V'$; on the other hand, if $(v_i, v_j) \notin T$, then edge (v_i, v_j) will be added in T' iff $v_i, v_j \in V'$ and $v_j \in D(v_i)$ and every vertex in $\mu[v_i, v_j]$ is not in V'. From these observations we derive:

$$\begin{aligned} E(G,T) = & \\ & \sum_{V' \subseteq V} \Pr\left[V'\right] m\left(G', T'\right) \\ = & \sum_{V' \subseteq V} \Pr\left[V'\right] \sum_{(v_i,v_j)\in T'} d_{ij} \\ = & \sum_{V' \subseteq V} \Pr\left[V'\right] \sum_{(v_i,v_j)\in T' \cap T} d_{ij} \\ & + \sum_{V' \subseteq V} \Pr\left[V'\right] \sum_{(v_i,v_j)\in T' \cap (E \backslash T)} d_{ij} \\ = & \sum_{(v_i,v_j)\in T} \sum_{V' \subseteq V} \Pr\left[V'\right] \mathbf{1}_{\{(v_i,v_j)\in T'\}} d_{ij} \\ & + \sum_{(v_i,v_j)\in (E \backslash T)} \sum_{V' \subseteq V} \Pr\left[V'\right] \mathbf{1}_{\{(v_i,v_j)\in T'\}} d_{ij} \\ = & \sum_{(v_i,v_j)\in T} p_i p_j d_{ij} \\ & + \sum_{v_i \in V} \sum_{\substack{v_j \in D(v_i) \\ (v_i,v_j) \notin T}} p_i p_j \prod_{v_k \in \mu[v_i,v_j]} \left(1 - p_k\right) d_{ij} \end{aligned}$$

Clearly, (1) can be computed in polynomial time, since the ranges of the indices implied are polynomial. ■

As already mentioned, PROBABILISTIC MIN SPANNING TREE consists of determining an anticipatory spanning tree T^* of G minimizing $E(G,T)$.

Unfortunately, Proposition 1 does not derive a compact combinatorial characterization for the optimal anticipatory solution of PROBABILISTIC MIN SPANNING TREE.

In particular, the form of the functional does not imply solution, for instance, of some well-defined weighted version of the (deterministic) MIN SPANNING TREE. This is due to the second term of the expression for $E(G,T)$ in (1). There, the "costs" assigned to the edges depend on the structure of the anticipatory solution chosen and of the present subgraph of G.

The decision version of PROBABILISTIC MIN SPANNING TREE, denoted by PROBABILISTIC MIN SPANNING TREE(K) can be stated as follows: "given an edge-weighted complete graph $G(V,E)$, provided with a vertex-probability system $(p_i)_{i=1,\dots,n}$ with $p_1 = 1$ and a constant K, does there exist a tree T such that $E(G,T) \leqslant K$?", where $E(G,T)$ is given by (1).

Dealing with PROBABILISTIC MIN SPANNING TREE(K), by a technical reduction from 3 EXACT COVER, the following proposition holds.

Proposition 2: PROBABILISTIC MIN SPANNING TREE(K) is **NP**-complete.

Sketch of proof: PROBABILISTIC MIN SPANNING TREE $\in$ **NP**. In order to show completeness, we reduce 3 EXACT COVER to PROBABILISTIC MIN SPANNING TREE. 3 EXACT COVER that is defined as follows: "given a ground set X of size $3q$ and a collection $\mathcal{E}$ of $3q$ subsets of X each of size 3, does there exist a subcollection $\mathcal{E}' = \{S_1,\dots,S_q\}$ of $\mathcal{E}$ such that $\bigcup_{i=1}^{q} S_i = X$?" (obviously, $\mathcal{E}'$ is a partition of X). 3 EXACT COVER is **NP**-complete ([21]).

Consider an arbitrary instance $I(X,\mathcal{E})$ of 3 EXACT COVER; we construct the following instance for PROBABILISTIC MIN SPANNING TREE:

- the vertex-set V is a set of $6q+2$ vertices built by associating a vertex x_i with an element $x_i \in X$, a vertex y_j with a set $S_j \in \mathcal{E}$ and by adding a vertex r (playing the role of the omnipresent root) and a vertex s (representing the solution); for some positive fixed constant $p < 1/2$, vertices x_i are provided with probability p, vertices y_j with probability $1-p$ and vertices r and s with probability 1;
- edge-weights are defined as follows:
  - for every $S_j = \{x_{i_1}, x_{i_2}, x_{i_3}\}$, $j = 1,\dots,3q$, $d_{i_1 j} = d_{i_2 j} = d_{i_3 j} = 1$;
  - edges linking s to vertices y_j have weight $M > 0$ and those linking s to vertices of x_i have weight $M/p + 2$;
  - edges linking r to vertices y_j as well as edge (r,s) have all weight 0, while edges linking r to vertices x_i have weight $M/p^2 + 1$;
  - all the other edges have arbitrarily large weight $B \gg M/p^2 + 1$;
- $K = q(M(1+2p) + 3p(p+1))$.

It is easy to see that this reduction is polynomial. It is illustrated in Figure 2 where, for readability, some edges, in particular those of weight B are omitted.

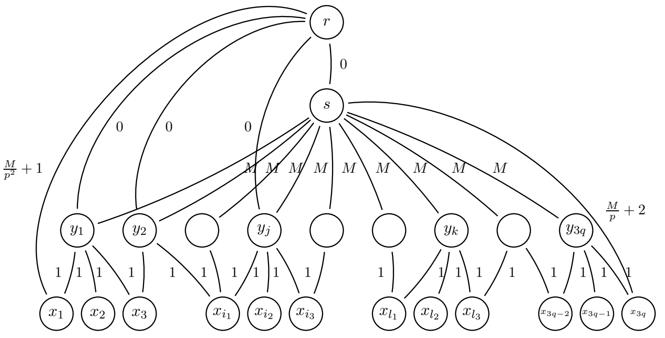


Fig. 2. An example for the reduction from 3 EXACT COVER to PROBABILISTIC MIN SPANNING TREE.

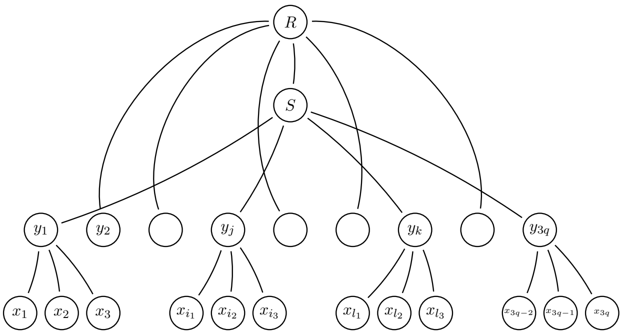


Fig. 3. The shape of T^*.

One can prove that if 3 EXACT COVER admits a solution $\mathcal{E}^*$, then G has a minimum spanning tree T^* the shape of which is as in Figure 3 and whose value is:

$$\begin{aligned} E(G,T^*) &= \\ &\quad 0 + 2q \times 0 + q[(1-p)M + 3p(1-p) \times 1 + \\ &\quad 3p(1-(1-p)) \times (M/p+2)] \\ &= q[M - pM + 3p - 3p^2 + 3pM + 6p^2] \\ &= q[M(1+2p) + 3p(p+1)] \;=\; K \end{aligned}$$

This can be done by inspection of all the possible solutions for MIN SPANNING TREE in G. ■

III. REOPTIMIZATION AND MIN SPANNING TREE

As mentioned in Section I, a complementary framework to the one of the a priori optimization, is the *reoptimization* consisting of solving ex nihilo and optimally the portion of the instance presented for optimization. Reoptimization is introduced in [4]. Let $\mathrm{opt}(G')$ refer to the weight of the optimum spanning tree on G' for every subgraph $G'(V',E')$ of G. The expected minimum weight over the distribution of subgraphs of G, i.e., the functional of reoptimization is defined by $E^*(G) = \sum_{V' \subseteq V} \Pr[V']\,\mathrm{opt}(G')$.

Obviously, denoting by T^* the optimal anticipatory solution of PROBABILISTIC MIN SPANNING TREE: $E^*(G) \leqslant E(G,T^*)$. Denote also by $\operatorname{opt}(G)$ the value of an optimal solution T^* for (deterministic) MIN SPANNING TREE. By elementary but technical combinatorial arguments, the following result holds.

Proposition 3: Consider a complete edge-weighted graph G defined on a set V of n vertices $V = \{v_1, \ldots, v_n\}$ associated with a system of vertex probabilities $p_1 = 1$, $p_i = p$, $i = 2, \ldots, n$. Then, $E^*(G) \geqslant p \operatorname{opt}(G)$.

Sketch of proof: Since vertex v_1 (assumed to be the root of every tree solution of MIN SPANNING TREE in every subgraph of G) is always present, setting $G' = G[V']$, $E^*(G)$ can be written as:

$$E^*(G) = \sum_{k=2}^{n} p^{k-1}(1-p)^{n-k} \times \sum_{\substack{V' \subseteq V \\ |V'|=k}} \operatorname{opt}(G')$$

Set $D_k = \sum_{V' \subseteq V, |V'|=k} \operatorname{opt}(G')$. Then:

$$E^*(G) = \sum_{k=2}^{n} p^{k-1}(1-p)^{n-k} D_k$$

By somewhat technical combinatorial arguments it can be proved that

$$D_k \geqslant \binom{n-2}{k-2} \operatorname{opt}(G)$$

and putting it together with the expression for $E^*(G)$ derives the claim. ■

Combining inequality $E^*(G) \leqslant E(G,T^*)$ and Proposition 3, the following holds: $E(G,T^*) \geqslant p \operatorname{opt}(G)$. Equality is attained for $p = 1$.

IV. PARTICULAR CASES

We study in this section some particular but natural cases carrying over assumptions either on the values of vertex-probabilities and/or edge-weights, or on the form of the anticipatory solution.

Revisit functional's expression (1). For a vertex v_i, denote by $f(v_i)$ its father in T, by $p_{f(i)}$ the presence probability of $f(v_i)$ and by $A(v_i)$ the set of its ancestors in T. Then, (1) can be rewritten as:

$$\begin{aligned} E(G,T) = & \\ & \sum_{v_i \in V \setminus \{v_1\}} p_i p_{f(i)} d_{if(v_i)} + \\ & \sum_{v_i \in V \setminus \{v_1\}} \sum_{v_j \in A(v_i) \setminus \{f(v_i)\}} p_i p_j \prod_{v_k \in \mu[v_i,v_j]} (1-p_k)\, d_{ij} \\ = & \sum_{v_i \in V \setminus \{v_1\}} C_i \end{aligned} \tag{2}$$

where:

$$\begin{aligned} C_i \quad = \quad & p_i p_{f(i)} d_{if(v_i)} + \\ & \sum_{v_j \in A(v_i) \setminus \{f(v_i)\}} p_i p_j \prod_{v_k \in \mu[v_i,v_j]} (1-p_k) d_{ij} \end{aligned}$$

and can be seen as the contribution of vertex v_i in $E(G,T)$.

Based upon (2) and the expression for C_i, the following result holds.

Proposition 4: If edge-weights are all identical, then:

$$E(G,T) = d \times \sum_{j=2}^{n} p_j$$

In this case all the anticipatory solutions have the same value.

Let us give an illustration of Proposition 4. It can easily be shown that, if $d_{ij} = d$, $(v_i, v_j) \in E$, then $C_i = d \times p_i$, $v_i \in V$. In order to give some intuition about it let us consider the anticipatory tree of Figure 4, assume that edge weights in the input-graph are identical and equal to d and take, say, vertex 7. The contribution of it in (1) is:

$$\begin{aligned} C_7 = & \\ & d \times p_7 \left[p_5 + p_4 (1-p_5) + p_2 (1-p_5)(1-p_4) + \right. \\ & \left. (1-p_5)(1-p_4)(1-p_2)\right] \\ = \quad & d \times p_7 \end{aligned} \tag{3}$$

The same holds for the contribution of any other vertex in the tree.

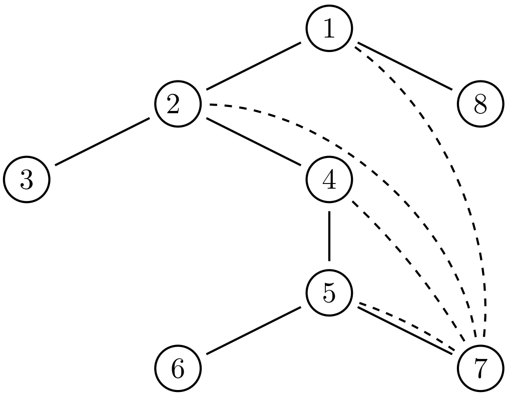


Fig. 4. About a reinterpretation of functional's expression (1).

Indeed, consider some vertex $v_i \in V$ and assume, for simplicity, that vertices in the path of T from v_1 to v_i are numbered from 1 to i. By writing down C_i and by some algebra as previously in (3) we derive $C_i = d \times p_i$. Hence, (2) becomes:

$$E(G,T) = d \times \sum_{j=2}^{n} p_j$$

Corollary 1: If $p_1 = 1$, $p_i = p$, $i = 2, \ldots, n$ and $d_{ij} = d$, $(v_i, v_j) \in E$, $i \neq j$, then, for any tree T spanning V, $E(G,T) = dp(n-1)$.

The above can be directly generalized for deriving a general upper bound for $E(G,T)$ and for every anticipatory solution T. Set $D = \max\{d_{ij} : (v_i, v_j) \in E\}$.

Corollary 2: If $D = \max\{d_{ij} : (v_i, v_j) \in E\}$ then, for any anticipatory solution T of PROBABILISTIC MIN SPANNING TREE, $E(G,T) \leqslant D \times \sum_{j=2}^{n} p_j$.

Let us now address the following question: "can an optimal solution for MIN SPANNING TREE remain an optimal solution for PROBABILISTIC MIN SPANNING TREE and if yes under which conditions?". In what follows we deal with two particular structures of trees, the star and the path.

Consider the star rooted at (the omnipresent) vertex v_1. The following result holds.

Proposition 5: Let T be a star rooted at v_1. If T is an optimal solution for MIN SPANNING TREE then it is also an optimal anticipatory solution for PROBABILISTIC MIN SPANNING TREE.

Proof: Recall that by (2), $E(G,T) = \sum_{v_i \in V \setminus \{v_1\}} C_i$ where C_i is given by:

$$\begin{aligned} C_i = {} & \\ & p_i p_{f(i)} d_{if(v_i)} + \\ & \sum_{v_j \in A(v_i) \setminus \{f(v_i)\}} p_i p_j \prod_{v_k \in \mu[v_i, v_j]} (1 - p_k)\, d_{ij} \end{aligned}$$

Observe now that, if the vertices of T are numbered in a dfs order (starting from the root) and if the set $A(v_i)$ of the ancestors of a vertex v_i in T is exactly the set $A(v_i) = \{v_1, v_2, \ldots, v_{i-1}\}$, then C_i can be written as:

$$C_i = \sum_{j=1}^{i-1} p_i p_j d_{ij} \prod_{l=j+1}^{i-1} (1 - p_l) \tag{4}$$

Since the star T is a minimum spanning tree, it holds that, for any vertex i, $d_{1i} \leqslant d_{ij}$, for every $j \neq 1, i$. Hence:

$$C_i \geqslant p_i \times d_{1i} \times \sum_{j=1}^{i-1} p_j \prod_{l=j+1}^{i-1} (1 - p_l)$$

If we denote by C_i^T the contribution of vertex v_i in the functional $E(T)$ of the star T, then, for every i, $C_i^T = p_i \times d_{1i}$. So, in order to complete the proof of the proposition, we have to show that, for any $v_i \in V$, $C_i^T \leqslant C_i$, where C_i refers to every other spanning tree of G. For this, it suffices to prove that:

$$\sum_{j=1}^{i-1} p_j \prod_{l=j+1}^{i-1} (1 - p_l) \geqslant 1 \tag{5}$$

We show (5) by induction on i. For $i = 2$, the lefthand side of (5) is equal to $p_1 = 1$, so the inequality claimed is true. Suppose it true for $i = n$, i.e.:

$$\sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n-1} (1 - p_l) \geqslant 1 \tag{6}$$

Then, at range $n + 1$ it holds:

$$\begin{aligned} & \sum_{j=1}^{n} p_j \prod_{l=j+1}^{n} (1 - p_l) = \\ & \quad \sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n} (1 - p_l) + p_n \\ = & \quad \sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n-1} (1 - p_l) \times (1 - p_n) + p_n \\ = & \quad (1 - p_n) \times \sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n-1} (1 - p_l) + p_n \\ \overset{(6)}{\geqslant} & \quad (1 - p_n) + p_n \quad = \quad 1 \end{aligned}$$

as claimed. ■

Unfortunately, in the case where optimal solution for MIN SPANNING TREE is a path, optimality of such a solution for PROBABILISTIC MIN SPANNING TREE cannot be derived as previously in the case of stars.

Indeed, consider a complete graph G, the adjacency matrix of which is given in Table I and its vertex-probability system is $(1, p \ldots, p)$, with $p < (K-2)/(K-1)$ and $K \geqslant n$.

TABLE I
THE ADJACENCY MATRIX OF A GRAPH G WHERE OPTIMAL SOLUTIONS FOR MIN SPANNING TREE AND PROBABILISTIC MIN SPANNING TREE DO NOT COINCIDE.

	v_1	v_2	v_3	$\cdots$	v_{n-2}	v_{n-1}	v_n
v_1	0	1	2	$\cdots$	2	2	2
v_2	1	0	1	$\cdots$	2	2	2
v_3	2	1	0	$\cdots$	2	2	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$
v_{n-2}	2	2	2	$\cdots$	2	1	K
v_{n-1}	2	2	2	$\cdots$	1	0	1
v_n	2	2	2	$\cdots$	K	1	0

Optimal MIN SPANNING TREE-solution in G is unique and is the path $P = (1, 2, \ldots, n-1)$ with value $n - 1$. The functional $E(G, P)$ of path P is:

$$E(G,P) = (2n-3)p + (K-n)p^2 - (K-2)p^3$$

On the other hand, the unique optimal anticipatory solution for PROBABILISTIC MIN SPANNING TREE is the tree T^* of Figure 5 with functional's value:

$$E(G,T^*) = (2n-3)p + (2-n)p^2 + p^3 < E(G,P)$$

when $p < (K-2)/(K-1)$.

V. APPROXIMATION OF PROBABILISTIC METRIC MIN SPANNING TREE

In this section, we study PROBABILISTIC METRIC MIN SPANNING TREE problem, that is PROBABILISTIC MIN SPANNING TREE in metric complete graphs, i.e., in complete graphs whose edge-weights satisfy the triangular property that can

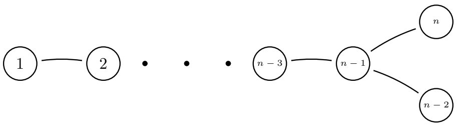


Fig. 5. The optimal solution of the graph of Table I.

be expressed as follows: if (v_i, v_j, v_k) is a K_3 in G, then $d_{ab} \leqslant d_{ac} + d_{bc}$, for any permutation (a, b, c) of $\{i, j, k\}$.

Note that, as it can be seen from the proof of Proposition 2, it does not apply in the case of metric graphs. Indeed, the complexity status of PROBABILISTIC METRIC MIN SPANNING TREE remains open, even if we feel that this variant is also **NP**-complete. In what follows we give approximation results for the general case of PROBABILISTIC METRIC MIN SPANNING TREE as well as for a natural subcase, namely PROBABILISTIC MIN SPANNING TREE 1,2, where edge weights are either 1, or 2.

We denote by $\hat{T}$ and T^* a tree computed by Kruskal's algorithm (i.e., a minimum spanning tree of G) and an optimal anticipatory solution for PROBABILISTIC METRIC MIN SPANNING TREE, respectively, and we assume that they are represented as sets of edges.

Observe also that, by the metric property the weight of any edge of G is smaller than the weight of any spanning tree of G and a fortiori than the weight $m(G, \hat{T}) = \text{opt}(G)$ of $\hat{T}$. Indeed, let (v_i, v_j) be any edge of G, T be some spanning tree of G and $P(v_i, v_j)$ be the unique path from v_i to v_j in T.

According to the metric hypothesis:

$$d_{ij} \leqslant w\left(P(v_i, v_j)\right) \leqslant m(G, T)$$

where $w(P(v_i, v_j))$ is the weight of the path $P(v_i, v_j)$. In what follows, given a set Q of (weighted) edges, we denote by $w(Q)$ its total weight.

We first handle the general metric case. For any $V_i \subseteq V$, we set $G_i(V_i, E_i) = G[V_i]$. Let T_i^* and $\hat{T}_i$ be the spanning trees on G_i resulting from the application of strategy LEV on T^* and $\hat{T}$, respectively. Set:

$$\begin{aligned} r\left(G_i\right) &= \frac{m\left(G_i, \hat{T}_i\right)}{m\left(G_i, T_i^*\right)} \\ E(r) &= \sum_{V_i \subseteq V} \Pr\left[V_i\right] r\left(G_i\right) \end{aligned}$$

Quantity $E(r)$ is indeed the average approximation ratio of a minimum spanning tree for PROBABILISTIC METRIC MIN SPANNING TREE. In the following proposition an upper bound is given for $E(r)$.

Proposition 6: $E(r) \leqslant (n+2)/4$.

Proof: Fix an induced subgraph G_i of G and let $\hat{T} \cap \hat{T}_i = S$. Edges of S are part of an optimal spanning tree on G and thus, they are also part of an optimal spanning tree of G_i. Indeed, revisiting the proof of optimality of Kruskal's Algorithm, one can see that a tree T is a minimum spanning tree on G, iff all the edges of T are of minimum weight in at least one cut of G. Applying this to PROBABILISTIC MIN SPANNING TREE, any edge e belonging to S is of minimum weight in at least one cut of G; thus, e is also of minimum weight in one cut in any subgraph G_i (provided that e appears in G_i) and, therefore, it belongs to a minimum spanning tree in all the subgraphs of G where it is present.

Discussion just above leads to:

$$w(S) \leqslant m\left(G_i, T_i^*\right) \tag{7}$$

The edge-set $\hat{T}_i \setminus S$ is the set of the edges used by LEV to reconnect the S. As observed in the beginning of Section V, the weight of each edge of $\hat{T}_i \setminus S$ is smaller than, or equal to, $m(G_i, T_i^*)$, so:

$$w\left(\hat{T}_i \setminus S\right) \leqslant \left|\hat{T}_i \setminus S\right| m\left(G_i, T_i^*\right) \tag{8}$$

Combining (7) and (8), we get:

$$r_i = \frac{m\left(G_i, \hat{T}_i\right)}{m\left(G_i, T_i^*\right)} \leqslant 1 + \left|\hat{T}_i \setminus S\right| \tag{9}$$

The quantity $|\hat{T}_i \setminus S|$ is, as mentioned above, the number of edges inserted by strategy LEV to reconnect S, but it also represents the number of vertices present in G_i, but whose fathers in $\hat{T}$ (assumed rooted at v_1) are absent from G_i. For each vertex of $\hat{T}$ except for those directly connected to the root v_1, the probability to be present in G_i but not its father is $p(1-p)$. Obviously, for the vertices directly connected to the root, this probability is 0. In order to count the number of edges in $\hat{T}_i \setminus S$, one can consider a set of $n-1-X$ Bernoulli trials (where X is the number of vertices directly connected to v_1 in $\hat{T}$), with a probability of success $p(1-p)$, each success adding an edge to $\hat{T}_i \setminus S$. In this way, $|\hat{T}_i \setminus S|$ is a random variable following a binomial law, so one can directly compute its expectation:

$$\begin{aligned} \left|\hat{T}_i \setminus S\right| &\sim B\left(n-1-X, p(1-p)\right) \\ E\left(\left|\hat{T}_i \setminus S\right|\right) &= (n-1-X)p(1-p) \end{aligned} \tag{10}$$

Summing (9) for each G_i, we derive:

$$E\left(\frac{m\left(G_i, \hat{T}_i\right)}{m\left(G_i, T_i^*\right)}\right) \leqslant 1 + E\left(\left|\hat{T}_i \setminus S\right|\right)$$

and combining it with (10), we can easily get:

$$\begin{aligned} E\left(\frac{m\left(G_i, \hat{T}_i\right)}{m\left(G_i, T_i^*\right)}\right) &\leqslant 1 + (n-1-X)p(1-p) \\ &\overset{X \geqslant 1}{\leqslant} \frac{n}{4} + \frac{1}{2} = \frac{n+2}{4} \end{aligned}$$

as claimed. ■

We now study the approximation of PROBABILISTIC METRIC MIN SPANNING TREE. The following result can be proved.

Proposition 7: The PROBABILISTIC METRIC MIN SPANNING TREE problem is approximable in polynomial time within ratio bounded above by:

$$\min\left\{1+(n-2)(1-p), \frac{2}{p}\right\}$$

To conclude the paper, let us focus on a particular but natural and well-studied class of metric complete graphs where edge weights are either 1 or 2. It is easy to see that any such graph is metric. The following result, that will be subsequently improved, holds for PROBABILISTIC MIN SPANNING TREE $1{,}2$.

Proposition 8: A minimum spanning tree of G is a $(2-p)$-approximation for PROBABILISTIC MIN SPANNING TREE $1{,}2$.

In what follows we refine the result above. For this, we consider an execution of Kruskal's algorithm that starts by introducing in the tree all the edges of weight 1 incident to the vertex v_1. Let us denote by $\tilde{T}$ the spanning tree so constructed; notice that $\tilde{T}$ is a minimum spanning tree for G.

Proposition 9: $\tilde{T}$ approximates T^* within ratio:

$$\frac{1+(2-p)(n-2)}{n-1+\frac{(1-p)^2-(1-p)^n}{p}}$$

One can see that when p is fixed (i.e., independent on n), the approximation ratio achieved is strictly better than 2. On the other hand, when $p \sim 1/n$ then, since:

$$\lim_{n\rightarrow+\infty}\frac{(1-p)^n}{p}=\frac{n}{e}$$

the ratio claimed in Proposition 9 tends to 1.225, for large values of n.

If $p \sim 1/n^k$, $k > 1$, then for large values of n, this ratio tends to 1. Finally, if $p \sim 1/n^k$, $k < 1$, then (always for large values of n) the ratio is asymptotically equal to 2.

VI. CONCLUSION

In this paper we have treated the PROBABILISTIC MIN SPANNING TREE problem under the framework of probabilistic combinatorial optimization. We have proposed a fast modification strategy (LEV) for reconstructing a second-stage tree and shown that problem of optimizing the expectation of the second-stage cost by selecting an appropriate first-stage (anticipatory) solution is in **NPO** under the proposed modification strategy and we have shown that the general case of PROBABILISTIC MIN SPANNING TREE is **NP**-hard.

We have also given approximation results for the probabilistic problem associated with the LEV strategy. We also have studied particular cases of anticipatory solutions.

Finally we have given approximation results for PROBABILISTIC METRIC MIN SPANNING TREE and PROBABILISTIC MIN SPANNING TREE $1{,}2$.

There are several open questions subject for further research. To our opinion, the major among them are the complexities of PROBABILISTIC METRIC MIN SPANNING TREE and PROBABILISTIC MIN SPANNING TREE $1{,}2$ (we conjecture that they are both **NP**-hard) and the improvement of their approximation ratios.

REFERENCES

[1] G. W. Dantzig, "Linear programming under uncertainty," *Management Sci.*, vol. 1, pp. 197–206, 1951.

[2] A. Prekopa, *Stochastic programming*. The Netherlands: Kluwer Academic Publishers, 1995.

[3] P. Kouvelis and G. Yu, *Robust discrete optimization and its applications*. Boston: Kluwer Academic Publishers, 1997.

[4] P. Jaillet, "Probabilistic traveling salesman problem," Operations Research Center, MIT, Cambridge Mass., USA, Technical Report 185, 1985.

[5] D. J. Bertsimas, "Probabilistic combinatorial optimization problems," PhD Thesis, Operations Research Center, MIT, Cambridge Mass., USA, 1988.

[6] I. Averbakh, O. Berman, and D. Simchi-Levi, "Probabilistic a priori routing-location problems," *Naval Res. Logistics*, vol. 41, pp. 973–989, 1994.

[7] D. J. Bertsimas, "On probabilistic traveling salesman facility location problems," *Transportation Sci.*, vol. 3, pp. 184–191, 1989.

[8] ——, "The probabilistic minimum spanning tree problem," *Networks*, vol. 20, pp. 245–275, 1990.

[9] D. J. Bertsimas, P. Jaillet, and A. Odoni, "A priori optimization," *Oper. Res.*, vol. 38, no. 6, pp. 1019–1033, 1990.

[10] P. Jaillet, "A priori solution of a traveling salesman problem in which a random subset of the customers are visited," *Oper. Res.*, vol. 36, no. 6, pp. 929–936, 1988.

[11] ——, "Shortest path problems with node failures," *Networks*, vol. 22, pp. 589–605, 1992.

[12] P. Jaillet and A. Odoni, "The probabilistic vehicle routing problem," in *Vehicle routing: methods and studies*, B. L. Golden and A. A. Assad, Eds. Amsterdam: North Holland, 1988.

[13] L. Bianchi, J. Knowles, and N. Bowler, "Local search for the probabilistic traveling salesman problem: correlation to the 2-p-opt and 1-shift algorithms," *European J. Oper. Res.*, vol. 161, no. 1, pp. 206–219, 2005.

[14] C. Murat and V. T. Paschos, "On the probabilistic minimum coloring and minimum k-coloring," *Discrete Appl. Math.*, vol. 154, pp. 564–586, 2006.

[15] N. Bourgeois, F. Della Croce, B. Escoffier, C. Murat, and V. T. Paschos, "Probabilistic coloring of bipartite and split graphs," *J. Comb. Optimization*, vol. 17, no. 3, pp. 274–311, 2009.

[16] C. Murat and V. T. Paschos, "A priori optimization for the probabilistic maximum independent set problem," *Theoret. Comput. Sci.*, vol. 270, pp. 561–590, 2002, preliminary version available at http://www.lamsade.dauphine.fr/~paschos/documents/c166.pdf.

[17] ——, "The probabilistic minimum vertex-covering problem," *Int. Trans. Opl Res.*, vol. 9, no. 1, pp. 19–32, 2002, preliminary version available at http://www.lamsade.dauphine.fr/~paschos/documents/c170.pdf.

[18] ——, "The probabilistic longest path problem," *Networks*, vol. 33, pp. 207–219, 1999.

[19] V. T. Paschos, O. A. Telelis, and V. Zissimopoulos, "Steiner forests on stochastic metric graphs," in *Proc. Conference on Combinatorial Optimization and Applications, COCOA'07*, ser. Lecture Notes in Computer Science, A. Dress, Y. Xu, and B. Zhu, Eds., vol. 4616. Springer-Verlag, 2007, pp. 112–123.

[20] ——, "Probabilistic models for the STEINER TREE problem," *Networks*, to appear.

[21] M. R. Garey and D. S. Johnson, *Computers and intractability. A guide to the theory of NP-completeness*. San Francisco: W. H. Freeman, 1979.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Efficient Portfolio Optimization with Conditional Value at Risk

Włodzimierz Ogryczak
Warsaw University of Technology
Institute of Control & Comput. Engg.
00–665 Warsaw, Poland
E-mail: ogryczak@ia.pw.edu.pl

Tomasz Sliwinski
Warsaw University of Technology
Institute of Control & Comput. Engg.
00–665 Warsaw, Poland
E-mail: tsliwins@ia.pw.edu.pl

Abstract—**The portfolio optimization problem is modeled as a mean-risk bicriteria optimization problem where the expected return is maximized and some (scalar) risk measure is minimized. In the original Markowitz model the risk is measured by the variance while several polyhedral risk measures have been introduced leading to Linear Programming (LP) computable portfolio optimization models in the case of discrete random variables represented by their realizations under specified scenarios. Among them, the second order quantile risk measures, recently, become popular in finance and banking. The simplest such measure, now commonly called the Conditional Value at Risk (CVaR) or Tail VaR, represents the mean shortfall at a specified confidence level. Recently, the second order quantile risk measures have been introduced and become popular in finance and banking. The corresponding portfolio optimization models can be solved with general purpose LP solvers. However, in the case of more advanced simulation models employed for scenario generation one may get several thousands of scenarios. This may lead to the LP model with huge number of variables and constraints thus decreasing the computational efficiency of the model since the number of constraints (matrix rows) is usually proportional to the number of scenarios. while the number of variables (matrix columns) is proportional to the total of the number of scenarios and the number of instruments. We show that the computational efficiency can be then dramatically improved with an alternative model taking advantages of the LP duality. In the introduced models the number of structural constraints (matrix rows) is proportional to the number of instruments thus not affecting seriously the simplex method efficiency by the number of scenarios.**

I. Introduction

FOLLOWING Markowitz [13], the portfolio selection problem is modeled as a mean-risk bicriteria optimization problem where the expected return is maximized and some (scalar) risk measure is minimized. In the original Markowitz model the risk is measured by the variance but several polyhedral risk measures have been introduced leading to Linear Programming (LP) computable portfolio optimization models in the case of discrete random variables represented by their realizations under specified scenarios. The simplest LP computable risk measures are dispersion measures similar to the variance. Konno and Yamazaki [6] presented the portfolio selection model with the mean absolute deviation (MAD) while Young [33] introduced the Minimax model. Yitzhaki [32] introduced the mean-risk model using Gini's mean (absolute) difference as the risk measure. The Gini's mean difference turn out to be a special aggregation technique of the multiple criteria LP model [18] based on the pointwise comparison of the absolute Lorenz curves. The latter leads the quantile shortfall risk measures directly related to the dual theory of choice under risk [26], [28], [31] which are more commonly used and accepted. Recently, the second order quantile risk measures have been introduced in different ways by many authors [2], [4], [16], [17], [27]. The measure, usually called the Conditional Value at Risk (CVaR) or Tail VaR, represents the mean shortfall at a specified confidence level. The CVaR measures maximization is consistent with the second degree stochastic dominance [19]. Several empirical analyses confirm its applicability to various financial optimization problems [1], [11].

This paper is focused on computational efficiency of the portfolio optimization models based on the CVaR or the Minimax measures. We assume that the instruments returns are represented by their realizations under T scenarios. The basic LP model for the CVaR portfolio optimization contains then T auxiliary variables as well as T corresponding linear inequalities. Actually, the number of structural constraints in the LP model (matrix rows) is proportional to the number of scenarios T, while the number of variables (matrix columns) is proportional to the total of the number of scenarios and the number of instruments $T+n$. Hence, its dimensionality is proportional to the number of scenarios T. It does not cause any computational difficulties for a few hundreds of scenarios as in computational analysis based on historical data. However, in the case of more advanced simulation models employed for scenario generation one may get several thousands of scenarios [25]. This may lead to the LP model with huge number of auxiliary variables and constraints thus decreasing the computational efficiency of the model. Actually, in the case of fifty thousand scenarios and one hundred instruments the model may require more than an hour computation time with the state-of-art LP solver (CPLEX code). To overcome this difficulty some alternative solution approaches are searched trying to reformulate the optimization problems as two-stage recourse problems [8] or to employ nondifferential optimization techniques [9]. We show that the computational efficiency can be then dramatically improved with an alternative model formulation taking advantages of the LP duality. In the introduced model the number of structural constraints is proportional to the number of instruments n

978-83-60810-27-9/09/$25.00 © 2010 IEEE

while only the number of variables is proportional to the number of scenarios T thus not affecting so seriously the simplex method efficiency. Indeed, the computation time is then below a minute.

The paper is organized as follows. In the next section we introduce briefly basics of the mean-risk portfolio optimization with the LP computable risk measures. In Section 3 we develop and test computationally efficient optimization models taking advantages of the LP duality.

II. Portfolio Optimization and Risk Measures

The portfolio optimization problem considered in this paper follows the original Markowitz' formulation and is based on a single period model of investment. At the beginning of a period, an investor allocates the capital among various securities, thus assigning a nonnegative weight (share of the capital) to each security. Let $J = \{1, 2, \ldots, n\}$ denote a set of securities considered for an investment. For each security $j \in J$, its rate of return is represented by a random variable R_j with a given mean $\mu_j = \mathbf{E}\{R_j\}$. Further, let $\mathbf{x} = (x_j)_{j=1,2,\ldots,n}$ denote a vector of decision variables x_j expressing the weights defining a portfolio. The weights must satisfy a set of constraints to represent a portfolio. The simplest way of defining a feasible set Q is by a requirement that the weights must sum to one and they are nonnegative (short sales are not allowed), i.e.

$$Q = \{\mathbf{x} \;:\; \sum_{j=1}^{n} x_j = 1, \quad x_j \geq 0 \quad \text{for } j = 1, \ldots, n\} \tag{1}$$

Hereafter, we perform detailed analysis for the set Q given with constraints (1). Nevertheless, the presented results can easily be adapted to a general LP feasible set given as a system of linear equations and inequalities, thus allowing one to include short sales, upper bounds on single shares or portfolio structure restrictions which may be faced by a real-life investor.

Each portfolio $\mathbf{x}$ defines a corresponding random variable $R_{\mathbf{x}} = \sum_{j=1}^{n} R_j x_j$ that represents the portfolio rate of return while the expected value can be computed as $\mu(\mathbf{x}) = \sum_{j=1}^{n} \mu_j x_j$. We consider T scenarios with probabilities p_t (where $t = 1, \ldots, T$). We assume that for each random variable R_j its realization r_{jt} under the scenario t is known. Typically, the realizations are derived from historical data treating T historical periods as equally probable scenarios ($p_t = 1/T$). Although the models we analyze do not take advantages of this simplification. The realizations of the portfolio return $R_{\mathbf{x}}$ are given as $y_t = \sum_{j=1}^{n} r_{jt} x_j$.

The portfolio optimization problem is modeled as a mean-risk bicriteria optimization problem where the mean $\mu(\mathbf{x})$ is maximized and the risk measure $\varrho(\mathbf{x})$ is minimized. In the original Markowitz model, the standard deviation was used as the risk measure. Several other risk measures have been later considered thus creating the entire family of mean-risk models (c.f., [10], [11]). These risk measures, similar to the standard deviation, are not affected by any shift of the outcome scale and are equal to 0 in the case of a risk-free portfolio while taking positive values for any risky portfolio. Unfortunately, such risk measures are not consistent with the stochastic dominance order [15] or other axiomatic models of risk-averse preferences [29] and coherent risk measurement [2].

In stochastic dominance, uncertain returns (modeled as random variables) are compared by pointwise comparison of some performance functions constructed from their distribution functions. The first performance function $F_{\mathbf{x}}^{(1)}$ is defined as the right-continuous cumulative distribution function: $F_{\mathbf{x}}^{(1)}(\eta) = F_{\mathbf{x}}(\eta) = \mathbf{P}\{R_{\mathbf{x}} \leq \eta\}$ and it defines the first degree stochastic dominance (FSD). The second function is derived from the first as $F_{\mathbf{x}}^{(2)}(\eta) = \int_{-\infty}^{\eta} F_{\mathbf{x}}(\xi)\, d\xi$ and it defines the second degree stochastic dominance (SSD). We say that portfolio $\mathbf{x}'$ dominates $\mathbf{x}''$ under the SSD ($R_{\mathbf{x}'} \succ_{SSD} R_{\mathbf{x}''}$), if $F_{\mathbf{x}'}^{(2)}(\eta) \leq F_{\mathbf{x}''}^{(2)}(\eta)$ for all η, with at least one strict inequality. A feasible portfolio $\mathbf{x}^0 \in Q$ is called SSD efficient if there is no $\mathbf{x} \in Q$ such that $R_{\mathbf{x}} \succ_{SSD} R_{\mathbf{x}^0}$. Stochastic dominance relates the notion of risk to a possible failure of achieving some targets. As shown by Ogryczak and Ruszczyński [19], function $F_{\mathbf{x}}^{(2)}$, used to define the SSD relation, can also be presented as follows: $F_{\mathbf{x}}^{(2)}(\eta) = \mathbf{E}\{\max\{\eta - R_{\mathbf{x}}, 0\}\}$ and thereby its values are LP computable for returns represented by their realizations y_t.

An alternative characterization of the SSD relation can be achieved with the so-called Absolute Lorenz Curves (ALC) [16], [30] which represent the second quantile functions defined as $F_{\mathbf{x}}^{(-2)}(0) = 0$ and

$$F_{\mathbf{x}}^{(-2)}(p) = \int_{0}^{p} F_{\mathbf{x}}^{(-1)}(\alpha) d\alpha \tag{2}$$

where $F_{\mathbf{x}}^{(-1)}(p) = \inf\ \{\eta : F_{\mathbf{x}}(\eta) \geq p\}$ is the left-continuous inverse of the cumulative distribution function $F_{\mathbf{x}}$. The pointwise comparison of ALCs is equivalent to the SSD relation [20] in the sense that $R_{\mathbf{x}'} \succeq_{SSD} R_{\mathbf{x}''}$ if and only if $F_{\mathbf{x}'}^{(-2)}(\beta) \geq F_{\mathbf{x}''}^{(-2)}(\beta)$ for all $0 < \beta \leq 1$. Moreover,

$$\begin{aligned} F_{\mathbf{x}}^{(-2)}(\beta) &= \max_{\eta \in R} \left[\beta\eta - F_{\mathbf{x}}^{(2)}(\eta)\right] \\ &= \max_{\eta \in R} \left[\beta\eta - \mathbf{E}\{\max\{\eta - R_{\mathbf{x}}, 0\}\}\right] \end{aligned} \tag{3}$$

where η is a real variable taking the value of β-quantile $Q_\beta(\mathbf{x})$ at the optimum. For a discrete random variable represented by its realizations y_t problem (3) becomes an LP.

For any real tolerance level $0 < \beta \leq 1$, the normalized value of the ALC defined as

$$M_\beta(\mathbf{x}) = F_{\mathbf{x}}^{(-2)}(\beta)/\beta \tag{4}$$

is called the *Conditional Value-at-Risk (CVaR)* or Tail VaR or Average VaR. The CVaR measure is an increasing function of the tolerance level β, with $M_1(\mathbf{x}) = \mu(\mathbf{x})$. For any $0 < \beta < 1$, the CVaR measure is SSD consistent [20] and coherent [24]. Opposite to deviation type risk measures, for coherent measures larger values are preferred and therefore the measures are sometimes called safety measures [11]. Due to (3), for a discrete random variable represented by its realizations y_t the CVaR measures are LP computable. It is important to notice that although the quantile risk measures (VaR and CVaR) were introduced in banking as extreme risk measures for very small tolerance levels (like $\beta = 0.05$), for the portfolio optimization good results have been provided by rather larger tolerance levels [11]. For β approaching 0, the CVaR measure tends to the Minimax measure $M(\mathbf{x}) = \min_{t=1,\ldots,T} y_t = \min_{t=1,\ldots,T} \sum_{j=1}^n r_{jt}x_j$ introduced to portfolio optimization by Young [33].

The commonly accepted approach to implementation of the Markowitz-type mean-risk model is based on the use of a specified lower bound μ_0 on expected returns while optimizing the risk measure. This bounding approach provides a clear understanding of investor preferences and a clear definition of optimal solution portfolio to be sought. For coherent and SSD consistent risk measures we consider the approach results in the following maximization problem:

$$\max\{\ \varrho(\mathbf{x}) :\quad \mu(\mathbf{x}) \geq \mu_0, \quad \mathbf{x} \in Q\} \tag{5}$$

where $\varrho(\mathbf{x}) = M_\beta(\mathbf{x})$ or $\varrho(\mathbf{x}) = M(\mathbf{x})$ respectively

We demonstrate that such portfolio optimization models can be effectively solved for large numbers of scenarios while taking advantages of appropriate dual LP formulations.

III. Computational LP Models

Let us consider portfolio optimization problem with security returns given by discrete random variables with realization r_{jt} thus leading to LP portfolio optimization model (5) for the risk measures we consider.

Following (3) and (4), the CVaR portfolio optimization model can be formulated as the following LP problem:

$$\begin{array}{ll} \max & \eta - \dfrac{1}{\beta}\sum_{t=1}^T p_t d_t \\ \text{s.t.} & \sum_{j=1}^n \mu_j x_j \geq \mu_0 \\ & \sum_{j=1}^n x_j = 1 \\ & x_j \geq 0,\ j = 1,\ldots,n \\ & d_t - \eta + \sum_{j=1}^n r_{jt}x_j \geq 0,\ d_t \geq 0,\ t = 1,\ldots,T \end{array} \tag{6}$$

where η is unbounded variable. Except from the core portfolio constraints (1) and the expected return bound, the model (6) contains T nonnegative variables d_t plus single η variable and T corresponding linear inequalities. Hence, its dimensionality is proportional to the number of scenarios T. Exactly, the LP model (6) contains $T + n + 1$ variables and $T + 2$ constraints. It does not cause any computational difficulties for a few hundreds of scenarios as in several computational analysis based on historical data [12]. However, in the case of more advanced simulation models employed for scenario generation one may get several thousands of scenarios. This may lead to the LP model (6) with huge number of variables and constraints thus decreasing the computational efficiency of the model. If the core portfolio constraints contain only linear relations, like (1), then the computational efficiency can easily be achieved by taking advantages of an alternative LP formulation.

Note that, due to the finite distribution of returns, the CVaR measure is well defined by the following optimization

$$\begin{aligned} M_\beta(\mathbf{x}) \quad = \quad & \min_{u_t}\{\sum_{t=1}^T(\sum_{j=1}^n r_{jt}x_j)u_t : \sum_{t=1}^T u_t = 1, \\ & 0 \leq u_t \leq \frac{p_t}{\beta}\ t = 1,\ldots,T\} \end{aligned} \tag{7}$$

that implements directly the ALC formula (2). The entire CVaR portfolio optimization problem may be respectively expressed as

$$\max_{\mathbf{x}\in Q} M_\beta(\mathbf{x}) = \max_{\mathbf{x}\in Q}\min_{\mathbf{u}\in U}\sum_{t=1}^T(\sum_{j=1}^n r_{jt}x_j)u_t \tag{8}$$

where

$$\begin{aligned} U \quad = \quad & \{(u_1,\ldots,u_T) : \sum_{t=1}^T u_t = 1, \\ & 0 \leq u_t \leq \frac{p_t}{\beta} \quad t = 1,\ldots,T\}. \end{aligned}$$

The inner optimization problem represents (7). It is an LP for a given vector $\mathbf{x}$. However, within the entire portfolio optimization problem (8) the objective function $\sum_{t=1}^T(\sum_{j=1}^n r_{jt}x_j)u_t$ becomes nonlinear for $\mathbf{x}$ being a vector of variables. This difficulty can be overcome by an equivalent dual LP formulation of problem (7). Indeed, introducing dual variable η corresponding to the equation $\sum_{t=1}^T u_t = 1$ and variables d_t corresponding to upper bounds on u_t one gets the LP dual

$$\begin{aligned} M_\beta(\mathbf{x}) \quad = \quad & \max_{q,d_t}\ \{\eta - \frac{1}{\beta}\sum_{t=1}^T p_t d_t : \\ & \sum_{j=1}^n r_{jt}x_j \geq \eta - d_t,\ d_t \geq 0\ \forall\ t\}. \end{aligned} \tag{9}$$

This leads us to the standard LP model (6) for the CVaR portfolio optimization.

An alternative CVaR optimization model can be built by taking advantages of the minimax theorem. Since both sets Q and U are convex polyhedra, formula (8) can be rewritten into a dual form

$$\begin{aligned} & \max_{\mathbf{x}\in Q}\min_{\mathbf{u}\in U}\sum_{t=1}^T(\sum_{j=1}^n r_{jt}x_j)u_t \\ & = \min_{\mathbf{u}\in U}\max_{\mathbf{x}\in Q}\sum_{t=1}^T(\sum_{j=1}^n r_{jt}x_j)u_t \\ & = \min_{\mathbf{u}\in U} D(\mathbf{u}) \end{aligned} \tag{10}$$

with the inner optimization problem

$$\begin{aligned} D(\mathbf{u}) &= \max_{x_j} && \{\sum_{t=1}^{T} u_t \sum_{j=1}^{n} r_{jt} x_j : \\ &&& \sum_{j=1}^{n} \mu_j x_j \geq \mu_0,\ \sum_{j=1}^{n} x_j = 1, \\ &&& x_j \geq 0,\ j = 1, \ldots, n\} \\ &= \max_{x_j} && \{\sum_{j=1}^{n} (\sum_{t=1}^{T} r_{jt} u_t) x_j : \\ &&& \sum_{j=1}^{n} \mu_j x_j \geq \mu_0,\ \sum_{j=1}^{n} x_j = 1, \\ &&& x_j \geq 0,\ j = 1, \ldots, n\}. \end{aligned} \tag{11}$$

Again, we may take advantages of the LP dual to the inner problem. Indeed, introducing dual variable q corresponding to the equation $\sum_{j=1}^{n} x_j = 1$ and variable u_0 corresponding to the inequality $\sum_{j=1}^{n} \mu_j x_j \geq \mu_0$ we get the LP dual

$$\begin{aligned} D(\mathbf{u}) = \min_{q, u_0} \quad & \{q - \mu_0 u_0 : \\ & q - \mu_j u_0 - \sum_{t=1}^{T} r_{jt} u_t \geq 0 \quad j = 1, \ldots, n\}. \end{aligned}$$

Hence, an alternative model for the CVaR portfolio optimization (10) can be expressed as the following LP:

$$\begin{aligned} \min \quad & q - \mu_0 u_0 \\ \text{s.t.} \quad & q - \mu_j u_0 - \sum_{t=1}^{T} r_{jt} u_t \geq 0,\ j = 1, \ldots, n \\ & \sum_{t=1}^{T} u_t = 1 \\ & 0 \leq u_t \leq \frac{p_t}{\beta},\ t = 1, \ldots, T \end{aligned} \tag{12}$$

LP model (12) contains $T + 1$ variables u_t, but the T constraints corresponding to variables d_t from (6) take the form of simple upper bounds on u_t (for $t = 1, \ldots, T$) thus not affecting the problem complexity. The number of constraints in (12) is proportional to the total of portfolio size n, thus it is independent from the number of scenarios. Exactly, there are $T + 1$ variables and $n + 1$ constraints. This guarantees a high computational efficiency of the model even for very large number of scenarios. Similarly, other portfolio structure requirements are modeled with rather small number of constraints thus generating small number of additional variables in the model. Actually, the model (12) is the LP dual to the model (6), thus similar to that introduced in [23]. Obviously, the optimal portfolio shares x_j are not directly represented within the solution vector of problem (12) but they are easily available as the dual variables (shadow prices) for inequalities $q - \mu_j u_0 - \sum_{t=1}^{T} r_{jt} u_t \geq 0$.

The Minimax portfolio optimization model can be written as the following LP problem:

$$\begin{aligned} \max \quad & \eta \\ \text{s.t.} \quad & \sum_{j=1}^{n} \mu_j x_j \geq \mu_0, \\ & \sum_{j=1}^{n} x_j = 1 \\ & x_j \geq 0,\ j = 1, \ldots, n \\ & -\eta + \sum_{j=1}^{n} r_{jt} x_j \geq 0,\ t = 1, \ldots, T \end{aligned} \tag{13}$$

which is simpler than the standard CVaR optimization model (6). Except from the portfolio weights x_j, the model contains only one additional variable η. Nevertheless, it still contains T linear inequalities in addition to the core constraints. Hence, its dimensionality is $(T + 2) \times (n + 1)$.

The Minimax portfolio optimization model representing a limiting case of the CVaR model for β tending to 0. Actually, for any $\beta \leq \min_{t=1,\ldots,T} p_t$ we gets $M_\beta(\mathbf{x}) = M(\mathbf{x})$ thus allowing to represent the Minimax portfolio optimization by the CVaR optimization model (6) and to take advantages of its dual form (12). Due to $\beta \leq p_t$ for all $t = 1, \ldots, T$, the upper bounds on variables u_t becomes redundant and we get the following dual form of the Minimax portfolio optimization:

$$\begin{aligned} \min \quad & q - \mu_0 u_0 \\ \text{s.t.} \quad & q - \mu_j u_0 - \sum_{t=1}^{T} r_{jt} u_t \geq 0,\ j = 1, \ldots, n \\ & \sum_{t=1}^{T} u_t = 1 \\ & u_t \ \geq 0,\ t = 1, \ldots, T \end{aligned} \tag{14}$$

The model dimensionality is only $(n + 1) \times (T + 2)$ thus guaranteeing a high computational efficiency even for very large number of scenarios.

The Mean Absolute Deviation (MAD) risk measure is directly given by the value of the second order cdf $F_{\mathbf{x}}^{(2)}$ at the mean $\bar{\delta}(\mathbf{x}) = \mathbf{E}\{\max\{\mu(\mathbf{x}) - R_{\mathbf{x}}, 0\}\} = F_{\mathbf{x}}^{(2)}(\mu(\mathbf{x}))$ [19]. Therefore, its leads to an LP portfolio optimization model very similar to that for the CVaR optimization (6). Indeed, we get:

$$\begin{aligned} \max \quad & -\sum_{t=1}^{T} p_t d_t \\ \text{s.t.} \quad & \sum_{j=1}^{n} \mu_j x_j \geq \mu_0, \quad \sum_{j=1}^{n} x_j = 1 \\ & d_t \geq \sum_{j=1}^{n} (\mu_j - r_{jt}) x_j,\ d_t \geq 0,\ t = 1, \ldots, T \\ & x_j \geq 0,\ j = 1, \ldots, n \end{aligned} \tag{15}$$

with $T + n$ variables and $T + 2$ constraints. The LP dual model

TABLE I
COMPUTATIONAL TIMES (IN SECONDS) FOR THE STANDARD MEAN-RISK MODELS

Scenarios	Securities	CVaR (6)						Minimax	MAD
(T)	(n)	$\beta = 0.05$	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$	(13)	(15)
5 000	76	2.5	2.6	3.7	5.2	6.7	7.6	0.5	19.0
7 000	76	4.1	4.4	6.9	9.3	11.8	14.8	0.9	9.3
10 000	76	6.5	8.3	13.9	18.4	24.2	29.8	1.3	18.4
50 000	50	3275.4	4876.6	–	–	–	–	7.8	–
50 000	100	–	–	–	–	–	–	24.1	–

TABLE II
COMPUTATIONAL TIMES (IN SECONDS) FOR THE DUAL MEAN-RISK MODELS

Scenarios	Securities	CVaR (12)						Minimax	MAD
(T)	(n)	$\beta = 0.05$	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$	(14)	(16)
5 000	76	0.9	0.9	1.1	1.3	1.4	1.6	0.5	2.1
7 000	76	1.2	1.4	1.8	2.0	2.2	2.3	0.7	3.1
10 000	76	2.0	2.3	2.9	3.4	3.9	4.0	1.0	10.8
50 000	50	14.9	19.4	24.0	26.8	28.2	27.9	3.9	25.8
50 000	100	40.0	54.6	68.7	77.7	78.2	78.8	8.2	76.7

takes then the form:

$$\begin{array}{ll} \min & q - \mu_0 u_0 \\ \text{s.t.} & q \geq \mu_j u_0 - \sum_{t=1}^{T} (\mu_j - r_{jt}) u_t, \ j = 1, \ldots, n \\ & 0 \leq u_t \ \leq p_t, \ t = 1, \ldots, T \end{array} \tag{16}$$

with dimensionality $n \times (T + 2)$. Hence, there is again guaranteed the high computational efficiency even for very large number of scenarios.

We have run two groups of computational tests. The medium scale tests of 5 000, 7 000 and 10 000 scenarios and 76 securities were generated following the FTSE 100 related data [5]. The large scale tests instances developed by Lim et al. [9] were generated from a multivariate normal distribution for 50 or 100 securities with the number of scenarios 50 000 just providing an adequate approximation to the underlying unknown continuous price distribution. When applying the lower bound on the required expected return, its value μ_0 was defined as the expected return of the portfolio with equal weights (market value). All computations were performed on a PC with the Pentium 4 2.6GHz processor and 3GB RAM employing the simplex code of the CPLEX 9.1 package.

In Tables I and II there are presented computation times for all the above primal and dual models. All results are presented as the averages of 10 different test instances of the same size. For the medium scale test problems the solution times of the dual CVaR models (12) ranging from 0.9 to 4.0 seconds are not much shorter than those for the primal models ranging from 2.5 to 29.8 seconds, respectively. However, an attempt to solve the primal CVaR model (6) of the large scale test problems was successful only for $\beta = 0.05$ and $\beta = 0.1$ and the times were dramatically longer than those for the dual model. For other values of β the timeout of 6000 seconds occurred (marked with '–'). For 100 securities the primal model was not solvable within the given time limit, while the dual models could be successfully solved in 40.0 to 78.8 seconds.

The Minimax models are computationally very easy. Running the computational tests we were able to solve the medium scale test instances of the dual model (14) in times below 1 second and the large scale test instances in up to 8.2 seconds on average. In fact, even the primal model could be solved in reasonable time between 0.5 and 24 seconds, for medium and large scale test instances, respectively.

The MAD models are computationally similar to the CVaR models. Indeed, only medium scale test instances of the primal model (15) could be solved within the given time limit. Much shorter computing times could be achieved for the dual MAD model (16) – not more than 10.8 seconds for the medium scale and 76.7 seconds for the large scale test instances.

To see how the value of the required expected return affects the solution times we have performed additional tests for the reformulated CVaR (with $\beta = 0.1$) and Minimax models with increased value μ_0. The increased value was set in the middle between the expected return of the market value and the maximum possible return for single security portfolio. The computational times are generally comparable with those for the market value constraints. Actually, for the medium scale problems (10 000 scenarios) the CVaR optimization time has remained unchanged (2.3 seconds) while the Minimax optimization time has only raised from 1.0 to 1.1 seconds and for the MAD model optimization it has raised from 10.8 to 13.9. For the large scale problems (50 000 scenarios) we have noticed even a drop in computation times for the CVaR model (reduction from 54.6 to 49.9 seconds) and similar reduction (from 76.7 to 55.4 seconds) has occured for the MAD model optimization while the Minimax optimization time has increased from 8.2 to 10.5 seconds.

IV. GINI'S MEAN DIFFERENCE AND RELATED MODELS

Yitzhaki [32] introduced the portfolio optimization model using Gini's mean difference (GMD) as risk measure. The GMD is given as $\Gamma(\mathbf{x}) = \frac{1}{2} \int \int |\eta - \xi| dF_{\mathbf{x}}(\eta) dF_{\mathbf{x}}(\xi)$ although several alternative formulae exist. For a discrete random variable represented by its realizations y_t, the measure

$$\Gamma(\mathbf{x}) = \sum_{t'=1}^{T} \sum_{t'' \neq t'-1} \max\{y_{t'} - y_{t''}, 0\} p_{t'} p_{t''}$$

is LP computable (when minimized) leading to the following portfolio optimization model:

$$\begin{array}{ll} \max & -\sum_{t=1}^{T} \sum_{t' \neq t} p_t p_{t'} d_{tt'} \\ \text{s.t.} & \sum_{j=1}^{n} \mu_j x_j \geq \mu_0, \quad \sum_{j=1}^{n} x_j = 1 \\ & d_{tt'} \geq \sum_{j=1}^{n} r_{jt} x_j - \sum_{j=1}^{n} r_{jt'} x_j \\ & d_{tt'} \geq 0, \ t \neq t' = 1, \dots, T \\ & x_j \geq 0, \ j = 1, \dots, n \end{array} \tag{17}$$

which contains $T(T-1)$ nonnegative variables $d_{tt'}$ and $T(T-1)$ inequalities to define them. This generates a huge LP problem even for the historical data case where the number of scenarios is 100 or 200. Krzemienowski and Ogryczak [7] have shown with the earlier experiments that the CPU time of 7 seconds on average for $T = 52$ has increased to above 30 sec. with $T = 104$ and even more than 180 sec. for $T = 156$. However, similar to the CVaR models, variables $d_{tt'}$ are associated with the singleton coefficient columns. Hence, while solving the dual instead of the original primal, the corresponding dual constraints take the form of simple upper bounds (SUB) which are handled implicitly outside the LP matrix. For the simplest form of the feasible set (1) the dual GMD model takes the following form:

$$\begin{array}{l} \min q - \mu_0 u_0 \\ q \geq \mu_j u_0 + \sum_{t=1}^{T} \sum_{t' \neq t} (r_{jt} - r_{jt'}) u_{tt'}, \ j = 1, \dots, n \\ 0 \leq u_{tt'} \ \leq p_t p_{t'}, \ t, t' = 1, \dots, T; t \neq t' \end{array} \tag{18}$$

where original portfolio variables x_j are dual prices to the inequalities. The dual model contains $T(T-1)$ variables $u_{tt'}$ but the number of constraints (excluding the SUB structure) $n+1$ is proportional to the number of securities. The above dual formulation can be further simplified by introducing variables:

$$\bar{u}_{tt'} = u_{tt'} - u_{t't}, \quad t, t' = 1, \dots, T; t < t' \tag{19}$$

which allows us to reduce the number of variables to $T(T-1)/2$ by replacing (18) with the following:

$$\begin{array}{l} \min q - \mu_0 u_0 \\ q \geq \mu_j u_0 + \sum_{t=1}^{T} \sum_{t' > t} (r_{jt} - r_{jt'}) \bar{u}_{tt'}, \ j = 1, \dots, n \\ -p_t p_{t'} \leq \bar{u}_{tt'} \ \leq p_t p_{t'}, \ t < t' = 1, \dots, T \end{array} \tag{20}$$

Such a dual approach may dramatically improve the LP model efficiency in the case of larger number of scenarios. Actually, as shown with the earlier experiments of [7], the above dual formulations let us to reduce the optimization time below 10 seconds for $T = 104$ and $T = 156$. Nevertheless, the case of really large number of scenarios still may cause computational difficulties, due to huge number of variables ($T(T-1)/2$). This may require some column generation techniques [3] or nondifferentiable optimization algorithms [9].

As shown by Yitzhaki [32] for the SSD consistency of the GMD model one needs to maximize the complementary measure

$$\mu_\Gamma(\mathbf{x}) = \mu(\mathbf{x}) - \Gamma(\mathbf{x}) = \mathbf{E}\{R_\mathbf{x} \wedge R_\mathbf{x}\} \tag{21}$$

where the cumulative distribution function of $R_\mathbf{x} \wedge R_\mathbf{x}$ for any $\eta \in \mathbf{R}$ is given as $F_\mathbf{x}(\eta)(2 - F_\mathbf{x}(\eta))$. Hence, (21) is the expectation of the minimum of two independent identically distributed random variables (i.i.d.r.v.) $R_\mathbf{x}$ thus representing the *mean worse return*. This provides us with another LP model although it is not more compact than that of (17) and its dual (18). Alternatively, the GMD may be expressed with integral of the absolute Lorenz curve as

$$\begin{aligned} \Gamma(\mathbf{x}) &= 2 \int_0^1 (\alpha \mu(\mathbf{x}) - F_\mathbf{x}^{(-2)}(\alpha)) d\alpha \\ &= 2 \int_0^1 \alpha (\mu(\mathbf{x}) - M_\alpha(\mathbf{x})) d\alpha \end{aligned}$$

and respectively

$$\mu_\Gamma(\mathbf{x}) = 2 \int_0^1 F_\mathbf{x}^{(-2)}(\alpha) d\alpha = 2 \int_0^1 \alpha M_\alpha(\mathbf{x}) d\alpha \tag{22}$$

thus combining all the CVaR measures. In order to enrich the modeling capabilities, one may treat differently some more or less extreme events. In order to model downside risk aversion, instead of the Gini's mean difference, the *tail Gini's* measure introduced by Ogryczak and Ruszczyński [21], [20] can be used:

$$\begin{aligned} \mu_{\Gamma_\beta}(\mathbf{x}) &= \mu(\mathbf{x}) - \frac{2}{\beta^2} \int_0^\beta (\mu(\mathbf{x})\alpha - F_\mathbf{x}^{(-2)}(\alpha)) d\alpha \\ &= \frac{2}{\beta^2} \int_0^\beta F_\mathbf{x}^{(-2)}(\alpha) d\alpha \end{aligned} \tag{23}$$

In the simplest case of equally probable T scenarios with $p_t = 1/T$ (historical data for T periods), the tail Gini's measure for $\beta = K/T$ may be expressed as the weighted combination of CVaRs $M_{\beta_k}(\mathbf{x})$ with tolerance levels $\beta_k = k/T$ for $k = 1, 2, \dots, K$ and properly defined weights [21]. In a general case, we may resort to an approximation based on some reasonably chosen grid β_k, $k = 1, \dots, m$ and weights w_k expressing the corresponding trapezoidal approximation of the integral in the formula (23). Exactly, for any $0 < \beta \leq 1$, while using the grid of m tolerance levels $0 < \beta_1 < \dots < \beta_k < \dots < \beta_m = \beta$ one may define weights:

$$\begin{aligned} w_k &= \frac{(\beta_{k+1} - \beta_{k-1})\beta_k}{\beta^2}, \ k = 1, \dots, m-1 \\ w_m &= \frac{\beta - \beta_{m-1}}{\beta} \end{aligned} \tag{24}$$

where $\beta_0 = 0$. This leads us to the Weighted CVaR (WCVaR) measure [12] defined as

$$\begin{aligned} M_{\mathbf{w}}^{(m)}(\mathbf{x}) &= \sum_{k=1}^{m} w_k M_{\beta_k}(\mathbf{x}) \\ \sum_{k=1}^{m} w_k &= 1, \quad w_k > 0,\ k = 1,\dots,m \end{aligned} \tag{25}$$

We emphasize that despite being only an approximation to (23), any WCVaR measure itself is a well defined LP computable measure with guaranteed SSD consistency and coherency, as a combination of the CVaR measures. Hence, it needs not to be built on a very dense grid to provide proper modeling of risk averse preferences. While analyzed on the real-life data from the Milan Stock Exchange the weighted CVaR models have usually performed better than the GMD itself, the Minimax or the extremal CVaR models [12].

Here we analyze only computational efficiency of the LP models representing the WCVaR portfolio optimization. For returns represented by their realizations we get the following LP optimization problem:

$$\begin{aligned} \max\quad & \sum_{k=1}^{m} w_k \eta_k - \sum_{k=1}^{m} \frac{w_k}{\beta_k} \sum_{t=1}^{T} p_t d_{tk} \\ \text{s.t.}\quad & \sum_{j=1}^{n} \mu_j x_j \geq \mu_0, \quad \sum_{j=1}^{n} x_j = 1 \\ & x_j \geq 0,\ j = 1,\dots,n \\ & d_{tk} \geq \eta_k - \sum_{j=1}^{n} r_{jt} x_j \\ & d_{tk} \geq 0,\ t = 1,\dots,T; k = 1,\dots,m \end{aligned} \tag{26}$$

where η_k (for $k = 1,\dots,m$) are unbounded variables taking the values of the corresponding β_k-quantiles (in the optimal solution). The problem dimensionality is proportional to the number of scenarios T and to the number of tolerance levels m. Exactly, the LP model contains $m \times T + n$ variables and $m \times T + 2$ constraints. The LP problem structure is similar to that of reperesenting the so-called WOWA optimization in fuzzy approaches [22]. It does not cause any computational difficulties for a few hundreds scenarios and a few tolerance levels, as in a simple computational analysis based on historical data [12]. However, in the case of more advanced simulation models employed for scenario generation one may get several thousands scenarios. This may lead to the LP model (26) with huge number of variables and constraints thus decreasing the computational efficiency of the model. If the core portfolio constraints contain only linear relations, like (1), then the computational efficiency can easily be achieved by taking advantages of the LP dual to model (26). The LP dual model takes the following form:

$$\begin{aligned} \min\quad & q - \mu_0 u_0 \\ \text{s.t.}\quad & q - \mu_j u_0 - \sum_{t=1}^{T} r_{jt} \sum_{k=1}^{m} u_{tk} \geq 0,\ j = 1,\dots,n \\ & \sum_{t=1}^{T} u_{tk} = w_k \\ & 0 \leq u_{tk} \leq \frac{p_t w_k}{\beta_k},\ t = 1,\dots,T; k = 1,\dots,m \end{aligned} \tag{27}$$

that contains $m \times T$ variables u_{tk}, but the $m \times T$ constraints corresponding to variables d_{tk} from (26) take the form of simple upper bounds on u_{tk} thus not affecting the problem complexity. Hence, again the number of constraints in (27) is proportional to the total of portfolio size n and the number of tolerance levels m, thus it is independent from the number of scenarios. Exactly, there are $m \times T + 2$ variables and $m + n$ constraints thus guaranteeing a high computational efficiency of for very large number of scenarios.

TABLE III
COMPUTATIONAL TIMES (IN SECONDS) FOR THE DUAL WCVAR MODELS

Scenarios	Securities	Model (27)	
(T)	(n)	$m = 3$	$m = 5$
5 000	76	6.2	13.8
7 000	76	10.0	22.8
10 000	76	16.2	37.7
50 000	50	121.8	281.0
50 000	100	335.3	731.7

We have tested computational efficiency of the dual model (27) using the same randomly generated test instances as for testing of the CVaR and other basic models in Section III. Table III presents average computation times of the dual models for $m = 3$ with tolerance levels $\beta_1 = 0.1$, $\beta_2 = 0.25$, $\beta_3 = 0.5$ and weights $w_1 = 0.1$, $w_2 = 0.4$ and $w_3 = 0.5$, thus representing the parameters leading to good results on real life data [12], as well as for $m = 5$ with uniformly distributed tolerance levels $\beta_1 = 0.1$, $\beta_2 = 0.2$, $\beta_3 = 0.3$, $\beta_4 = 0.4$, $\beta_5 = 0.5$ and weights (24).

V. CONCLUDING REMARKS

The classical Markowitz model uses the variance as the risk measure, thus resulting in a quadratic optimization problem. There were introduced several alternative risk measures which are computationally attractive as (for discrete random variables) they result in solving linear programming (LP) problems. The LP solvability is very important for applications to real-life financial decisions where the constructed portfolios have to meet numerous side constraints and take into account transaction costs. A gamut of LP computable risk measures has been presented in the portfolio optimization literature although most of them are related to the absolute Lorenz curve and thereby the CVaR measures. We have shown that all the risk measures used in the LP solvable portfolio optimization models can be derived from the SSD shortfall criteria. This allows us to guarantee their SSD consistency for any distribution of outcomes.

The corresponding portfolio optimization models can be solved with general purpose LP solvers. However, in the case of more advanced simulation models employed for scenario generation one may get several thousands of scenarios. This may lead to the LP model with huge number of variables and constraints thus decreasing the computational efficiency of the models. For the CVaR model, the number of constraints (matrix rows) is proportional to the number of scenarios. while the number of variables (matrix columns) is proportional to the total of the number of scenarios and the number of instruments. We have shown that the computational efficiency can be then dramatically improved with an alternative model taking advantages of the LP duality. In the introduced model the number of structural constraints (matrix rows) is proportional to the number of instruments thus not affecting seriously the simplex method efficiency by the number of scenarios. In particular, for the case of 50 000 scenarios, it has resulted in computation times below 30 seconds for 50 securities or below a minute for 100 instruments. Similar computational times have also been achieved for the reformulated Minimax model.

Similar reformulation can be developed for other LP computable portfolio optimization models as many of them are related to the Absolute Lorenz Curve [18], [10]. In particular, this applies to the classical LP portfolio optimization models based on the mean absolute deviation as well as to the Gini's mean difference. The standard LP models for the Gini's mean difference [32] and its downside version [7] require T^2 auxiliary constraints which makes them hard already for medium numbers of scenarios, like a few hundred scenarios given by historical data. The models taking advantages of the LP duality allow one to limit the number of structural constraints making it proportional to the number of scenarios T thus increasing dramatically computational performances for medium numbers of scenario although still remaining hard for very large numbers of scenarios.

Acknowledgment

The authors are indebted to Churlzu Lim from the University of North Carolina at Charlotte for providing the test data of 50 000 scenarios portfolio selection problem and to Diana Roman from CARISMA, Brunel University for providing the FTSE 100 related test data of 5 000, 7 000 and 10 000 scenarios we have used in our computational studies.

References

[1] Andersson, F, H Mausser, D Rosen and S Uryasev, (2001), Credit risk optimization with conditional value-at-risk criterion, *Mathematical Programming* 89, 273–291.

[2] Artzner, P, F Delbaen, J-M Eber and D Heath, (1999), Coherent measures of risk, *Mathematical Finance* 9, 203–228.

[3] Desrosiers, J and M Luebbecke (2005), A primer in column generation, in: G Desaulniers, J Desrosier, and M Solomon, eds.: *Column Generation*, Springer, New York, 132.

[4] Embrechts, P, C Klüppelberg and T Mikosch (1997), *Modelling Extremal Events for Insurance and Finance*, Springer, New York.

[5] Fabian, C I, G Mitra, D Roman and V Zverovich (2010), An enhanced model for portfolio choice with SSD criteria: a constructive approach, *Quantitative Finance*, forthcoming, DOI: 10.1080/14697680903493607.

[6] Konno, H and H Yamazaki (1991), Mean-absolute deviation portfolio optimization model and its application to Tokyo stock market, *Management Science* 37, 519–531.

[7] Krzemienowski, A and W Ogryczak (2005), On extending the LP computable risk measures to account downside risk, *Computational Optimization and Applications* 32, 133–160.

[8] Künzi-Bay, A and J Mayer (2006). Computational aspects of minimizing conditional value-at-risk. *Computational Management Science*, 3, 3–27.

[9] Lim, C, H D Sherali and S Uryasev (2010), Portfolio optimization by minimizing conditional value-at-risk via nondifferentiable optimization, *Computational Optimization and Applications* 46, 391–415

[10] Mansini, R, W Ogryczak and M G Speranza (2003), On LP solvable models for portfolio selection, *Informatica* 14, 37–62.

[11] Mansini, R, W Ogryczak and M G Speranza (2003), LP solvable models for portfolio optimization: A classification and computational comparison, *IMA Journal of Management Mathematics* 14, 187–220.

[12] Mansini, R, W Ogryczak and M G Speranza (2007), Conditional value at risk and related linear programming models for portfolio optimization, *Annals of Operations Research* 152, 227–256.

[13] Markowitz, H M (1952), Portfolio selection, *Journal of Finance* 7, 77–91.

[14] Miller, N and A Ruszczyński (2008). Risk-adjusted probability measures in portfolio optimization with coherent measures of risk. *European Journal of Operational Research*, 191, 193–206.

[15] Müller, A and D Stoyan (2002), *Comparison Methods for Stochastic Models and Risks*, Wiley, Chichester.

[16] Ogryczak, W (1999), Stochastic dominance relation and linear risk measures, in A M J Skulimowski, ed.: *Financial Modelling – Proc. 23rd Meeting EURO WG Financial Modelling, Cracow, 1998*, Progress & Business Publisher, Cracow, 191–212.

[17] Ogryczak, W (2000), Risk measurement: Mean absolute deviation versus Ginis mean difference, in W G Wanka, ed.: *Decision Theory and Optimization in Theory and Practice – Proc. 9th Workshop GOR WG, Chemnitz, 1999*, Shaker Verlag, Aachen, 33–51.

[18] Ogryczak, W (2000), Multiple criteria linear programming model for portfolio selection, *Annals of Operations Research* 97, 143–162.

[19] Ogryczak, W and A Ruszczyński (1999), From stochastic dominance to mean-risk models: semideviations as risk measures, *European Journal of Operational Research* 116, 33–50.

[20] Ogryczak, W and A Ruszczyński (2002), Dual stochastic dominance and related mean-risk models, *SIAM J. Optimization* 13, 60–78.

[21] Ogryczak, W and A Ruszczyński (2002), Dual Stochastic dominance and quantile risk measures, *International Transactions in Operational Research* 9, 661–680.

[22] Ogryczak W and T Śliwiński (2009), On efficient WOWA optimization for decision support under risk, *International Journal of Approximate Reasoning* 50, 915–928.

[23] Ogryczak W and T Śliwiński (2010), On solving the dual for portfolio selection by optimizing Conditional Value at Risk, *Computational Optimization and Applications*, forthcoming, DOI 10.1007/s10589-010-9321-y.

[24] Pflug, G Ch (2000), Some remarks on the value-at-risk and the conditional value-at-risk, in S Uryasev, ed.: *Probabilistic Constrained Optimization: Methodology and Applications*, Kluwer AP, Dordrecht.

[25] Pflug, G Ch (2001), Scenario tree generation for multiperiod financial optimization by optimal discretization, *Mathematical Programming* 89, 251-271.

[26] Quiggin, J (1982), A theory of anticipated utility, *Journal of Economic Behavior and Organization* 3, 323–343.

[27] Rockafellar, R T and S Uryasev (2000), Optimization of conditional value-at-risk, *Journal of Risk* 2, 21–41.

[28] Roell, A (1987), Risk aversion in Quiggin and Yaari's rank-order model of choice under uncertainty, *Economic Journal* 97, 143–159.

[29] Rothschild, M and J E Stiglitz (1969), Increasing risk: I. A definition, *Journal of Economic Theory* 2, 225–243.

[30] Shorrocks, A F (1983), Ranking income distributions, *Economica* 50, 3–17.

[31] Yaari, M E (1987), The dual theory of choice under risk, *Econometrica* 55, 95–115.

[32] Yitzhaki, S (1982), Stochastic dominance, mean variance, and Gini's mean difference, *The American Economic Revue* 72, 178–185.

[33] Young, M R (1998), A minimax portfolio selection rule with linear programming solution, *Management Science* 44, 673–683.

ISBN 978-83-60810-27-9
ISSN 1896-7094

Enhanced Competitive Differential Evolution for Constrained Optimization

Josef Tvrdík
University of Ostrava
Department of Computer Science
Ostrava, Czech Republic
Email: josef.tvrdik@osu.cz

Radka Poláková
University of Ostrava
Department of Mathematics
Ostrava, Czech Republic
Email: radka.polakova@wo.cz

***Abstract*—The constrained optimization with differential evolution (DE) is addressed. A novel variant of competitive differential evolution with a hybridized search of feasibility region is proposed, where opposition-based optimization and adaptive controlled random search are combined. Various variants of the algorithm are experimentally compared on the benchmark set developed for the special session of IEEE Congress of Evolutionary Computation (CEC) 2010. The results of the enhanced competitive DE show effective search of feasible solutions, in difficult problems significantly better than the competitive DE variant presented at CEC 2010.**

I. Introduction

DIFFERENTIAL evolution (DE) was proposed by Storn and Price [1] as a global optimizer for continuous optimization problems with a real-value objective function, where only the search space (domain) S is specified by lower (a_j) and upper (b_j) limits of each component j, $S = \prod_{j=1}^{D}[a_j, b_j]$, $a_j < b_j$, $j = 1, 2, \ldots, D$, D is the dimension of the domain. The global minimum point $\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \in S} f(\boldsymbol{x})$ is the solution of the problem.

DE has been studied intensively in recent years and many new variants of DE have been proposed. The research has mostly focused on proper settings of control parameters and their adaptation during the search process. However, variants of DE for solving discrete or constrained problems have also appeared, for overview see e.g. [2].

The paper is organized as follows. In Section II, the constrained optimization problem is defined and sevral stochastic algorithms for its solution are mentioned. The DE algorithm is described in Section III, the description is focused on important features of the competitive DE used in this study. Techniques of enhanced search of the feasibility region are summarized and all algorithms tested in this study are described in Section IV. Section V involves specification of experiments and the results of comparison of the feasibility rates obtained by all the competitive DE variants in the tests as well as detailed results of the best performing algorithm. Concluding remarks are made in Section VI.

II. Constrained Optimization

We consider the optimization problem in the following format [3]:

$$\textit{Minimize: } f(\boldsymbol{x}),\ \boldsymbol{x} = (x_1, x_2, \ldots, x_D) \textit{ and } \boldsymbol{x} \in S \qquad (1)$$

$$\textit{subject to: } \begin{array}{ll} g_i(\boldsymbol{x}) \leq 0, & i = 1, \ldots, p \\ h_j(\boldsymbol{x}) = 0, & j = p+1, \ldots, m. \end{array}$$

A solution is regarded feasible if $g_i(\boldsymbol{x}) \leq 0,\ \textit{for}\ i = 1, \ldots, p$, and $|h_j(\boldsymbol{x})| - \varepsilon \leq 0,\ \textit{for}\ j = p+1, \ldots, m$. Mean violation $\bar{v}$ of any point $\boldsymbol{x}$ in population defined according to [3] can be evaluated

$$\bar{v} = \frac{\sum_{i=1}^{p} G_i(\boldsymbol{x}) + \sum_{j=p+1}^{m} H_j(\boldsymbol{x})}{m},$$

where

$$G_i(\boldsymbol{x}) = \begin{cases} g_i(\boldsymbol{x}) & \text{if} \quad g_i(\boldsymbol{x}) > 0 \\ 0 & \text{if} \quad g_i(\boldsymbol{x}) \leq 0 \end{cases}$$

$$H_j(\boldsymbol{x}) = \begin{cases} |h_j(\boldsymbol{x})| & \text{if} \quad |h_j(\boldsymbol{x})| - \varepsilon > 0 \\ 0 & \text{if} \quad |h_j(\boldsymbol{x})| - \varepsilon \leq 0. \end{cases}$$

Some versions of DE for constrained problems have been published recently. One of the first modifications of DE for constrained problems was published by Lampinen [4]. In [5], the authors proposed self-adaptive algorithm ε-jDE which is a modification of jDE algorithm [6] extended for constrained optimization. They presented ε level controlling. The ε level is updated according to the mean violation of individuals in the current generation until the number of generations G reaches the control generation G_C. After the number of generations exceeds G_C, the ε level is set to 0 to obtain a solution with minimum constraint violation.

An up-to-date overview of evolutionary techniques for constrained problems is presented in [7]. The authors also proposed a novel algorithm using an ensemble of constraint handling techniques (ECHT). The included techniques are superiority of feasible solution, self-adaptive penalty, ε-constraint, and stochastic ranking. In the algorithm which uses ECHT together with differential evolution (ECHT-DE) each constraint handling method has its own population and new trial points are computed separately for each constraint handling method in the current generation.

This work was supported by Czech Ministry of Education, the research project MSM 6198898701.

A simple DE algorithm for constrained problem was presented in [8]. This algorithm alternates minimization of mean violation $\bar{v}$ and $f(\boldsymbol{x})$ using one generation of the adaptive DE algorithm with 12 competing DE strategies, where various values of control parameters and two types of crossover are used. The algorithm performed well on most problems of the benchmark test set [3]. However, in a few problems the algorithm found feasible solution rarely or even never, see counts of feasible solutions in Tables II and III. This failure in search of the feasible region is a motivation for the proposal of a new algorithm with enhanced search of the feasible region.

III. Differential Evolution

Algorithm of DE works with a population of individuals (*NP* points in domain S) that are considered as candidates of solution. The population develops iteratively by using evolutionary operators of selection, mutation, and crossover. Each iteration corresponds to an evolutionary generation. Let us denote two subsequent generations P and Q. Application of evolutionary operators in the old generations P creates a new generation Q. After completing the new generation Q, the Q becomes the old generation for next iteration. The basic scheme of DE is written in a pseudo-code in Figure 1.

```
1   generate an initial generation $P$, $(\boldsymbol{x}_i,\ i = 1, 2, \dots, NP)$
2   while stopping condition not achieved
3       for $i$ := 1 to $NP$ do
4           generate a new trial vector $\boldsymbol{y}$ using $P$
5           if $\boldsymbol{y}$ better $\boldsymbol{x}_i$ then insert $\boldsymbol{y}$ into $Q$
6                                 else insert $\boldsymbol{x}_i$ into $Q$
7           endif
8       endfor
9       $P := Q$
10  endwhile
```

Fig. 1. **Algorithm 1** – Basic scheme of DE in pseudo-code

The relation "better" (line 5 in Algorithm 1) means $f(\boldsymbol{y}) < f(\boldsymbol{x}_i)$ in unconstrained problems. In constrained problems, the preference of a more feasible solution is usually applied according to the following rule: Accept a new trial point $\boldsymbol{y}$ if

$$\bar{v}_y < \bar{v}_i \text{ or } (\bar{v}_y = 0 \text{ and } \bar{v}_i = 0 \text{ and } f(\boldsymbol{y}) < f(\boldsymbol{x}_i)), \qquad (2)$$

where $\bar{v}_i$ is mean violation of the current point $\boldsymbol{x}_i$.

A new trial point $\boldsymbol{y}$ (line 4 in Algorithm 1) is generated by using mutation and crossover. There are various strategies of mutation and crossover [1], [2], [9]. The most popular mutation strategy called DE/rand/1/ generates the mutant point $\boldsymbol{u}$ by adding the weighted difference of two points,

$$\boldsymbol{u} = \boldsymbol{r}_1 + F\,(\boldsymbol{r}_2 - \boldsymbol{r}_3)\,, \quad F > 0\,, \qquad (3)$$

where $\boldsymbol{r}_1, \boldsymbol{r}_2$, and $\boldsymbol{r}_3$ are three mutually distinct points randomly taken from population P, not coinciding with the current point $\boldsymbol{x}_i$, and F is an input parameter. In the implementation of the DE algorithm in this paper is used a slightly modified mutation strategy. It was proposed by Kaelo and Ali [10]. They called it random localization and it can be denoted as DE/randrl/1/. The point $\boldsymbol{r}_1$ in (3) is the best one among $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, and $\boldsymbol{r}_3$.

The elements y_j, $j = 1, 2, \dots, D$, of the trial point $\boldsymbol{y}$ are built up by the crossover of the current point $\boldsymbol{x}_i$ and the mutant point $\boldsymbol{u}$. The most frequently used kind of crossover is called *binomial*. It uses the following rule of combination of parents' elements

$$y_j = \begin{cases} u_j & \text{if} \quad U_j \le CR \quad \text{or} \quad j = l \\ x_{ij} & \text{if} \quad U_j > CR \quad \text{and} \quad j \ne l\,, \end{cases} \qquad (4)$$

where l is a randomly chosen integer from $\{1, 2, \dots, D\}$, $U_1, U_2, \dots, U_D$ are independent random variables uniformly distributed in $[0,\ 1)$, and $CR \in [0, 1]$ is an input parameter influencing the number of elements to be exchanged by the crossover. The rule given by Eq. (4) ensures that at least one element of vector $\boldsymbol{x}_i$ is changed, even if $CR = 0$. The DE variants applying binomial crossover according to (4) are denoted DE/·/·/bin in literature.

The *exponential* crossover denoted by abbreviation DE/·/·/exp was also proposed in the first version of DE by Price and Storn [1]. The exponential crossover in DE is similar to the two-point crossover in genetic algorithms. For the exponential crossover, the starting position of crossover $(k = 1)$ is randomly chosen from $\{1, \dots, D\}$, and L consecutive elements (counted in circular manner) are taken from the mutant vector $\boldsymbol{u}$. The probability of replacing the k-th element in the sequence $1, 2, \dots, L,\ \ L \le D$, decreases exponentially with increasing k.

The probability of crossover p_m can be defined as the mean relative length of overwritten elements of $\boldsymbol{x}_i$, i.e. $p_m = E(L)/D$. The relation between the p_m and control parameter *CR* was studied by Zaharie [11], [12]. For the binomial crossover, the relation between p_m and control parameter *CR* is linear,

$$p_m = CR(1 - 1/D) + 1/D, \qquad (5)$$

while for the exponential crossover the relationship is strongly non-linear,

$$p_m = \frac{1 - CR^D}{D(1 - CR)}, \quad \text{for} \quad CR < 1. \qquad (6)$$

This non-linearity should be taken into account when we set the *CR* value for a problem to be solved.

Differential evolution has a few control parameters, namely the size of population *NP*, mutation strategy, crossover type, and couple of parameters F and *CR*. However, the efficiency of differential evolution is very sensitive especially regarding the setting of F and *CR* values. The most suitable control-parameter values for a specific problem may be found by trial-and-error tuning, which requires a lot of time. There are some recommendations for setting of these parameters [1], [2], [9], [13], [14] but their applicability is not universal.

Because trial-and-error control-parameter tuning is time-consuming, several new adaptive variants of DE have been recently proposed, e.g. [6], [15]–[21].

Competitive setting of the control parameters was proposed in [22]. In this adaptive approach to the choice of proper DE strategy, we randomly choose among H different settings of control parameters (F, *CR*, mutation, and crossover) with probabilities $q_h,\ h = 1, 2, \ldots, H$.

These probabilities change according to the success rate of the settings in the preceding steps of the search process. The h-th setting is considered successful if it generates a trial point $\boldsymbol{y}$ better than its counter-partner $\boldsymbol{x}_i$ in the old generation P, see line 5 of Algorithm 1 in Figure 1. Probability q_h is evaluated as the relative frequency

$$q_h = \frac{n_h + n_0}{\sum_{j=1}^{H}(n_j + n_0)}\,, \tag{7}$$

where n_h is the current count of the h-th setting's successes, and $n_0 > 0$ is a constant. The input parameter $n_0 > 1$ prevents a dramatic change in q_h by one random successful use of the h-th parameter setting. To avoid degeneration of the search process, the current values of q_h are reset to their starting values $q_h = 1/H$ if any probability q_h decreases bellow the given limit $\delta > 0$ during the search process.

Mutation according to (3) could cause that a new trial point $\boldsymbol{y}$ moves out of the domain S. In such a case, the point $\boldsymbol{y}$ can be skipped and a new one generated. A more effective attempt is to replace the values of $y_j < a_j$ or $y_j > b_j$, either by random value in $[a_j,\, b_j]$, see [9], or by reversed value of this coordinate y_j into S over the a_j or b_j by mirroring, see [23]. The latter method is used in algorithms implemented for numerical tests.

TABLE I
VALUES OF CROSSOVER PROBABILITY AND THE CORRESPONDING VALUES OF *CR* PARAMETER FOR EXPONENTIAL CROSSOVER

	$D = 10$				$D = 30$		
i	1	2	3		1	2	3
p_{mi}	0.3250	0.5500	0.7750	p_i	0.2750	0.5167	0.7583
CR_i	0.7011	0.8571	0.9418	CR_i	0.8815	0.9488	0.9801

In several benchmark tests [24], [25], the competitive variant of DE/randrl/1/ with $H = 12$ competing settings was among the most efficient. In this variant, six various settings of (F, CR) for the binomial crossover and six settings for the exponential crossover take part in the competition. Two values of F are used for the both type of crossover, one of them $F = 0.5$ is rather small and the second one $F = 0.8$ is rather big when the recommendation in [1], [2], [9], [13], [14] are followed. Three values of *CR* for the binomial crossover are set to their minimum, middle, and maximum value, that is $\mathit{CR} = 0$, $\mathit{CR} = 0.5$, and $\mathit{CR} = 1$. All the three values are combined in tuples with two values of F, which gives six settings mentioned above. For the exponential crossover, three values of *CR* are used as well, but they are set in such a way in order to have the values of the crossover probability given by (6) equally spaced inside the interval of their possible values, $[1/D,\ 1]$. Notice, that for extremal values of the crossover probability $p_m = 1/D$ or $p_m = 1$ the both types of crossover do not differ. Both the mutation probability values and the corresponding *CR* values are dependent on the dimension of the problem. The values of mutation probability and the corresponding values of *CR* evaluated from (6) for $D = 10$ and $D = 30$ are given in Table I. Like in the case of the binomial crossover, three values of *CR* are combined with two values of F, which gives six settings of (F, CR) for the exponential crossover taking part in competition. The competitive DE variant using these 12 settings is hereafter referred as *b6e6rl*.

Thus, the *b6e6rl* algorithm modified for constrained problems has only the following control parameters:

- Size of the population, *NP* – can be intuitively set with respect to the dimension of the problem.
- Parameters n_0 and δ to control the competition – can be set to the values, $n_0 = 2$, $\delta = 1/(5 \times H)$ used in other problems.
- Stopping condition – can be formed by maximum of objective function evaluations (MaxFES) or in other way.
- Parameter ε for the tolerance of feasibility violation – the value depends on the user's request.

IV. Enhanced Search of Feasibility Region

The basic idea, how to enhance the search of the feasibility region, consists in exploitation of strategies different from DE in the search. Opposite-based optimization and adaptive controlled random search were applied for minimization of mean violation $\bar{v}$ in this study.

A. Opposition-Based Optimization

Application of opposition-based learning in stochastic optimization algorithms has appeared recently [26], [27] and it is called opposition-based optimization (OBO). Basic idea is to search for solution not only within the individuals of the population developed by evolutionary operators but also in the opposite part of the search space T. The opposite point to the point $\boldsymbol{x} \in T$, $T = \prod_{j=1}^{D}[l_j, u_j],\ l_j < u_j,\ j = 1, 2, \ldots, D$ is defined as symmetric with respect to the center of T by

$$\breve{\boldsymbol{x}} = \boldsymbol{l} + \boldsymbol{u} - \boldsymbol{x}. \tag{8}$$

The opposite population O of *NP* points to the population P according to relation (8) is generated occasionally, then the function values in the opposite points are evaluated and from the set $P \cup O$ the *NP* fittest individuals are selected to the new population. At the start of the optimization, $T = S$, which is $\boldsymbol{l} = \boldsymbol{a}$ and $\boldsymbol{u} = \boldsymbol{b}$. The search space T for the opposite points shrinks dynamically during the search process. If P is a matrix of size (*NP*, D), then $\boldsymbol{l} = \min(P)$ and $\boldsymbol{u} = \max(P)$, the functions $\min(\cdot)$ and $\max(\cdot)$ return the vectors of column minima and maxima, respectively. Thus, current dynamically shrinking search space for the opposite population can be written as

$$T = \prod_{j=1}^{D}[l_j, u_j],\ l_j < u_j,\ j = 1, 2, \ldots, D, \qquad \text{where} \quad \boldsymbol{l} = \min(P), \quad \boldsymbol{u} = \max(P). \tag{9}$$

The opposite population is generated after the initialization of the population for the first time. During the search process, the opposite population is generated in randomly selected generations if the condition of $rand(0,1) < \mathit{JR}$, where $rand(0,1)$ is a random value from uniform distribution on $[0,1)$ and *JR* (jumping rate) is an input parameter of the algorithm. The jumping rate is usually set to constant value, e.g. $\mathit{JR} = 0.3$ [26].

B. Adaptive Controlled Random Search

Another algorithm for enhanced search of the feasible solution is the adaptive controlled random search (CRS) with four competing local heuristics. This algorithm was applied successfully to the estimation of parameters in non-linear regression models [28].

Three of the competing local heuristics are based on a randomized reflection in the simplex Σ created by $D+1$ points chosen from P. A new trial point $\boldsymbol{y}$ is generated from the simplex by the relation

$$\boldsymbol{y} = \boldsymbol{g} + U\left(\boldsymbol{g} - \boldsymbol{x}_H\right), \tag{10}$$

where $\boldsymbol{x}_H = \arg\max_{\boldsymbol{x}\in\Sigma} f(\boldsymbol{x})$ and $\boldsymbol{g}$ is the centroid of remaining D points of the simplex Σ. The multiplication factor U is a random variable distributed uniformly in $[s, \alpha - s)$, $\alpha > 0$ and s being input parameters, $0 < s < \alpha/2$. All the $D + 1$ points of simplex are chosen randomly from P in two heuristics. Regarding the third heuristic, one point of the simplex is the point of P with the minimum objective function value and the remaining D points of the simplex Σ are chosen randomly from the remaining points of P. The fourth competing local heuristic is derived from differential evolution, where strategy DE/rand/1/bin is used and adaptive setting of control parameter F according to [15] is applied. The algorithm was used as described in [28] with the same setting of the parameters controlling the competition of local heuristics.

C. Enhanced Search of Feasible Individuals

The scheme of enhanced search of the feasible region is written in a pseudo-code in Figure 2. By the term "one generation" of an algorithm is meant one generation of DE (algorithm *b6e6rl*) or a run of another algorithm, where *NP* evaluations of objective function is needed. Notice that except the *b6e6rl* all the other algorithms minimize only the mean violation $\bar{v}$ as objective function. In the algorithm described in Figure 2 (hereafter denoted *CRSrOBO*) are shown all the enhanced search techniques used in this study. *CRSrOBO* appeared best performing in the experimental tests. The other algorithms tested miss one or more enhancement parts. The algorithm labeled by *CRSOBO* does not use the refreshment of population for the controlled random search, i. e. lines 4 to 6 in Figure 2 are not applied. The algorithm not using lines 4 to 6 and no opposition-based optimization (lines 10 to 12 Figure 2) is labeled *CRS*. The algorithm without enhancing the search of the feasible solutions by controlled random search (skipped lines 3 to 9 in Figure 2) is marked as *OBO*. The dynamically shrinking search space according to relation (9) is applied in all the algorithms using enhanced search by OBO, whereas the static search space S is used in refreshment of the population for CRS, because of no a priori information on allocation of the feasibility regions is assumed. One generation of OBO with static search space S is also applied immediately after the initialization of P at line 1 in Figure 2.

```
 1 generate an initial generation $P$
 2 while stopping condition not achieved
 3    if no individual feasible then
 4       if $rand(0,1) < \mathit{JR}_{\mathit{CRS}}$ then
 5          refresh $P$ using opposite population in $S$
 6       endif
 7       perform one generation of CRS
 8    endif
 9    perform one generation of b6e6rl
10    if no individual feasible & $rand(0,1) < \mathit{JR}_{\mathit{OBO}}$ then
11       perform one generation of OBO
12    endif
13    $P := Q$
14 endwhile
```

Fig. 2. **Algorithm 2** – Enhanced Search of Feasible Solutions

The mnemonic labels of the algorithms with enhanced search respect the applied techniques and the basic algorithm of competitive DE is not stressed in labels of variants because it is present in all the tested algorithms. Finally, if we omit all enhanced search (lines 3 to 8 and 10 to 12) then the rest is the algorithm *b6e6rl* described in Section III.

V. Experiments and Results

The modification of DE for the solution of constrained problems presented in [8] consists in alternating minimization of mean violation $\bar{v}$ and $f(\boldsymbol{x})$ using one generation of the *b6e6rl* algorithm. An extra minimization of mean violation could be considered as enhanced search of the feasibility region by *b6e6rl*. This variant labeled by *CEC10* hereafter is compared with the algorithms described in this study. All the variants of algorithm were implemented in Matlab.

The algorithms were tested on the benchmark set developed for the special CEC 2010 session [3]. The set consists of 18 problems at two levels of dimension. The code for test-function evaluation was loaded from the web page of the organizers [3].

Six algorithms were compared in experimental tests, namely algorithm labeled *CEC10*, simple competitive DE for constrained optimization (*b6e6rl*) and four new variants with enhanced search of the feasible solutions (*CRS*, *OBO*, *CRSOBO*, and *CRSrOBO*).

The control parameters of the algorithms were set to the same values for all the test problems, some parameters depend on the dimensionality of the test problems, the values follows:

- Size of the population, $NP = 50$ for $D = 10$, $NP = 100$ for $D = 30$.
- Parameters to control the competition, $n_0 = 2$, $\delta = 1/(5 \times H)$ for all the test problems.
- Stopping condition, MaxFES $= 2 \times 10^5$ for $D = 10$, MaxFES $= 6 \times 10^5$ for $D = 30$, as given in [3].
- Parameter for tolerance of feasibility violation, the value $\varepsilon = 0.0001$ is prescribed in [3].
- Jumping rate for OBO was set to $JR_{OBO} = 0.3$ as mostly applied in [27].
- Jumping rate of population refreshment for CRS was set to $JR_{CRS} = 0.1$ in order not to waste time by too frequent using static OBO.
- Control parameters of adaptive CRS were used the same as in [28].

No tuning of control-parameter was performed before the test experiments.

For each problem and dimension, 25 independent optimization runs were performed. From each run the function value $f(\boldsymbol{x}^*)$ found at the minimum of the mean violation $\bar{v}$ is returned together with the corresponding values of variables needed for the processing of the results presented in the tables. If there are more feasible solutions in the last generation, the minimum function value $f(\boldsymbol{x}^*)$ of the feasible solutions is returned.

TABLE II
COUNT OF FEASIBLE SOLUTIONS OF 25 FOR $D = 10$.

Problem	CEC10	b6e6rl	CRS	OBO	CRSOBO	CRSrOBO
1	25	25	25	25	25	25
2	25	25	24	18	22	21
3	25	25	25	24	25	25
4	25	25	25	25	25	25
5	4	0	0	16	14	16
6	2	0	1	23	21	20
7	25	25	25	25	25	25
8	25	25	25	25	25	25
9	3	2	0	12	18	22
10	3	5	3	11	20	17
11	20	0	22	0	24	24
12	25	16	25	21	25	25
13	25	25	25	25	25	25
14	25	25	25	25	25	25
15	24	25	25	22	24	24
16	19	25	20	25	23	24
17	19	21	21	19	23	21
18	22	24	25	16	23	20
Average	18.94	17.67	18.94	19.83	22.89	22.72

TABLE III
COUNT OF FEASIBLE SOLUTIONS OF 25 FOR $D = 30$.

Problem	CEC10	b6e6rl	CRS	OBO	CRSOBO	CRSrOBO
1	25	25	25	25	25	25
2	25	25	25	25	25	24
3	25	25	25	25	24	24
4	25	25	13	25	13	17
5	0	0	0	9	4	14
6	4	0	0	19	7	17
7	25	25	25	25	25	25
8	25	25	25	25	25	25
9	1	6	2	19	16	20
10	2	5	0	19	20	21
11	25	0	17	0	21	23
12	13	1	23	0	20	23
13	25	25	25	25	25	25
14	25	25	25	25	25	25
15	25	25	25	25	25	25
16	25	25	25	25	25	25
17	23	25	24	25	25	25
18	25	25	25	24	25	24
Average	19.06	17.33	18.28	20.28	20.83	22.61

Comparison of the feasibility rates obtained by all the tested variants are shown in Tables II and III, where counts of runs that found a feasible solution ($\bar{v} = 0$) of 25 runs are given. If we compare the results of the algorithms in problems 5, 6, and 9 to 12, we see a complementary effect of different enhanced techniques. Where CRS does not help, OBO is useful, and vice versa. Moreover, a synergic effect appears if more types of enhanced search are applied together. The best results were obtained by *CRSrOBO*, where the average count of the feasible solutions is almost 23 of 25 and the minimum of feasible solutions is 14, i.e. more than a half of 25.

The two-tailed Fisher's exact test was applied to the statistical comparison of the feasibility rates of the algorithms. All the algorithms were compared with respect to *CRSrOBO*. The values of the feasibility rate for which the proportion of the feasible solutions is different significantly from the corresponding proportion of *CRSrOBO* at level of 0.05 are underlined.

Detailed results of the best performing variant of enhanced competitive DE (*CRSrOBO*) in 18 test problems at two levels of the dimension are presented in Tables IV and V. The structure of the results follows the requirements given in [3]. The results obtained from 25 runs of each problem were sorted according to their values of mean violation $\bar{v}$ in an ascending way. Then the part of the feasible solutions was sorted according to their function values $f(\boldsymbol{x}^*)$ in an ascending way. The presented values of $f(\boldsymbol{x}^*)$ in columns Best, Median, and Worst refer to this order. The sorting according to the values of mean violation $\bar{v}$ can cause that the best or median of $f(\boldsymbol{x}^*)$ need not be necessarily less than the worst one in some of the problems. The values of $\bar{v}$ and c correspond to the row of Median. Three numbers in vector c indicate the number of violations more than 1, the number of violations in $(0.1,\ 1]$, and the number of violations in $(0.0001,\ 0.1]$. The mean and the standard deviation (Stdev) of $f(\boldsymbol{x}^*)$ are evaluated from all

TABLE IV
DETAILED RESULTS OF CRSROBO ALGORITHM AFTER ACHIEVING MAXFES FOR $D = 10$.

Problem	Best	Median	Worst	c	$\bar{v}$	Mean	Stdev	FR
1	-0.74731	-0.74731	-0.74056	0,0,0	0	-0.74623	0.00253	100 %
2	-0.85975	3.0025	3.1052	0,0,0	0	2.5482	1.3519	84 %
3	0	0	5.180e+11	0,0,0	0	2.072e+10	1.036e+11	100 %
4	-0.00001	-0.00001	-0.00001	0,0,0	0	-0.00001	0	100 %
5	6.8037	241.51	533.08	0,0,0	0	205.13	167.99	64 %
6	-36.480	203.55	469.88	0,0,0	0	197.07	145.76	80 %
7	0	0	3.9866	0,0,0	0	0.31893	1.1038	100 %
8	0	3.6229	40.876	0,0,0	0	6.6088	8.9421	100 %
9	2.099e+10	2.905e+12	5.438e+12	0,0,0	0	3.895e+12	3.727e+12	88 %
10	6.005e+10	1.437e+12	3.676e+12	0,0,0	0	3.989e+12	4.606e+12	68 %
11	-0.00152	-0.00152	-0.08734	0,0,0	0	-0.00496	0.01716	96 %
12	-305.49	-0.19925	-0.19925	0,0,0	0	-26.310	69.644	100 %
13	-68.429	-68.429	-68.429	0,0,0	0	-68.429	7.234e-08	100 %
14	6.8326	3.197e+08	1.496e+10	0,0,0	0	1.680e+09	3.389e+09	100 %
15	3.876e+11	2.925e+13	1.012e+14	0,0,0	0	4.242e+13	5.102e+13	96 %
16	0	0.90857	0.92313	0,0,0	0	0.74910	0.32623	96 %
17	17.319	242.92	751.13	0,0,0	0	247.91	186.64	84 %
18	113.90	4379.01	10223.3	0,0,0	0	5823.3	4195.9	80 %

TABLE V
DETAILED RESULTS OF CRSROBO ALGORITHM AFTER ACHIEVING MAXFES FOR $D = 30$.

Problem	Best	Median	Worst	c	$\bar{v}$	Mean	Stdev	FR
1	-0.82188	-0.82188	-0.81795	0,0,0	0	-0.82095	0.00169	100 %
2	2.0180	3.4494	4.1897	0,0,0	0	3.4143	0.63100	96 %
3	0	1.129e+08	9.749e+11	0,0,0	0	3.350e+12	5.015e+12	96 %
4	-3.263e-06	0.00191	0.92952	0,0,0	0	0.25626	0.42471	68 %
5	73.320	506.36	556.27	0,0,0	0	352.56	176.17	56 %
6	196.45	524.00	582.50	0,0,0	0	439.48	125.78	68 %
7	0	0	0	0,0,0	0	0	0	100 %
8	0	0	2.335e-26	0,0,0	0	1.081e-27	4.693e-27	100 %
9	2.937e+11	1.934e+13	2.445e+13	0,0,0	0	1.773e+13	1.672e+13	80 %
10	1.861e+11	1.822e+13	3.883e+13	0,0,0	0	1.869e+13	1.572e+13	84 %
11	-0.00039	-0.00039	0.01867	0,0,0	0	0.00113	0.00528	92 %
12	-0.19926	-0.19926	196.42	0,0,0	0	7.4997	39.368	92 %
13	-67.224	-65.433	-64.449	0,0,0	0	-65.617	0.74208	100 %
14	2.783e+08	3.130e+10	2.708e+11	0,0,0	0	4.778e+10	6.011e+10	100 %
15	1.111e+13	1.349e+14	2.599e+14	0,0,0	0	1.312e+14	7.212e+13	100 %
16	0.29150	0.80218	1.0364	0,0,0	0	0.78285	0.20885	100 %
17	86.084	781.38	1533.7	0,0,0	0	719.01	491.01	100 %
18	55.420	15580.9	42534.6	0,0,0	0	19026.4	14999.8	96 %

the 25 runs. The feasibility rate in the last column of tables is defined as

$$FR = 100 \times \frac{\text{number of runs with feasible solution}}{\text{number of all runs}}.$$

In comparison with the results of *CEC10* [8], not only the feasibility rate of *CRSrOBO* is significantly better in some problems, but also the minimum function values (column Best) found by *CRSrOBO* are less than those found by *CEC10* in more than a half of problems. Thus, the novel enhanced competitive DE variant combining adaptive controlled random search with population refreshment and opposition-based optimization for the search of feasible solution proved to be a substantial improvement of the competitive DE for constrained single-objective optimization.

VI. CONCLUSION

A novel competitive DE variant with enhanced search of the feasibility region for constrained single-objective optimization was proposed. This algorithm includes adaptive controlled random search with population refreshment by applying opposite points in the search space and opposition-based optimization. These techniques are used for the search of the feasibility region (minimization of violation), whereas the competitive differential evolution is used both for minimizing the constraints' violation and the function value. The novel algorithm proved to be a substantial improvement of the competitive DE for constrained single-objective optimization presented recently [8].

In near future, we are planning to compare this algorithm with good performing algorithms available in literature [5], [7] and with the other algorithms succeeded in the competition of Congress on Evolutionary Computation 2010 at Barcelona. It is needed for a decision on applicability of the algorithm in real-world problems. A more detailed analysis of the algorithms is intended. Its results can bring a new impuls for further development of algorithms for constrained single-objective optimization with enhanced search of the feasible solutions.

REFERENCES

[1] R. Storn and K. V. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimization*, vol. 11, pp. 341–359, 1997.

[2] K. V. Price, R. Storn, and J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Springer, 2005.

[3] R. Mallipeddi and P. N. Suganthan, "Problem definition and evaluation criteria for the CEC 2010 competition and special session on single objective constrained real-parameter optimization," Tech. Rep., Nanyang Technological University, April, 2010. [Online]. Available: http://www3.ntu.edu.sg/home/EPNSugan/

[4] J. Lampinen, "A constrained handling approach for differential evolution algorithm," in *IEEE Congress on Evolutionary Computation*, 2002, pp. 1468–1473.

[5] J. Brest, "Constrained real-parameter optimization with ε-self-adaptive differential evolution," in *Constraint-Handling in Evolutionary Optimization*, E. Mezura-Montez, Ed. Springer, 2009, pp. 73–93.

[6] J. Brest, S. Greiner, B. Boškovič, M. Mernik, and V. Žumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, pp. 646–657, 2006.

[7] R. Mallipeddi and P. N. Suganthan, "Ensemble of constraint handling techniques," *IEEE Transactions on Evolutionary Computation*, 2010, doi 10.1109/TEVC.2009.2033582.

[8] J. Tvrdík and R. Poláková, "Competitive differential evolution for constrained problems," in *IEEE Congress on Evolutionary Computation*, 2010, pp. 1632–1639.

[9] V. Feoktistov, *Differential Evolution in Search of Sotution*. Springer, 2006.

[10] P. Kaelo and M. M. Ali, "A numerical study of some modified differential evolution algorithms," *European J. Operational Research*, vol. 169, pp. 1176–1184, 2006.

[11] D. Zaharie, "A comparative analysis of crossover variants in differential evolution," in *Proceedings of IMCSIT 2007*, U. Markowska-Kaczmar and H. Kwasnicka, Eds. Wisla: PTI, 2007, pp. 171–181.

[12] D. Zaharie, "Influence of crossover on the behavior of differential evolution algorithms," *Applied Soft Computing*, vol. 9, pp. 1126–1138, 2009.

[13] R. Gämperle, S. D. Müller, and P. Koumoutsakos, "A parameter study for differential evolution," in *Advances in Intelligent Systems Fuzzy Systems, Evolutionary Computing*, A. Grmela and N. E. Mastorakis, Eds. Athens: WSEAS Press, 2002, pp. 293–298.

[14] D. Zaharie, "Critical values for the control parameter of the differential evolution algorithms," in *MENDEL 2002, 8th International Conference on Soft Computing*, R. Matoušek and P. Ošmera, Eds. Brno: University of Technology, 2002, pp. 62–67.

[15] M. M. Ali and A. Törn, "Population set based global optimization algorithms: Some modifications and numerical studies," *Computers and Operations Research*, vol. 31, pp. 1703–1725, 2004.

[16] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algortithm," *Soft Computing*, vol. 9, pp. 448–462, 2005.

[17] A. K. Qin and P. N. Suganthan, "Self-adaptive differential evolution for numerical optimization," in *IEEE Congress on Evolutionary Computation*, 2005, pp. 1785–1791.

[18] M. G. H. Omran, A. Salman, and A. P. Engelbrecht, "Self-adaptive differential evolution," in *Lecture Notes in Artifitial Intelligence*, ser. 3801. Springer, 2005, pp. 192–199.

[19] A. Qin, V. Huang, and P. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, pp. 398–417, 2009.

[20] J. Zhang and A. C. Sanderson, *Adaptive Differential Evolution, A Robust Approach to Multimodal Problem Optimization*. Springer, 2009.

[21] M. Weber, V. Tironen, and F. Neri, "Scale factor inheritance mechanism in distributed differential evolution," *Soft Computing*, vol. 14, pp. 1187–1207, 2010.

[22] J. Tvrdík, "Competitive differential evolution," in *MENDEL 2006, 12th International Conference on Soft Computing*, R. Matoušek and P. Ošmera, Eds. Brno: University of Technology, 2006, pp. 7–12.

[23] V. Kvasnička, J. Pospíchal, and P. Tiňo, *Evolutionary Algorithms*. Bratislava: Slovak Technical University, 2000, (In Slovak).

[24] J. Tvrdík, "Adaptive differential evolution and exponential crossover," in *Proceedings of IMCSIT 2008*, U. Markowska-Kaczmar and H. Kwasnicka, Eds. Wisla: PTI, 2008, pp. 927–931.

[25] J. Tvrdík, "Self-adaptive variants of differential evolution with exponential crossover," *Analele of West University Timisoara, Series Mathematics-Informatics*, vol. 47, pp. 151–168, 2009.

[26] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, pp. 64–79, 2008.

[27] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," in *Advances in Differential Evolution*, U. K. Chakraborty, Ed. Springer, 2008, pp. 155–171.

[28] J. Tvrdík, I. Křivý, and L. Mišík, "Adaptive population-based search: application to estimation of nonlinear regression parameters," *Computational Statistics and Data Analysis*, vol. 52, pp. 713–724, 2007.

Author Index